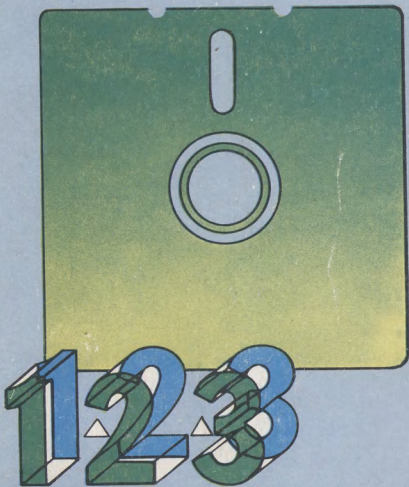


В.З. Аладьев
Н.А. Гершгорн

ВЫЧИСЛИТЕЛЬНЫЕ ЗАДАЧИ НА ПЕРСОНАЛЬНОМ КОМПЬЮТЕРЕ



**В.З. Аладьев
Н.А. Гершгорн**

ВЫЧИСЛИТЕЛЬНЫЕ ЗАДАЧИ НА ПЕРСОНАЛЬНОМ КОМПЬЮТЕРЕ

**КИЕВ
«ТЭХНИКА»
1991**



ББК 32.973.2
А45
УДК 519.682:681.3.06

Редакция литературы по электронике, кибернетике и связи
Зав. редакцией *З. В. Божко*
Издание осуществлено за счет средств автора

Аладьев В. З., Гершгорн Н. А.

А45 Вычислительные задачи на персональном компьютере.—
К. : Техника, 1991.— 245 с.
ISBN 5-335-01064-9

Описан пакет MathCAD 2.52 как новое уникальное средство автоматизации научно-технических задач вычислительного характера. Пакет позволяет оперировать с текстовой, графической и изобразительной информацией, подготовленной в соответствующих САД-системах или системах программирования.

Рассчитана на широкий круг пользователей персональных компьютеров, совместимых с IBM PC/XT/AT и PS/2. Может быть использована в качестве научно-методического пособия по курсу «Основы информатики и вычислительной техники».

А 2404090000-125 БЗ—4—16.91
М202(04)-90

ББК 32.973.2

ISBN 5-335-01064-9

© Аладьев В. З., Гершгорн Н. А., 1991

ПРЕДИСЛОВИЕ

Компьютеризация общества — одна из наиболее актуальных проблем современного технического прогресса, одним из путей решения которой является наличие специальной литературы различного назначения. В наши дни, наряду с относительно небольшим количеством больших и средних ЭВМ, предназначенных в основном для профессионального пользователя, в народное хозяйство страны начинают массово поступать как отечественные, так и зарубежные персональные компьютеры (ПК). Персональный характер использования данного класса ЭВМ ставит вопросы подготовки массового пользователя, значительную часть которых составляют непрофессиональные пользователи ПК. Имеющаяся на сегодня литература по ПК явно недостаточна и порой не удовлетворяет целому ряду важных требований. Более того, постоянное появление различного рода новых программных средств для ПК требует адекватного отражения основных из них в соответствующей специальной литературе. В связи с этим нами предпринята попытка создания серии книг по ПК и их программному обеспечению (ПО), используя опыт, полученный при создании серии книг по ЕС ЭВМ [1—4], СМ ЭВМ [5] и ПК ИСКРА-226 [6—9] и во время проведения лекций по вычислительной технике (ВТ) и программированию для специалистов различного уровня подготовки в ряде городов страны.

Планируемая серия книг посвящается ПО ПК, совместимых с хорошо известными за рубежом системами IBM PC/XT/AT и PS/2. Несмотря на все многообразие, ПО ПК можно сгруппировать в три основные группы:

операционные системы ПК;

инструментальные средства и системы программирования;

пакеты прикладных программ (ППП).

Операционные системы (ОС) явились естественным следствием развития вычислительной техники и ее программного обеспечения, предназначенного для управления ресурсами ЭВМ, включая и самого пользователя. В настоящее время ЭВМ любого класса обязательно поставляются пользователю с той или иной ОС, например, MS DOS для ПК, совместимых с IBM PC/XT/AT.

Для обеспечения пользователя ПК средствами разработки ПО любого назначения с ОС, как правило, поставляется набор инструментальных средств и систем программирования. Например, Turbo-C, Turbo-Basic, Turbo-Pascal, Turbo-Prolog, PL/1, Fortran-77, Modula и ряд других систем программирования.

В настоящее время ППП являются наиболее обширной областью программных средств ПК. Для удобства будем подразделять их на десять основных групп:

обработка текстовой информации (текстовые редакторы);
обработка графической информации (графические редакторы);
средства обработки различного рода электронных таблиц;
системы управления базами данных (СУБД);
коммуникационные пакеты для работы ПК в сетях связи;
пакеты, ориентированные на специальные применения;
пакеты для обеспечения наиболее важных сервисных функций;
интегрированные пакеты;
средства поддержки интеграции пакетов программ;
экспертные системы и базы знаний, САПР.

Предполагается, что в планируемую серию книг будут включены наиболее перспективные как зарубежные, так и отечественные программные средства (ПС) согласно приведенной классификации. При этом большое внимание будет уделяться опыту работы с предлагаемым средством и особенностям его использования, а также вытекающим из этого рекомендациям пользователю. По мере возможности будут предлагаться наиболее эффективные технологии применения данных средств для решения тех или иных прикладных задач пользователя. Довольно подробный обзор наиболее популярных ПС для ПК, совместимых с IBM PC/XT/AT и PS/2, в рамках приведенной классификации можно найти в первой книге запланированной серии, вышедшей в свет в 1990 г. [10].

Данная книга охватывает ряд основных программных средств ПК и в значительной мере представляет базовое ПО отечественного ПК ИСКРА 1030, совместимого с IBM PC/XT. В книге довольно детально рассматриваются вопросы использования операционной системы MS DOS, системы программирования на основе алгоритмического языка Basic-IBM, являющегося наиболее распространенным диалектом для ПК различных типов и положенного в основу отечественного гостандарта языка БЕЙСИК, а также одной из разновидностей текстового редактора. Книга представляет интерес в первую очередь для пользователей, начинающих применять в своей профессиональной деятельности ПК, совместимые с IBM PC/XT/AT и PS/2.

Придерживаясь выработанной методики подготовки книг, вместе с тем не будем строго придерживаться заранее объявленного перечня программных средств, а будем руководствоваться текущими тенденциями развития ПО ПК и перспективами его использования на отечественном рынке и наряду с этим учитывать рекомендации и пожелания фирм, с которыми фирма VASCO Ltd. поддерживает творческие и коммерческие отношения. Это, в первую очередь, относится к таким известным фирмам как Borland International Inc., MathSoft Inc., Expert Choice Inc., PKWARE Inc., The Cobb Group Inc. (США) и некоторым другим.

В настоящее время проводятся работы по созданию целого ряда книг по программному обеспечению ПК, совместимых с IBM PC/XT/AT и PS/2, в частности следующих двух книг:

Аладьев В. З., Гиссовский В. Б. Компьютерная смесь (знакомит с известными пакетами Sprint, Quattro, Expert Choice, MathCAD, PKWARE, News Master, Auto Sketch и др.);

Аладьев В. З., Гиссовский В. Б., Пицюра И. С. Система программирования Turbo-Pascal (знакомит с объектно-ориентированным средством программирования Turbo-Pascal 6.0).

Для более точного определения начального тиража книг заказы на них следует направлять в адрес фирмы VASCO Ltd. В заказе рекомендуется указать заинтересованность в последующих изданиях. На основе данной информации будет формироваться банк потенциальных читателей последующих книг предлагаемой серии.

Созданная в апреле 1991 г. фирма VASCO (Victor Aladyev Software Company), наряду с разработкой программных средств и исследованиями в области теории и приложений однородных структур [12], одной из основных своих задач ставит подготовку и издание серий книг по ПС для ПК, совместимых с IBM PC/XT/AT и PS/2. В этом плане фирма VASCO Ltd. заинтересована в расширении творческих и коммерческих контактов с другими отечественными фирмами, организациями, ведомствами и частными лицами, в той или иной мере использующими в своей деятельности персональную вычислительную технику или имеющими дело с вопросами издательской деятельности.

Предлагаемая книга организована таким образом, что не только дает всю необходимую информацию по работе с пакетом MathCAD, но и предлагает наиболее эффективные приемы использования, акцентируя внимание на сильных и слабых сторонах. По всем вопросам, связанным с пакетом MathCAD, можно обращаться по адресу фирмы VASCO Ltd.

Приводимые примеры в основном ориентированы на иллюстрацию особенностей и основных возможностей пакета MathCAD и служат для помощи начинающему пользователю в овладении основными навыками и приемами работы с пакетом. Целый ряд примеров можно использовать в качестве готовых программных конструкций в практической работе пользователя.

Настоящая книга носит методический, справочный и практический характер и рассчитана прежде всего на читателя, знакомого с основами работы на ПК, совместимых с IBM PC/XT/AT и PS/2. Наряду с этим, ее может использовать и начинающий пользователь ПК данных типов. Предлагаемая книга создана на основе опыта эксплуатации и всесторонней апробации пакета MathCAD 2.52 и ориентирована на широкий круг читателей различных специальностей, имеющих в своей профессиональной деятельности дело с различного рода вычислительными задачами.

Всесторонняя апробация пакета MathCAD 2.52 и создание системы MINIDOS/MathCAD, описанные в настоящей книге, проводились в мае—ноябре 1990 г. в ПТИП Министерства строительства ЭССР на отечественном ПК ИСКРА 1030, совместимом с IBM PC/XT.

По всем вопросам и предложениям, связанным с вышеупомянутой проблематикой, следует обращаться по адресу:

ЭССР, Таллинн 200106
ул. Комсомоли, 1, км. 221
Фирма VASCO Ltd.

Телефон: (8—0142)—601—937, 453—888

Телефакс: 453—923

Директор фирмы VASCO Ltd. **Виктор Аладьев**

Одной из важнейших компонент инструментария современного пользователя ПК любого типа являются утилиты — сервисные программные средства широкого назначения, позволяющие существенно повышать эффективность разработок и эксплуатации самого компьютера. В настоящее время разработано и находится в обращении большое количество таких средств самого различного назначения. Однако большинство отечественных пользователей данным средствам уделяет недостаточно внимания. В связи с этим нами предпринята попытка систематизации и описания как наиболее доступных в стране утилит, так и популярных зарубежных средств [45]. В процессе данной работы апробировано более 100 утилит самого широкого назначения зарубежного и отечественного производства, которые объединены в пакет **UTILITY**. Среди утилит пакета большое внимание уделено антивирусным, информационным и средствам расширения возможностей периферии ПК (принтера, монитора, клавиатуры, дисковых УВВ и др.), а также средствам архивирования файлов, обслуживания файловых структур, администрирования работы ПК. Некоторые из указанных утилит в последние годы входили в число лучших утилит по материалам известных зарубежных компьютерных журналов. На сегодня пакет **UTILITY** является уникальным отечественным средством подобного типа для ПК, совместимых с IBM PC/XT/AT и PS/2.

Пакет занимает шесть 5,25-дискет формата 720 К и поставляется на носителях заказчика с предварительной оплатой 800 р. на р/с 3345924 в Эстпромстройкоммерцбанке г. Таллинна; МФО 781200 (с пометкой «УТИЛИТЫ»). Поставка в адрес заказчика производится по получении достаточного количества отформатированных дискет любого формата (с копией платежного поручения и четким обратным адресом) в адрес ППО фирмы VASCO Ltd.: Таллинн 200038, Партизаны 13—75. Телефон для справок: 601-937.

В заключении пользуемся случаем выразить благодарность фирме MathSoft Inc. (США) за предоставленную возможность ознакомления со своими новыми разработками в области создания программного обеспечения ПФ, ориентированного на решение вычислительных задач в различных сферах человеческой деятельности.

ВВЕДЕНИЕ

ЭВМ любого класса в первую очередь предназначены для различного рода вычислений и только после этого для решения других задач автоматизации интеллектуального труда.

В настоящее время вычислительные задачи с помощью ПК можно решать на трех основных уровнях: 1) в среде выбранной системы программирования; 2) на основе специальных вычислительных процедур и программ; 3) с помощью специальных пакетов программ той или иной степени общности.

На *первом уровне* пользователь в среде некоторой системы программирования (например, Basic, Turbo-C, Turbo-Pascal, Fortran и др.) может запрограммировать и выполнить практически любой вычислительный алгоритм. Такая универсальность является несомненным преимуществом работы на первом уровне. Но это требует не только достаточного профессионализма в программировании, но и хорошего владения соответствующим аппаратом вычислительной математики [8—12,46].

На *втором уровне* пользователь может использовать для проведения расчетов соответствующие программы, разработанные в той или иной системе программирования и настраиваемые на требуемый режим выполнения. В этом случае пользователь существенно снижает затраты на получение искомых результатов вычислений при относительно небольших требованиях к его программистской квалификации. Однако данный подход (называемый иногда методом стандартных программ) в значительной мере узок и специализирован по предоставляемым средствам, а в ряде случаев сами средства далеки от универсальности [8, 14—18].

К *третьему уровню* относятся развитые пакеты программ, предназначенные для обеспечения решения различного рода вычислительных задач на ПК. Среди пакетов данной группы можно отметить REDUCE, предназначенный для алгебраических символьных преобразований. В отличие от него, пакет EUREKA фирмы Borland International Inc. (США) служит для численного решения широкого класса математических и коммерческих задач: системы линейных и нелинейных уравнений, интерполяция, вычисления определенных интегралов и производных, решение дифференциальных уравнений, матричные операции, метод наименьших квадратов, дискретное преобразование Фурье и др. Пакет ориентирован на работу с IBM PC/XT/AT и требует для своего функционирования не менее 384 К оперативной памяти.

Пакет Turbo-EUREKA той же фирмы является дальнейшим развитием пакета EUREKA и дополнительно к перечисленному включает экранный редактор, развитые систему меню, HELP-функцию, элек-

тронный калькулятор, средства для проведения разнообразных коммерческих расчетов, решения минимаксных задач, а также возможность вывода различных результатов как на экран, так и на принтер и/или плоттер.

В значительной мере к пакету EUREKA примыкают и пакеты NUMAT-1/NUMAT-2 DCP фирмы Robotron (Германия) для решения задач вычислительной математики. Пакеты STAVE DCP и OPTI-LO DCP той же фирмы представляют собой диалоговые системы решения задач соответственно математической статистики и линейной оптимизации в различных областях применения.

Пакеты CSS и StatGraf фирмы StatSoft Inc. (США) предоставляют пользователю широкие возможности по обработке статистических данных самого разного характера. Пакет программ PHASER предназначен для проведения различного рода экспериментов на ПК, совместимых с IBM PC/XT/AT, с дифференциальными и разностными уравнениями, а также динамическими системами в интерактивном режиме с использованием графических дисплеев типов CGA, EGA, VGA. Наряду с перечисленными, имеется множество других программных средств, которые можно отнести к третьему уровню автоматизации решения вычислительных задач на ПК [47].

Однако среди средств третьего уровня, на наш взгляд, можно особо выделить интегрированный многофункциональный пакет MathCAD фирмы MathSoft Inc. (США). Само название пакета представляет собой аббревиатуру выражения **Mathematical Computer Aided Design** (*математическое автоматизированное проектирование*), что говорит о назначении пакета — решение различного рода вычислительных задач, описываемых средствами математики. В ряде приложений аббревиатура «CAD» (Computer Aided Drafting) расшифровывается и как *автоматизированное проектирование чертежей*. В данной книге под «CAD» будет пониматься первая интерпретация, как имеющая более широкий смысл.

Возможности пакета MathCAD позволяют не только изящно описывать вычислительный алгоритм в общепринятых математических обозначениях и выполнять его, но и оформлять полученные результаты в обычном математическом виде. При этом пакет весьма удачно сочетает элегантность математического описания решаемой проблемы, вычислительные возможности и средства отображения информации вашего ПК. Интегрированный пакет MathCAD — это еще один шаг к совершенствованию в области компьютерной технологии решения вычислительных задач и автоматизированного проектирования научно-технических исследований в различных областях человеческой деятельности. Навыки, полученные при работе в среде пакета MathCAD, заложат хорошую основу для перехода к эксплуатации других программных средств подобного типа.

Представляя в среде пакета MathCAD вычислительный алгоритм и поясняющее его описание, пользователь формирует так называемый *документ*, который можно затем сохранять для последующего использования. Возможности пакета позволяют исследователю работать в его среде с проблемой подобно тому, как он это делает за столом.

В дальнейшем можно модифицировать как вычислительную часть, так и имеющуюся текстовую, графическую и иллюстративную информацию документа. При этом допускается включение в документ чертежей и рисунков, полученных с помощью других известных CAD-программ и систем.

Хорошим введением в основы CAD-программ является книга [29], в которой описываются принципы, методы и средства, общие для компьютерных приложений задач проектирования в вычислительных науках и технике. При этом сама CAD рассматривается как дисциплина, определяющая основы использования компьютерных систем для задач проектирования. Вопросам теории систем компьютерного проектирования (CAST) была посвящена также интересная конференция [30].

Готовый документ можно вывести на экран дисплея, принтер и/или плоттер, сохранив всю первоначальную изящность оформления, что особенно актуально при подготовке различного рода научно-технических отчетов, статей, рукописей книг и т. д.

Настоящая работа выполнена в рамках творческого сотрудничества ПТИП Минстроя (Эстония) и MathSoft Inc. (США) и раскрывает одно из важнейших применений ПК в различных областях человеческой деятельности. В настоящее время время свыше 300 тыс. систем MathCAD используются во всем мире научными, инженерно-техническими работниками, проектировщиками, конструкторами различных специальностей для решения разнообразных профессиональных задач вычислительного характера. Наряду с этим, наш опыт показывает, что пакет MathCAD можно успешно использовать в учебных заведениях страны в качестве великолепного пособия по курсу «Вычислительные методы и программирование», основы для создания различного рода методик по курсам высшей математики, физики и химии, а также эффективного средства по подготовке к печати рукописей работ, носящих ярко выраженный математический характер. Следует отметить, что и рукопись настоящей книги была подготовлена с помощью пакета MathCAD.

Универсальность и высокая эффективность системы MathCAD способствовали становлению ее в качестве международного стандарта для программных средств такого рода, а также позволили найти новых ее приверженцев в различных областях человеческой деятельности. Система MathCAD достаточно проста в обучении и использовании и не требует большого опыта работы на ПК, совместимых с IBM PC/XT/AT или PS/2. Практические вопросы применения системы MathCAD для решения задач САПР научно-технического характера можно найти в работе [43].

ОСНОВНЫЕ ХАРАКТЕРИСТИКИ ПАКЕТА MathCAD И ПОДГОТОВКА К РАБОТЕ С НИМ

В данной главе представлена краткая характеристика основных возможностей пакета, принципы его организации, подготовка его к работе и требуемые для этого ресурсы ПК, совместимого с IBM PC/XT/AT или PS/2. В настоящей книге рассматривается версия 2.52 пакета MathCAD, но, не оговаривая этого особо, в дальнейшем будут использоваться термины «пакет» или «MathCAD», эквивалентные между собой и обозначающие пакет MathCAD 2.52.

1.1. ОСНОВНЫЕ ХАРАКТЕРИСТИКИ ПАКЕТА MathCAD

Интегрированный многофункциональный пакет MathCAD предназначен для решения различного рода вычислительных задач, алгоритм которых описывается в общепринятых математических терминах и обозначениях. Наряду с этим пакет имеет развитые средства работы с текстовой, графической и иллюстративной информацией, представляя в совокупности развитое интегрированное средство с элементами искусственного интеллекта для автоматизации решения задач вычислительного характера. Главные характеристики пакета можно подразделить на три основные группы: интерфейс с пользователем, вычислительные возможности и возможности по работе с графо-текстовой информацией. При этом в процессе общения с пакетом в интерактивном режиме пользователь получает следующие возможности:

1) в любом месте экрана помещать текст, математические конструкции в общепринятом виде, графики, рисунки, диаграммы, схемы и т. п.;

2) легко вводить графические объекты (ГО) и достаточно сложные математические конструкции, помещая их на свободное место экрана; при этом в автоматическом режиме вычислений в случае появления ошибки сразу же происходит ее идентификация;

3) с большим произволом редактировать текст, ГО и математические конструкции, а также документ в целом;

4) редактировать два документа одновременно на основе разбиения экрана на два независимых окна;

5) вводить по мере необходимости команды, управляющие работой пакета, по их именам, посредством ключей, определяемых одной или несколькими клавишами, или через специальное командное меню путем выбора требуемого действия;

6) по клавише 'F1' в любое время получать краткую информацию по всем видам работ, поддерживаемых пакетом;

7) выводить копию всего или части текущего документа на принтер, плоттер или в дисковый файл;

8) сохранять текущий документ в дисковом файле;
9) изменять глобальные или локальные форматы результатов вычислений и/или ГО текущего документа, а также основные характеристики пакета;

10) запрашивать выполнение функций системы MS DOS.

Из вычислительных возможностей пакета можно выделить следующие:

1) проведение вычислений с точностью до 15 десятичных цифр после запятой;

2) проверка на соответствие единиц измерений вычисляемых величин и их размерности (использование вычислений с размерными величинами);

3) наличие встроенного решателя систем уравнений и неравенств;

4) вычисления производных (обычных и частных) и интегралов (обычных, многомерных и контурных);

5) выполнение операций ранжированного суммирования и умножения;

6) использование 8-, 10-, 16-ричных и комплексных чисел и переменных;

7) встроенные математические функции (общим числом 70) 15 групп, включая функции Бесселя, комплексного аргумента, популяционной статистики, линейной регрессии, интерполяции, быстрого дискретного преобразования Фурье, матрично-векторные и целый ряд других;

8) использование функций, определяемых самим пользователем;

9) развитые средства доступа к файлам данных на дисках;

10) операции с векторами и матрицами, включая умножение, обращение, транспозицию, вычисление определителя матрицы, векторную и матричную сортировки и целый ряд других;

11) использование кубической сплайн-интерполяции;

12) использование массивов, содержащих до 8000 элементов.

Наконец, пакет предоставляет пользователю широкие возможности по работе с текстовой, графической и иллюстративной информацией. Это обеспечивается как средствами создания и редактирования данных видов информации, так и поддержкой пакетом широкого набора типов дисплеев, принтеров и плоттеров. Из представленных возможностей в этом плане можно отметить следующие:

1) обработка документов практически неограниченного размера (определяемого объемом доступной памяти ПК) и вывод их на печать (принтер, плоттер) полностью или частично;

2) наличие развитых средств редактирования текстовой и графической информации документа;

3) использование единого клавишного принципа генерации ГО;

4) создание ГО из широкого набора элементов (13 типов линий, линии координатной сетки, логарифмические оси координат, специальные символы);

5) возможность создания одного и более графиков в одном ГО;

6) возможность построения трехмерных поверхностных графиков в требуемой пользователем перспективе;

7) существование неограниченного числа ГО в текущем документе;

8) использование рисунков, схем, диаграмм и так далее из файлов, подготовленных графическими пакетами AutoCAD и AutoSketch фирмы AutoDesk Ltd. (Англия) и другими HPGL-совместимыми рисующими CAD-программами;

9) установка глобальных и локальных форматов для ГО;

10) поддержка 17 греческих букв и символов в кодах PRN (принтера) и ASCII;

11) обеспечение работы с EXEC-файлами и файлом конфигурации пакета;

12) поддержка широкого набора типов дисплеев, принтеров и плоттеров, а также спецификаций расширенной памяти типов (Intel | Lotus | Microsoft).

Как следует из уже перечисленного, пакет MathCAD предоставляет весьма широкие возможности любому пользователю, применяющему его для решения своих вычислительных задач из различных областей. В последующих главах будут достаточно подробно освещены вопросы подготовки, редактирования и вывода на печать документов в среде пакета MathCAD; организации вычислений согласно требуемому алгоритму с использованием единиц измерений, различных систем счисления, встроенных и пользовательских функций, уравнений, векторов, матриц и т. д., а также работы с различного рода графической двумерной и трехмерной информацией, включая методы использования различного типа иллюстративного материала из CAD-программ и программ, совместимых с HPGL-программами, т. е. с программами, написанными на языке Hewlett Packard Graphics.

Полнота изложения материала в большинстве случаев не требует обращения к фирменной документации [19—22], хотя наличие ее представляется нам полезным. Приводимые на протяжении последующих глав примеры иллюстрируют не только основные возможности пакета MathCAD и приемы работы с ним, но и в целом ряде случаев отражают особенности его использования. При этом структурная организация книги такова, что позволит пользователю легко выбирать необходимую информацию именно по тем возможностям пакета, которые в данный момент требуются. Вся дополнительная и справочная информация по пакету вынесена в соответствующие приложения, чтобы не загромождать основной текст такими данными.

В период с 1986 г. по 1989 г. фирма MathSoft Inc. поставила на рынок ряд версий пакета MathCAD, из которых наиболее известными являются пакеты версий 1.1, 2.0, 2.5 и 2.52. Именно на основе версии MathCAD 2.52, поступившей на рынок в 1989 г. и являющейся значительным развитием предыдущих версий, и ведется изложение в данной книге. При этом сохраняется обычный принцип совместимости версий сверху вниз, т. е. MathCAD 2.52 поддерживает файлы документов, созданные предыдущими версиями 1.1, 2.0 и 2.5 пакета для IBM PC/XT/AT и версией 2.0 для ПК Macintosh, тогда как обратное в общем случае неверно. Ввиду новизны данного программного средства для отечественного пользователя и наличия более совершенной версии 2.52 пакета нецелесообразно рассматривать в деталях вопрос совместимости версий пакета MathCAD. Заинтересованного читателя

отсылаем за справками к фирменной документации [22]. На протяжении дальнейшего изложения под терминами «пакет» и «MathCAD» будет, если не оговорено противного, пониматься именно пакет MathCAD 2.52.

Учитывая большое число реальных и потенциальных пользователей пакета MathCAD и сопутствующих ему программных средств, фирмой MathSoft Inc. ежеквартально издается специальный журнал [23], в котором пользователь может получить самые последние рекомендации по более полному и эффективному использованию пакета, информацию о новых областях его применения, рекламе новых сопутствующих и подобных ему программных средствах, а также публикациях в этом направлении. Официальным пользователям пакета MathCAD этот журнал поставляется бесплатно.

1.2. СОСТАВ ПАКЕТА, ТРЕБУЕМЫЕ РЕСУРСЫ И КОМПОНОВКА ПАКЕТА для КОНКРЕТНЫХ УСЛОВИЙ ЭКСПЛУАТАЦИИ

Пакет MathCAD предназначен для выполнения на ПК, совместимых с IBM PC/XT/AT, включая серию IBM PS/2. Требуемые для его функционирования минимальные ресурсы сводятся к следующему: оперативная память не менее 512 К, графический монитор и накопитель на гибком магнитном диске (НГМД) для дискет емкостью не менее 360 К. Естественно, что при функционировании пакета на таких ресурсах не раскрываются эффективно все его возможности. Поэтому для эффективного функционирования пакета рекомендуется следующая конфигурация ПК, совместимого с IBM PC/XT/AT или PS/2:

- 1) ОС MS DOS версии 2.0 и выше (желательно версии 3.3);
- 2) оперативная память не менее 512 К (рекомендуется не менее 640 К; дополнительно пакет поддерживает расширяемую память спецификаций типа Intel, Lotus и MicroSoft);
- 3) графический монитор, совместимый с одним из следующих типов: IBM CGA, EGA или VGA; Hercules Monochrome; Toshiba T3100; серии AT & T 6300;
- 4) принтер и/или плоттер, совместимые с устройствами печати (прил. 1);
- 5) два НГМД или один НГМД и один жесткий диск (наличие последнего весьма желательно);
- 6) сопроцессор Intel {8087 | 80287 | 80387} необязателен, но поддерживается пакетом и рекомендуется при работе с ним.

При этом следует отметить один существенный момент. Наш опыт работы с пакетом (и, в первую очередь, на отечественном ПК ИСКРА 1030) показал, что перечисленные требования к конфигурации ПК вполне достаточны, но для эффективного использования пакета в промышленных целях в полном объеме его возможностей требуются достаточно мощные ПК как по быстродействию процессора, так и по объему оперативной памяти. Например, ПК с памятью 512 К не позволяет работать в среде пакета MathCAD с массивами максимального размера (8000 элементов). Более того, по временным характеристикам (особенно при работе в его среде со сложными ранжированными,

интегральными, матричными и другими конструкциями) пакет не отвечает требованиям промышленной эксплуатации на ПК невысокого быстродействия.

В дальнейшем будут приведены некоторые временные характеристики пакета при решении различного типа задач на ПК ИСКРА 1030. Поэтому на таком (в целом неплохом) отечественном ПК, как ИСКРА 1030, пакет предназначен, скорее, для целей использования его в качестве, например, развитого изобразительного средства для широкого класса естественно-научных задач вычислительного характера, позволяющего не только представлять вычислительные алгоритмы в обычных математических обозначениях, но и снабжать их графической информацией в двух и трех измерениях, а также необходимыми текстовыми и иллюстративными данными. Эта особенность пакета позволяет использовать его и на малопроизводительных ПК также в качестве хорошего средства для подготовки различного типа документов, имеющих ярко выраженный математический характер (статьи, книги, отчеты и т. д.).

В дальнейшем изложении используются следующие основные соглашения [10] при описании тех или иных конструкций и возможностей пакета MathCAD:

необязательные элементы заключаются в квадратные скобки;

ввод каждой команды завершается нажатием клавиши 'Enter';

альтернативные элементы разделяются прямой чертой «|» и заключаются в фигурные скобки. Например, {A | B} — A или B;

смысл элемента указывается в угловых скобках «...»;

соединение наименований клавиш знаком «+» говорит о необходимости одновременного нажатия этих клавиш;

выражения «системный» и «пакетный» относятся соответственно к ОС MS DOS и пакету MathCAD;

под термином «выражение» понимается любая допустимая пакетом математическая конструкция, состоящая из чисел, переменных и функций, объединенных знаками математических операций, поддерживаемых пакетом;

под выражением «по умолчанию» понимается значение той или иной характеристики MS DOS или пакета MathCAD, которое используется, если пользователем не определено для нее другое значение;

используя различного типа имена (переменных, функций, команд и т. д.), под *буквозависимыми* будем понимать такие имена, в которых использование прописных и строчных букв существенно. Например, буквозависимое имя 'AVZ' отлично от имен 'Avz' и 'avz', тогда как в случае *буквонезависимости* все эти три имени эквивалентны. Если не оговорено противного, то используемые в книге имена будут полагаться *буквонезависимыми*;

приводимые в книге временные оценки и характеристики получены при эксплуатации пакета на отечественном ПК ИСКРА 1030;

приводимые в качестве иллюстрационных примеров фрагменты документов пакета выделяются в тексте наборами символов «[N]» (начало) и «[K]» (конец) соответственно в их левых верхнем и нижнем

углах. Относящиеся к примерам рисунки в связи с их тесной привязкой к месту в документе приводятся в прил. 14.

При работе в среде пакета MathCAD много внимания уделяется файловой структуре данных и программ, поэтому для описания местоположения требуемого файла будем использовать обозначения, принятые в книге [10]:

⟨ИУ:⟩ — имя устройства с двоеточием; по умолчанию полагается активное или системное устройство;

⟨путь⟩ — цепочка подкаталогов, разделенных символами «\» ;

⟨ИФ⟩ — имя файла, удовлетворяющее соглашениям MS DOS;

⟨.РФ⟩ — расширение имени файла, удовлетворяющее соглашениям MS DOS;

⟨СФ⟩ — спецификатор файла, представляющий собой конструкцию вида [⟨ИУ:⟩] [⟨путь⟩] ⟨ИФ⟩ [⟨.РФ⟩].

По умолчанию для параметров ⟨ИУ:⟩, ⟨путь⟩ и ⟨.РФ⟩ полагаются соответственно активное устройство УВВ, текущий каталог и три символа в зависимости от типа файла, используемого в среде пакета MathCAD.

Следует отметить, что на эффективность функционирования пакета особое влияние оказывают: доступный объем оперативной памяти, наличие сопроцессора и жесткого диска, а также тактовая частота процессора, определяющая его производительность. Тогда как качество выводимых документов, подготовленных пакетом, определяется типами имеющихся монитора, принтера и/или плоттера. Естественно, чем более современной моделью ПК обладает пользователь, тем эффективнее он может организовать использование пакета MathCAD для решения своих задач.

Пользователю пакет поставляется, как правило, в одном из двух основных вариантов: комплект фирменной документации [19—22] и системные файлы пакета на двух дискетах размером 5,25 дюйма или одной дискете размером 3,5 дюйма. В дальнейшем дискеты таких размеров будем называть соответственно 5,25-дискеты и 3,5-дискеты. Полный состав файлов пакета и краткая их характеристика приведены в табл. 1. Дальнейшее изложение предполагает знакомство читателя с основами работы в операционной среде MS DOS и общепринятой терминологией, для чего рекомендуется ознакомиться с гл. 4 книги [10] или с другими источниками по ОС MS DOS любой версии.

Файлы под номерами 1—6 являются обязательными, тогда как остальные носят дополнительный, сервисный, информационный или иллюстративный характер и могут при конкретных компоновках пакета не использоваться. При поставке пакета на 3,5-дискете все его файлы размещаются на одном томе, тогда как в случае 5,25-дискет на системном томе размещаются только файл MCAD.EXE и четыре файла с примерами MathCAD-документов, а на дополнительном томе — все остальные файлы, поставляемые с пакетом. Если не реорганизована файловая структура пакета на носителях, то именно дополнительный том должен постоянно находиться на активном устройстве НГМД. Приведем несколько более подробную характеристику файлов пакета.

Файл MCAD.EXE содержит резидентное ядро пакета, загружаемое один раз и находящееся в памяти в течение всего его функционирования.

Таблица 1

№	Имя файла	Назначение файла пакета MathCAD
1	MCAD.EXE	Содержит резидентное ядро пакета
2	MCAD.OVL	Содержит оверлейный модуль пакета
3	MCAD.MCP	Содержит основные драйверы устройств печати, поддерживаемых пакетом
4	*.MCF	Содержат таблицы-шрифты символов
5	MC*.PS	Используется только PostScript-принтером
6	MCAD.HLP	Содержит HELP-информацию по пакету
7	MCSTRANS.EXE	Утилита для перевода графических объектов, подготовленных в HPGL-формате, в формат пакета MathCAD
8	MORE.MCP	Содержит дополнительные драйверы устройств печати
9	DIRECTRY.TXT	Краткая информация о файлах пакета
10	README.TXT	Дополнительная информация по пакету
11	*.MCD	Примеры документов, подготовленных в среде пакета MathCAD
12	*.PRN	Данные для одного из документов
13	*.MCS	Рисунки для некоторых документов, подготовленных пакетами AutoCAD и TurboCAD
14	MCAD.MCC	Содержит конфигурацию пакета

ния. Во время работы с пакетом обращений к данному файлу не происходит. Объем ядра составляет 283 К, что уже само по себе говорит о его возможностях.

Файл **MCAD.OVL** содержит часть пакета, имеющую оверлейную структуру, модули которой время от времени загружаются в память в период функционирования пакета. Объем оверлейной части пакета составляет 108 К, что, принимая во внимание и объем ядра, в определенной мере говорит о широких возможностях пакета MathCAD в целом.

Файл **MCAD.MCP** содержит основные драйверы для принтеров и плоттеров, поддерживаемых пакетом. Он является резидентным и нужен только при выводе информации на печать. Файл **MORE.MCP** содержит дополнительные драйверы устройств печати. Использовать драйверы из этого файла можно путем добавления их в основной файл **MCAD.MCP**. Подробнее об этом можно прочитать в прил. 1. Следует отметить, что **MCAD.MCP** и **MORE.MCP** являются обычными текстовыми файлами в коде ASCII и доступны для обработки большинством текстовых процессоров.

Файлы с именами ***.MCF** (MCF-файлы) содержат таблицы-шифры символов, выводимых на различные типы мониторов и принтеров, а файлы с именами **MC*.PS** используются только принтерами типа PostScript и представляют собой также текстовые ASCII-файлы.

Файл **MCAD.HLP** содержит справочную информацию по пакету, которую можно вывести на экран в любое время по функциональной клавише 'F1'. Содержимое данного текстового файла можно вывести на экран или печать системными командами COPY, PRINT или любым текстовым редактором.

Файл **MCSTRANS.EXE** содержит утилиту, служащую для перевода графических объектов, подготовленных в HPGL-формате, в MathCAD-формат. Подробнее об этом средстве будет говориться ниже.

Файл **DIRECTRY.TXT** содержит перечень всех файлов пакета MathCAD на поставляемых пользователю дискетах, распределение на дискетах (системная или дополнительная для случая двух дискет) и краткое их назначение. Файл является текстовым в коде ASCII и легко выводится на экран или печать.

Файл **README.TXT** также является текстовым и содержит некоторые рекомендации и последние уточнения по работе с пакетом. Файл легко выводится на экран или печать.

Файлы с именами ***.MCD** (MCD-файлы) содержат примеры документов, подготовленных пакетом MathCAD, которые можно загружать и выполнять в среде пакета. Предлагаемые документы носят не только иллюстративный характер, но могут представлять и самостоятельный интерес в качестве готовых документов или их макетов, решающих те или иные вычислительные задачи. Полный перечень MCD-файлов пакета с их краткой характеристикой приведен в прил. 2. Уже данный перечень позволяет в определенной мере судить о возможностях пакета с точки зрения его приложений. Файлы с именами ***.PRN** содержат данные для одного из упомянутых документов, а файлы с именами ***.MCS** — эскизы ГО для некоторых документов, взятые из программ, подготовленных пакетами AutoCAD и TurboCAD.

Наконец, файл **MCAD.MCC** (как правило не поставляется) содержит информацию о конфигурации пакета. При частом использовании пакета пользователю рекомендуется его создать, чтобы упростить настройку пакета на требуемый режим эксплуатации при его загрузке. Подробнее об этом речь будет идти ниже.

После получения фирменного ПО пакета MathCAD в ряде случаев целесообразно реорганизовать его файловое хозяйство в зависимости от ресурсов имеющегося ПК и режима эксплуатации самого пакета. Здесь рассмотрим несколько наиболее общих подходов. Если ваш ПК имеет жесткий диск, то рекомендуется все необходимые файлы пакета поместить на этот диск в подкаталог с именем, скажем, **MathCAD**. В большинстве случаев дисковые устройства ПК имеют логические имена: 'A', 'B' — НГМД и 'C' — НМД (жесткий диск). Поэтому в дальнейшем, если не оговорено противного, именно это распределение имен устройств будем иметь в виду, при этом активным будет полагаться устройство с логическим именем 'A'. В свете сказанного последством следующих четырех команд MS DOS:

- 1) C: ; 2) **MKDIR \ MathCAD**; 3) **CHDIR \ MathCAD**;
- 4) **COPYA:*.***

сначала (1) активным делается устройство с именем 'C', затем создается (2) на томе жесткого диска подкаталог с именем **MathCAD** и делается активным (3) и, наконец, с устройства 'A', содержащего дискету с файлами пакета осуществляется их полное копирование (4) в активный подкаталог на устройстве 'C'. При этом, если файлы пакета находятся на 3,5-дискете или 5,25-дискете двойной плотности

(720 К), то требуется только одна установка носителя в устройство 'А'. В случае же 5,25-дискет одинарной плотности (360 К) при копировании всех файлов пакета требуется поочередная смена в кармане УВВ 'А' системной и дополнительной дискет в произвольном порядке, т. е. операция копирования проводится в два приема. Следует отметить, что для нормальной работы пакета достаточно скопировать только файлы 1—6, указанные в табл. 1, так как остальные не являются обязательными. Однако при наличии жесткого диска достаточного объема можно проводить и полное копирование файлов для обеспечения функциональной полноты. Отметим, что при копировании файлов во избежание ошибок при выполнении пакета не рекомендуется их переименовывать.

В случае отсутствия у ПК жесткого диска возможны следующие варианты. Если ПК имеет НГМД для 5,25-дискет (на 720 К или 1,2 М) или 3,5-дискет (на 720 К или 1,4 М), то весь пакет располагается на одном томе, который достаточно поместить в активное устройство, скажем, УВВ 'А'. При наличии же у ПК только НГМД для 5,25-дискет (на 360 К) в УВВ 'В' помещается системная дискета, содержащая только файл-ядро MCAD.EXE, а в активное устройство 'А' — дополнительная дискета, содержащая остальные необходимые файлы пакета, включая оверлейный файл MCAD.OVL. При работе с пакетом MathCAD следует предусматривать создание архивной копии всех его файлов, которая может понадобиться при нарушениях файловой структуры рабочей версии пакета.

Если пакет занимает только один том в НГМД, то, поместив его в активное устройство, запустить пакет можно по команде «MCAD» MS DOS. Если же файлы пакета находятся на жестком диске УВВ 'С' в подкаталоге MathCAD, то, поместив в системный файл AUTOEXEC.BAT две команды

SET MATHCADDIR=C:\MATHCAD и PATH C:\MATHCAD 11

можно определить путь к файлам пакета и запустить его также по команде «MCAD» MS DOS. Если файл AUTOEXEC.BAT уже содержит команду «PATH», то в ней просто дозаписывается путь к подкаталогу MATHCAD. В случае же размещения пакета на двух 5,25-дискетах в активное устройство 'А' устанавливается дополнительная дискета, а в устройство 'В' — системная дискета, содержащая файл MCAD.EXE. В этом случае запуск пакета на выполнение осуществляется по команде 'B : MCAD' и после загрузки файла MCAD.EXE устройство 'В' может быть использовано для работы с другими дискетами, так как при нормальном функционировании пакета обращений к файлу MCAD.EXE не требуется.

В процессе загрузки и функционирования пакет постоянно или эпизодически обращается к своим основным файлам, поэтому к ним должен быть обеспечен беспрепятственный доступ. К файлу MCAD.OVL обращение происходит в течение всего функционирования пакета и его запуск невозможен при отсутствии этого файла. При этом пользователю предоставляется возможность указать путь к файлу

MCAD.OVL, активное УВВ, главный каталог тома которого содержит данный файл (клавиша 'Enter'), либо выйти из пакета по клавише «.» (точка).

Загрузка пакета продолжается только после указания местоположения MCAD.OVL в ответ на запрос ядра пакета. Файл конфигурации пакета MCAD.MCC (если он найден) обрабатывается в самом начале загрузки. К MCF-файлам пакет обращается как в момент загрузки (для экрана монитора), так и при необходимости вывода информации на печать. К файлам MCAD.MCP и MCAD.HLP пакет обращается только при необходимости вывода на печать и запроса главного меню пакета (по клавише 'F1'). Именно поэтому желательно определить расположение всех перечисленных файлов в активных УВВ или каталогах тома. При этом алгоритмы поиска файла MCAD.OVL и других основных файлов отличаются и состоят в следующем. Для файла MCAD.OVL поиск осуществляется:

- в активном УВВ и каталоге (подкаталоге) тома;

- в УВВ и каталоге, содержащем файл MCAD.EXE (только для MS DOS версий 3.3 и выше);

- в каталогах (подкаталогах), указанных в системной команде PATH системного файла AUTOEXEC.BAT.

Для остальных основных файлов пакета поиск идет в порядке:

- в активном УВВ и каталоге (подкаталоге) тома;

- в каталоге (подкаталоге), определенном системной переменной MATHCADDIR окружения процессора команд. Ее значение можно установить системной командой SET MATHCADDIR = <имя каталога> перед загрузкой пакета. В этом случае информация о пути поиска заносится в окружение процессора команд системы MS DOS;

- в каталоге, содержащем файл MCAD.EXE (только для MS DOS версий 3.3 и выше).

Предложенные выше способы организации файловой структуры пакета MathCAD удовлетворяют перечисленным алгоритмам поиска.

Для ПК, совместимых с IBM PC/XT/AT и имеющих только два НГМД для 5,25-дискет (на 360 К), на базе системы MINIDOS [10] и пакета MathCAD была создана достаточно эффективная система MINIDOS/MathCAD, занимающая две 5,25-дискеты и ориентированная, в первую очередь, на отечественный ПК ИСКРА 1030. При этом на первом томе размещаются основная часть MINIDOS с набором важнейших системных команд и файлы пакета с номерами 2—6 и 7 (табл. 1). При этом в файл AUTOEXEC.BAT включены две системные команды 'PATH В:' и 'В: МСAD', позволяющие автоматически загружать файл MCAD.EXE, а в файл MCAD.MCC конфигурации пакета — параметры его настройки на наиболее типичные условия применения. Вторым том содержит файл MCAD.EXE, компилятор с языка БЕЙСИК и ряд сервисных средств. При такой организации на активное устройство 'А' устанавливается том 1, а на устройство 'В' — том 2 и включается ПК. Происходит автоматическая загрузка системы с УВВ 'А', а после получения ответа на запрос даты и времени — автоматическая загрузка пакета MathCAD с выходом на первоначальную заставку пакета и ожиданием действий пользователя. Данная

программная система MINIDOS/MathCAD обладает следующими положительными чертами:

1) повышенная эффективность функционирования пакета на ПК с ограниченными ресурсами;

2) простота использования в среде пакета команд MS DOS.

Опыт работы с системой MINIDOS/MathCAD подтвердил ее хорошие эксплуатационные характеристики при работе на ПК, имеющих память объемом {512 К | 640 К}, два НГМД для 5,25-дискет (на 360 К) и не имеющих жесткого диска и сопроцессора.

В заключении раздела следует отметить, что при наличии у ПК памяти объемом не менее 1 М для существенного повышения реактивности пакета MathCAD рекомендуется перед его использованием создать виртуальный диск (ВД) [10] размером 190 К, сделать его активным и скопировать на него файлы MCAD.OVL, MCAD.MCP, MCAD.HLP, MCAD.MCC, а также необходимые для принтера типа PostScript (если он имеется) MCF- и PS-файлы. Обсудив файловый состав пакета, требуемые для него ресурсы ПК и компоновку пакета для конкретных условий эксплуатации, перейдем к вопросам загрузки пакета на выполнение и изучения возможностей настройки его на требуемый режим функционирования.

1.3. ЗАГРУЗКА ПАКЕТА НА ВЫПОЛНЕНИЕ И ПАРАМЕТРЫ ЗАГРУЗКИ

В предыдущем разделе были рассмотрены вопросы файловой организации пакета и модификации файла AUTOEXEC.BAT MS DOS, которые позволяют загружать пакет на выполнение по команде MCAD. Если в команде загрузки ничего больше не указано, то происходит загрузка ядра пакета, управляющие его таблицы заполняются значениями по умолчанию и делается попытка распознать тип монитора ПК. Если он опознан, то его тип высвечивается на экране и настройка пакета продолжается. При распознавании совместимых типов мониторов пакет может встретить ряд затруднений, что хорошо видно по реакции экрана. В этом случае следует нажатием на клавиши 'Ctrl + Q' прекратить выполнение пакета и запустить его заново, указав требуемый параметр из табл. 2. Явно для этой и ряда других целей служат так называемые *параметры загрузки*, являющиеся ключевыми и идентифицирующиеся символом «/» (слэш) и буквенно-цифровым набором. В этом случае для запуска пакета требуется ввести команду с набором соответствующих параметров загрузки:

$$\text{MCAD}/P_1/P_2 \dots /P_k$$

где $/P_j$ ($j = \overline{1, k}$) — требуемые параметры загрузки. Рассмотрим каждый из допустимых параметров загрузки подробнее. Для задания типа монитора кодируется параметр загрузки согласно табл. 2. Из таблицы видно, что пакет поддерживает работу со всеми наиболее массовыми зарубежными типами мониторов.

Для мониторов типов EGA и VGA пакет допускает использование параметров загрузки, определяющих цвета переднего плана (символы,

Таблица 2

Параметр	Тип задаваемого монитора
/C	CGA; используется для совместимых с IBM Color / Graphic и Professional Graphics Adapter (PGA) адаптеров, а также мониторов типов EGA и VGA
/EC	Определяет стандартный цветной EGA-монитор и его действие подобно параметру '/C', но число цветов экрана равно восьми
/EH	Определяет цветной модифицированный EGA-монитор
/E	Требует использования EGA-монитора, определяются его цветовые характеристики и объем доступной ему ОП; используются максимально доступные монитору ресурсы; параметр может использоваться и для VGA-мониторов
/EM	Определяет монохромный EGA-монитор
/V	Определяет цветной совместимый с VGA монитор
/VM	Определяет монохромный VGA-монитор
/T	Определяет монохромный монитор Toshiba T3100
/A	Определяет монохромный монитор серии AT&T 6300
/H	Определяет монохромный HGA-монитор
/H	Определяет монохромный графический Hercules -монитор

буквы, знаки, линии и т. д.) и фона экрана. При этом для EGA-монитора фирмы IBM требуется 256 К памяти. Для других типов мониторов данные параметры игнорируются. Для указания цветов переднего плана и фона экрана параметры загрузки имеют соответственно вид '/Fnn' и '/Bnn', где числовое значение 'nn' определяет код одного из 64 цветов, используемых EGA-монитором.

В табл. 3 приведены значения кодов для 6 цветов, тогда как остальные цвета легко получаются комбинацией путем сложения значений их кодов. Например, код 48 соответствует желтому цвету, получаемому путем соединения зеленого и красного цветов. Параметры загрузки '/F02 /B48' устанавливают ярко-зеленый цвет для переднего плана (символы, знаки, линии и т. д.) и желтый цвет для фона экрана. Изменить цвет фона в процессе функционирования пакета можно по клавишам 'Ctrl + D'. Каждое нажатие этой пары клавиш приводит к изменению цвета фона, пока не будет достигнут синий цвет.

При загрузке пакет резервирует для своих нужд всю доступную область ОП, которая включает и *расширяемую память* спецификаций {Intel | Lotus | Microsoft} (если такая имеется). Для указания об использовании пакетом только части расширяемой памяти следует задать параметр загрузки '/Lnnnn', где nnnn — объем используемой памяти в килобайтах. При задании параметра '/LO' расширяемая память пакетом не используется. Использование данного параметра обязательно в том случае, когда пакет MathCAD функционирует в среде некоторых мультипрограммных ОС, ибо в этом случае прикладные программы также используют часть расширяемой памяти ПК.

Пакет позволяет сразу же после своей загрузки начать обработку

Таблица 3

Значение 'nn'	Цвет соответствующей части экрана
01	Ярко-синий
02	Ярко-зеленый
04	Ярко-красный
08	Синий
16	Зеленый
32	Красный

ранее подготовленного документа, для чего в команде загрузки следует указать спецификатор требуемого файла, содержащего данный документ, а именно, MCAD [/M] <СФ>. При этом, если не требуется автоматический режим обработки документа (вычисления выражений, таблиц, решающих блоков или построения графиков), то указывается параметр '/M'. Этот же параметр загрузки используется для отмены автоматического режима, устанавливаемого пакетом по умолчанию. Если расширения имени файла не указано, то по умолчанию подразумевается '.MCD'.

При своей загрузке пакет автоматически отыскивает (если он есть) файл MCAD.MCC, содержащий конфигурацию пакета. Для указания другого командного или конфигурационного файла следует использовать параметр загрузки '/R'; MCAD /R [СФ)]. Например, данный параметр позволяет использовать из ранее подготовленного набора файлов конфигураций пакета нужные для данного конкретного режима настройки. Кодирование параметра без указания спецификатора файла отменяет использование стандартного файла MCAD.MCC. В следующем комплексном примере:

MCAD/V/F02/B48/M PTIP.ES/RCONFIG/LO

команда загрузки пакета посредством указанных параметров определяет использование цветного EGA-монитора, цвета для переднего плана и фона экрана задаются соответственно ярко-зеленый и желтый, для обработки в неавтоматическом режиме загружается документ из файла PTIP.ES, определяется нестандартный файл CONFIG.MCC конфигурации пакета и запрещается использование расширяемой памяти ПК.

1.4. ОБЩИЕ БАЗОВЫЕ ОПЕРАЦИИ В СРЕДЕ ПАКЕТА MathCAD

В разделе рассматриваются базовые операции в среде пакета, позволяющие в любой момент выполнять те или иные функции по обеспечению обработки документов, файлов, получению оперативной информации, изменению глобальных параметров пакета и ряд других, включая завершение работы с пакетом.

После успешной загрузки на экране появляется **начальная заставка пакета**, имеющая вид, показанный на рис. 1.

Верхняя часть заставки (первая строка экрана) состоит из **служебной строки пакета** (ССП), которая высвечивается во время функционирования пакета и содержит основную текущую информацию по работе с ним. При этом левая половина ССП отражает имя файла (если он был загружен), выполняемые команды, сообщения, через нее пользователь ведет диалог с пакетом. Тогда как правая половина ССП содержит номера строки и столбца, в которых установлен курсор, а также режим обработки документов (Auto — *автоматический*, *неавтоматический* идентификатора не имеет). Строки экрана нумеруются сверху вниз в пределах 0—23, а столбцы в строке слева направо в пределах 0—79. Правая половина служебной строки может содержать и другую информацию, отражающую результат выполнения команд пакета. Приве-

денный пример начальной заставки (рис. 1) говорит о том, что при загрузке пакета файл документа не определялся, а режим вычислений определен автоматическим по умолчанию. Курсор установлен в левом верхнем углу экрана (строка 0 и столбец 0).

Нижняя часть заставки кратко информирует о данном пакете и предлагает для более полного знакомства с его функциями использовать функциональную клавишу 'F1'. По этой клавише пользователь получает двухуровневый доступ к Help-информации пакета. На первом уровне пакет выводит на экран перечень всех своих функций и указывает буквенные клавиши, по которым на втором уровне можно получить более детальную информацию об использовании выбранных функций, а в ряде случаев и простые примеры. Там же указывается соответствующая глава из работы [19], где можно найти полную информацию. Из любого уровня можно выйти в основную заставку паке-

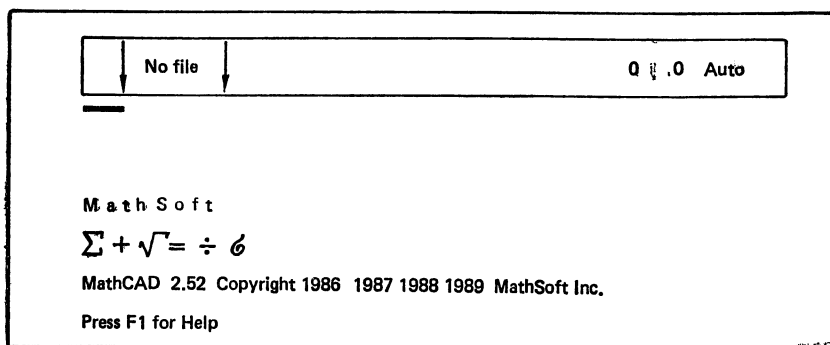


Рис. 1. Начальная заставка пакета MathCAD 2.52

та по клавише 'ESC'. Тогда как со второго на первый уровень возврат осуществляется по клавише 'F1'. В дальнейшем под основной заставкой (или просто заставкой) пакета будем понимать состояние экрана, отражающее состояние служебной строки и, возможно, текущую часть обрабатываемого документа, находящегося в памяти ПК.

Основным объектом, с которым работает пакет, является документ, который содержит не только текст, графические объекты, но и все проводимые вычисления. Общими словами документ можно охарактеризовать как всю ту работу, которую выполняет пакет по требованию пользователя и которая может быть сохранена для последующего использования. Этим он существенно отличается от традиционного понятия документа. В дальнейшем понятие документа будет более детализировано. Пакет дает возможность документировать все необходимые результаты. Для сохранения и последующего использования документ может быть записан в файл, считан из него и выведен на экран и/или печать (принтер, плоттер). В дальнейшем будем, если это не вызывает недоразумений, отождествлять дисковый файл с находящимся в нем документом и наоборот.

Пакет MathCAD имеет развитый командный язык (62 команды), который позволяет весьма эффективно управлять его функционированием и использовать предоставляемые им возможности. Для получения оперативной информации по всем основным командам пакета следует использовать клавишу 'F10'. Вводить команды на выполнение можно комбинациями клавиш, по их именам или через **командное меню** (клавиша 'F10'). Наиболее простым представляется ввод через командное меню. В этом случае по клавише 'F10' в служебной строке появляется перечень доступных групп команд. Используя клавиши управления движением курсора (КУК), можно переходить циклически от группы к группе; в группе, выбрав и высветив нужную команду, нажать клавишу 'Enter'. Выбирать нужную команду можно как посредством КУК, так и вводом первой буквы ее имени. После этого выбранная команда сразу же поступает на выполнение. Недостатком данного способа является невозможность как использования всего набора команд пакета, так и прямого указания параметров ряда команд.

Для ввода команды по имени следует предварительно перевести пакет в **командный** режим клавишей 'ESC', которая переводит пакет из командного режима в режим обработки документа и наоборот. В командном режиме можно выполнять только команды пакета, а в режиме документа — проводить обработку текущего документа. Командный режим определяется в ССП идентификатором 'Command:', предлагающим ввести требуемую команду. Команда вводится либо полным именем, либо сокращенно (прил. 10). Ряд команд пакета можно вводить по клавишам, находясь в состоянии документа. Подробно командный язык пакета будет рассмотрен на протяжении остальных глав книги, здесь же опишем команды общего назначения, с помощью которых без специальных знаний по пакету можно проводить определенного вида работы, не только позволяющие быстрее приобщиться к пакету, но и весьма полезные при дальнейшем его использовании.

Далее будем полагать, что при использовании той или иной команды по ее имени пакет уже переведен клавишей 'ESC' в командный режим.

Так как в общем случае ввод команд пакета возможен на одном из трех уровней (по имени, комбинацией клавиш или через командное меню), то в дальнейшем при описании команд будет использоваться следующая форма их представления:

(Имя команды) [{Параметры}] | (Клавиши) | (B1) (B2)

При этом в первой колонке указывается имя команды (или ее допустимое сокращение), по которому она вводится и распознается пакетом в командном режиме; во второй колонке — клавиши, по которым можно вводить команду в состоянии документа, наконец, в третьей — буквы B1 и B2, определяющие соответственно **подменю** командного меню и саму команду пакета. Если какая-либо из перечисленных возможностей ввода для данной команды отсутствует, то это отмечается прочерком в соответствующей колонке описания. Например, описание

LOAD [⟨CF⟩] | F5 | F L

говорит о том, что команду 'Load' можно вводить в командном режиме по имени 'Load' с параметром или без него, по клавише 'F5' или через командное меню посредством последовательного нажатия клавиш: 'F10'—'F'—'L'. При этом, так как в командном меню команды можно вводить по первой букве их имени (а для некоторых команд она одинакова), то, возможно, придется до двух раз ввести эту букву, чтобы выйти на нужную команду. Если не указано противного, то при кодировании имени команды допускается использование произвольной комбинации прописных и строчных букв.

Все команды пакета условно разбиваются на восемь групп в зависимости от их назначения [19—21]. В данной книге будем придерживаться несколько иной классификации, которая, на наш взгляд, не только более адекватна самой сути команд, но и более естественна для изучения командного языка пакета как основного средства общения с ним. Рассматриваемые в данном разделе команды относятся к группе так называемых **общепакетных** команд и требуются пользователю на самых первых шагах освоения пакета MathCAD. При описании команд пакета будут использоваться как приведенные выше соглашения, так и соглашения по описанию команд MS DOS, принятые в работе [10]. Группа общепакетных команд приведена в табл. 4.

Таблица 4

№	Команда	Основное назначение команды
1	Help (F1)	Справочная информация по пакету
2	Quit	Выход из пакета в систему MS DOS
3	Select	Выбор активного устройства печати
4	ConfigSave	Сохранение текущей конфигурации пакета
5	Execute	Выполнение командного файла пакета
6	Memory	Мониторинг памяти в среде пакета
7	Dos	Вход в среду MS DOS с возвратом
8	Automatic	Режим автоматической обработки документа
9	Manual	Режим неавтоматической обработки
10	Clear	Удаление текущего документа из памяти ПК с перезагрузкой конфигурации пакета
11	Redraw	Восстановление содержимого экрана
12	Reset	Удаление текущего документа пакета
13	Set	Изменение значений пакетных переменных

Рассмотрим подробнее назначение команд данной группы пакета. Команда пакета

HELP | F1 | S H

служит для вывода краткой информации по пакету MathCAD. Команда

QUIT | Ctrl + Q | S Q

предназначена для выхода из пакета и возврата в среду MS DOS. Использовать эту команду можно в любом состоянии пакета. Текущая информация пакета при этом не сохраняется, поэтому перед выходом из пакета следует позаботиться о сохранении нужной информации,

о чем пакет информирует пользователя. При выходе из пакета имеется ряд особенностей, отраженных в прил. 3.

Для указания типа активного принтера или плоттера используется команда

SELECTPRINTER [n] | — | S S

которая выполняется в зависимости от наличия параметра 'n'. Если данный параметр присутствует, то из драйверов файла MCAD.MCP выбирается соответствующий номеру 'n' драйвер. При этом целое положительное число 'n' не должно превышать числа драйверов в файле MCAD.MCP, в противном случае выдается соответствующее диагностическое сообщение. Если же драйвер с указанным номером существует, то он актуализируется и сохраняет свой статус до следующего переназначения принтера. После выбора принтера пакет переходит в режим документа и проверить сделанное назначение можно по команде 'Select' без параметра.

При вводе команды 'Select' без параметра 'n' на экран выводится перечень существующих в файле MCAD.MCP основных драйверов и высвечивается актуальный из них. Посредством клавиш 'PgDn' и 'PgUp' можно быстро просмотреть весь список основных принтеров и плоттеров, а клавишами '↑' и '↓' высветить требуемое устройство печати и нажать клавишу 'Enter'. После этого выбранное устройство становится актуальным и пакет переходит в состояние документа, являющееся его основным состоянием. Поиск файла MCAD.MCP осуществляется сначала в активном каталоге, затем в каталоге, указанном в переменной MATHCADDIR окружения процессора команд MS DOS, и в каталоге с MCAD.EXE на активном устройстве. Если попытка поиска оказалась неудачной, то в ССП появляется сообщение 'Cannot find driver file MCAD.MCP'. Следует отметить, что выбор типа устройства вывода (если он не определен в файле MCAD.MCC конфигурации пакета) определяет не только формат вывода для выбранного устройства, но и формат файла, если документ будет помещаться в дисковый файл (так называемый PRN-файл) с возможностью его последующей распечатки. Работа с командой 'Select' имеет особенность, отраженную в прил. 3.

Для последующего сохранения информации об актуальном устройстве печати, а также других глобальных параметров пакета используется команда

CONFIGSAVE [<ИУ:>] [<Путь>] [<ИФ>] [<РФ>] | — | S C

Если спецификатор файла в команде отсутствует, то информация помещается в стандартный файл MCAD.MCC, обновляя его, если он уже существовал и перезапись санкционирована пользователем. В противном случае информация пишется в файл согласно указанному спецификатору. При этом, если в спецификаторе не указан явно параметр 'РФ', то по умолчанию полагается '.МСС'. Остальные значения подпараметров спецификатора файла соответствуют принципу умолчания, принятому для файловой организации MS DOS и описанному в книге [10]. При записи конфигурации по команде 'ConfigSave'

в уже существующий файл запрашивается подтверждение на его обновление, иначе файл не обновляется. Наряду с этим имеются особенности записи в файл конфигурации, описанные в прил. 3. При последующих загрузках пакета информация об активном устройстве печати и других глобальных характеристиках пакета выбирается из файла MCAD.MCC, либо из файла, указанного в параметре загрузки '/R <СФ>' команды MCAD. Этот случай также имеет особенности, отраженные в прил. 3.

Файл конфигурации пакета, определяемый командой 'Config-Save', является текстовым и имеет следующую структуру:

```
SURFACEFORMAT <Параметры>
SKETCHFORMAT <Параметры>
PLOTFORMAT <Параметры>
FORMAT <Параметры>
SET ORIGIN <Параметры>
SET TOL <Параметр>
MARGIN <Параметр>
LINELENGTH <Параметр>
SELECTPRINTER <Параметр>
PAGELENGTH <Параметр>
DIMENSIONS <Параметры>
```

Из данной структуры следует, что конфигурационный файл содержит 11 команд пакета, определяющих такие важные характеристики, как основные глобальные форматы, значения системных переменных пакета, активное устройство печати и названия единиц измерений. Изменяя значения параметров в командах данного файла и указывая на него в команде загрузки пакета посредством параметра '/R <СФ>' (если файл не определен как стандартный файл конфигурации MCAD.MCC), можно в зависимости от нужд пользователя изменять глобальные характеристики пакета. Ниже о назначении и кодировании команд данного файла речь будет идти подробнее.

Однако изменять глобальные параметры можно не только в момент загрузки пакета, но и в период его функционирования. Для этого достаточно использовать команду пакета

EXECUTE [<СФ>] | — | S E

по которой выполняется определенный параметром 'СФ' командный файл, включая и файл конфигурации пакета. При отсутствии параметра 'СФ' пакет выводит на экран перечень всех имеющихся на активном УВВ MCC-файлов и предлагает указать файл для выполнения, в противном случае выдается сообщение 'файл не найден'. **Командный файл пакета (КФП)** является текстовым файлом в коде ASCII и может содержать любые команды пакета, которые допустимы в командном режиме. Для создания такого файла у пакета нет средств, поэтому следует воспользоваться любым текстовым редактором или командой MS DOS 'Сору'.

Для каждой команды в КФП отводится одна строка длиной до 80 байтов, которая содержит имя команды и необходимые параметры.

В КФП допускается использование комментария, кодируемого как это указано в прил. 3. п. 3, а не согласно фирменной документации [19]. Пустые строки в КФП игнорируются, что совместно с комментарием позволяет более наглядно оформлять файлы такого типа. По умолчанию КФП имеет расширение имени '.MCC'.

Таким образом, при загрузке пакета автоматически либо посредством параметра загрузки '/R <СФ>' читается КФП, определяющий конфигурацию пакета, тогда как в общем случае для выполнения КФП требуется использование команды 'Execute'. Для создания цепочки КФП следует использовать эту же команду 'Execute [<СФ>]', которая должна быть последней в каждом файле цепочки. Все команды после нее в КФП игнорируются. Например, КФП с именем AVZ.MCC вида

Manual

; Перевод пакета в неавтоматический режим

Select 7

Выбор активного устройства печати

Config Save A:PTIP.ES

Сохранение текущей конфигурации пакета

Execute A:PTIP.ES

; Выполнение командного файла пакета

после запуска по команде 'Execute A : AVZ' выполняет действия, описанные в его комментариях. Следует отметить, что в процессе выполнения КФП AVZ.MCC происходят два останова с диагностикой 'Неопределенная команда', на которые следует ответить нажатием клавиши 'Enter' для продолжения выполнения файла. Механизм командных файлов позволяет более эффективно организовывать работу с документами в среде пакета. Ниже будут рассмотрены и более сложные примеры организации и использования КФП.

Для мониторинга используемой памяти в среде пакета применяется команда

MEMORY |—| S M

По этой команде в ССП в общем случае высвечиваются четыре цифры, определяющие в килобайтах соответственно:

- 1) объем ОП, используемый документом пакета (ОП1);
- 2) объем ОП, доступный пакету сразу после загрузки (ОП2);
- 3) объем используемой расширенной памяти при ее наличии;
- 4) общий объем доступной пакету расширенной памяти при ее наличии у ПК.

При отсутствии у ПК расширенной памяти типа {Intel|Lotus |Microsoft} две последние цифры равны нулю. Следует помнить, что после загрузки пакета для собственно документа пользователя отводится часть ОП, остающаяся от ядра MS DOS, ядра самого пакета и резидентных программ, загруженных ранее. Так, например, при работе пакета на ПК с ОП в 512 К под управлением MS DOS 3.3 для документа предоставляется 125 К памяти, тогда как при работе под управлением MS DOS 2.0 уже 142 К.

Когда при работе с документом значение (ОП2—ОП1) приближается к 6 К, пакет может потребовать освобождения памяти ПК. В этом случае происходит следующее:

в правом верхнем углу экрана появляется сообщение «MEMORY»; в служебной строке появляется сообщение «OUT OF MEMORY. Save 'F6' or Quit 'Ctrl + Q'».

Данные сообщения появляются, когда пакет пытается затребовать больше имеющейся в данный момент свободной памяти. Такая ситуация, например, может возникнуть при использовании переменных с многоуровневой индексацией или появлении промежуточных результатов, требующих большой памяти. В данной особой ситуации пользователю предоставляются следующие возможности:

1) для выхода из пакета без сохранения результатов нажать клавиши 'Ctrl + Q';

2) для выхода из пакета с сохранением результатов нажать клавишу 'F6' и указать файл для записи, а затем нажать клавиши 'Ctrl + Q';

3) при работе с двумя окнами экрана и желании сохранить результаты противоположного окна следует нажать клавишу 'F8' для переключения окна, а затем выполнить пункт 2.

При попытке повторной загрузки файла с сохраненной информацией снова возникнет особая ситуация 'Memory', поэтому предварительно следует командой 'Manual' (или при загрузке пакета командой 'MCAD /M<СФ>') перевести пакет в М-режим и только после этого произвести загрузку. Загруженный документ следует отредактировать таким образом (разделить на несколько частей, изменить алгоритм вычислений и т. д.), чтобы вложиться в требования по объему памяти. Во избежание появления данной особой ситуации рекомендуется периодически проводить командой 'Memory' мониторинг памяти ПК.

При наличии у ПК расширенной памяти типа {Intel | Lotus | Microsoft} она также может использоваться пакетом. Однако пакет использует память этого типа только для размещения в ней массивов и графической информации, тогда как все остальные конструкции документа располагаются в ОП ПК. Поэтому ситуация 'Memory' возможна и в случае наличия доступной расширенной памяти при нехватке основной памяти.

Посредством команды пакета

DOS [{<Команда MS DOS> | <СФ>}] | — | S D

пользователь получает возможность выполнить любую корректную команду MS DOS, а также COM- или EXE-файл, заданный его спецификатором. При этом основное ограничение на выполняемый файл обусловлено наличием доступной памяти ПК. Например, интерпретатор языка БЕЙСИК загружен, пользователь может провести нужные ему действия в среде языка БЕЙСИК, тогда как при попытке использовать сервисную программу «Norton Utilities» выдается сообщение 'Exceeded program's memory capability' и пакет MathCAD зависает, требуя перезагрузки системы MS DOS. После успешного выполнения команды MS DOS или программного файла для возврата

в среду пакета следует нажать любую клавишу в ответ на его сообщение.

Для возможности более длительной работы вне среды пакета следует выполнить команду 'Dos' без параметров, а затем нажать клавишу 'Enter'. После этого пакет передает управление MS DOS и пользователь получает возможность проделать необходимую ему работу. Для возврата в среду пакета следует в состоянии MS DOS ввести команду 'Exit' для выхода из процессора команд COMMAND.COM. В следующем простом примере:

```
'ESC'
Command: Dos
DOS>
'Enter'
A > Dir C:
A > Redactor
A > Basic <СФ>
A > Exit
Press any key to return to MathCAD:
```

находясь в среде пакета в состоянии документа и нажав клавишу 'ESC', переводим пакет в командный режим, в котором используем команду пакета 'Dos' для выхода в среду MS DOS. В среде MS DOS по команде 'Dir' получаем состояние главного каталога тома на УВВ 'C', затем текстовым редактором готовим файл исходных данных для БЕЙСИК-программы, запускаем интерпретатор БЕЙСИК и выполняем уже в его среде программу, определенную параметром 'СФ'. После чего по команде 'Exit' получаем возможность вернуться в пакет MathCAD в то его состояние, которое было на момент выхода из него в среду MS DOS. Перед использованием команды 'Dos' следует обеспечить доступ пакета к системному файлу COMMAND.COM, ибо в противном случае выдается сообщение 'Unable to run DOS', говорящее о невозможности перехода в состояние MS DOS. Аналогичная ситуация возникает, если у пакета остается от текущего документа недостаточно памяти, поэтому после данного диагностического сообщения в первую очередь следует провести мониторинг памяти посредством команды 'Memory'. С помощью команды 'Dos' не рекомендуется выполнять резидентные программы, работающие со стеками и некоторыми системными командами, включая команды PRINT и MODE. Использование в команде 'Dos' таких программ может привести к непредсказуемым конфликтам, связанным с управлением памятью. Возможности команды 'Dos' позволяют в оперативном режиме без перезагрузки пакета MathCAD проводить различные работы в средах MS DOS, языков программирования, текстовых редакторов и т. д., используя их результаты для нужд текущего обрабатываемого пакетом документа.

Команды пакета

AUTOMATIC | — | C A
MANUAL | — | C M

являются противоположными друг другу и служат соответственно для перевода пакета в автоматический и неавтоматический режимы

обработки документов. В автоматическом режиме (полагается в пакете по умолчанию) все вычисления выполняются по мере их появления в документе, тогда как в неавтоматическом (М-режим) для этого требуется специальное указание. В дальнейшем эти два режима пакета будем называть соответственно А-режимом и М-режимом. Подробнее об этом будет идти речь в следующей главе.

По команде пакета

CLEAR | — | F C

с санкции пользователя из памяти удаляется текущий документ и происходит перезагрузка конфигурации пакета. При наличии файла конфигурации MCAD.MCC перезагрузка происходит именно из него. Команду 'Clear' рекомендуется использовать перед загрузкой в память нового документа. При этом, если в текущем документе были сделаны какие-либо изменения, то пакет предоставляет возможность сохранить его в дисковом файле перед выполнением команды.

По команде пакета

REDRAW | Ctrl + R | S R

происходит восстановление содержимого экрана в случае искажения отображаемой им информации. Например, наложение одной области экрана на другую или прерывание вывода на экран графической информации по клавишам 'Ctrl + A' или 'Ctrl + Break'. Как правило, причины искажения информации на экране не влияют на проводимые пакетом вычисления и их последствия аннулируются командой 'Redraw'.

В отличие от предыдущей по команде пакета

RESET | — | F R

происходит удаление текущего документа, конфигурация из файла MCAD.MCC не перезагружается и для глобальных характеристик пакета принимаются значения по умолчанию.

Наконец, команда пакета

SET [⟨Переменная⟩ ⟨Значение⟩] | — | —

позволяет изменять значения четырех пакетных переменных: PRNPRECISION, PRNCOLWIDTH, ORIGIN и TOL, назначение которых будет рассмотрено ниже. Если команда 'Set' вводится без параметра, то на экране появляется ее меню с четырьмя указанными переменными и пользователю предоставляется возможность изменить значение любой из них.

На этом завершается рассмотрение команд, характеристик и возможностей пакета, носящих в значительной мере глобальный характер и дающих возможность пользователю представлять основные его возможности, подготавливать пакет к работе и загружать его, а также проводить в его среде ряд общего назначения операций. В следующей главе подробно рассматриваются вычислительные возможности пакета MathCAD.

ОСНОВЫ РАБОТЫ В СРЕДЕ ПАКЕТА MathCAD

В данной главе рассматриваются основы работы в среде пакета, включающие такие моменты как определение переменных и векторов, арифметические вычисления, встроенные и пользовательские функции, использование текстовой информации, построение графиков, системы измерений, а также сохранение созданного документа в файле на диске, вывод его на печать и загрузка в память на обработку. Информация данной главы позволяет быстро ввести пользователя в вычислительную среду пакета и организовать достаточно сложные вычисления с сохранением их результатов во внешней памяти ПК и/или на бумаге в общепринятом математическом виде. Уже на материалах главы можно убедиться в эффективности принципов пакета, делающих его мощным, гибким и легким для использования при обработке различного рода информации вычислительного характера.

Для удобства работы с документом непрофессионального пользователя информация математического характера (выражения, графики, операции и т. п.) появляется в общепринятых обозначениях. Для ввода стандартных математических операций, конструкций и ряда констант используются специальные комбинации клавиш. Тогда как сложные математические конструкции легко строятся из более простых по стандартным правилам, а такие сложные конструкции как интегралы, графики и т. д. вводятся специальными комбинациями клавиш. В процессе подготовки документа пользователь может использовать весь арсенал средств командного языка пакета (для чего в требуемом месте документа следует по клавише 'ESC' выйти в режим команд), командное меню по клавише 'F10' либо для некоторых команд специальные комбинации клавиш. После выполнения требуемых функций пакет возвращается в состояние документа на момент выхода из него. Вся существенная для пользователя информация отображается на экране дисплея, как правило, в служебной строке пакета.

При организации вычислений пакет использует устойчивые к особым ситуациям стандартные и предсказуемые алгоритмы. При этом для вычисления интегралов, производных, обращения матриц и решения уравнений используются наилучшие из известных на сегодня численных алгоритмов. Весь процесс вычислений, включая сопутствующую текстовую и графическую информацию, сохраняется в памяти ПК и адекватно отображается на экране, образуя документ, который может быть сохранен во внешней памяти для последующего использования. Для редактирования документа имеются развитые специальные текстовые и графические средства, о которых речь будет идти в следующих главах книги.

2.1. ОСНОВНЫЕ ПОНЯТИЯ, ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОГЛАШЕНИЯ

Вся область экрана, исключая служебную строку пакета (ССП), отводится под текущий документ, создаваемый или обрабатываемый в данное время. Цифры в правой половине СПП указывают соответ-

ственно номер строки и номер столбца документа, в которых находится курсор. Оба числа могут быть достаточно велики и определяются размером текущего документа и просматриваемой в данный момент его частью. Размер документа ограничивается только величиной доступной памяти ПК, выделяемой ему динамически.

Используемые при работе с пакетом клавиши указываются в кавычках и соответствуют обозначениям клавиатуры, описанной в прил. 4. В кавычках указываются также команды MS DOS, самого пакета и различного рода сообщения, суть которых будет ясна из самого хода изложения. Вводя числа, определения переменных, математические выражения, текст и графики, формируем в среде пакета MathCAD текущий документ. Наряду с фиксацией вводимой информации в документе автоматически или по указанию производятся все необходимые вычисления. При вычислении любого выражения в правой половине ССП высвечивается слово 'WAIT' (ждать), идентифицирующее процесс вычисления, за которым появляется мигающая с периодом порядка 8 с точка для индикации наличия таких вычислений. Наличие указанной информации может охватывать достаточно большой промежуток времени, например, при вычислениях n -мерных интегралов, решающих блоков, графиков, ранжированных выражений и т. д. Время вычисления существенно зависит от тактовой частоты процессора ПК.

После загрузки пакета и выхода на основную заставку можно приступить к созданию документа. В процессе работы над документом пользователь имеет дело с тремя важными понятиями: курсор (■), дополняющий курсор (□) и указатель (■). Понятие стандартного курсора достаточно хорошо известно каждому пользователю ЭВМ, а подробнее о работе с ним можно прочитать в книге [10]. Курсор определяет место в документе, где можно формировать или редактировать нужную конструкцию любого типа. Два других понятия являются относительно новыми, хотя они также указывают место формирования информации в документе.

Дополняющий курсор (□) используется при формировании математических выражений и указывает, что следующий формируемый символ будет размещен сразу же за ним справа. Тогда как указатель (■) определяет место в математическом выражении, куда должна быть помещена соответствующая конструкция (число, выражение, единица измерения), завершающая формирование данного выражения. В сложных конструкциях (например, интегралах) может быть сразу несколько указателей. Переход от указателя к указателю в конструкциях можно осуществлять либо клавишами {→, ←, ↓, ↑}, либо клавишей 'Tab'. Второй способ более быстрый и позволяющий дополняющим курсором просмотреть все указатели выражения. В месте совпадения указателя и дополняющего курсора (■□) на место указателя помещается необходимая конструкция, доопределяющая выражение. Затем описанным способом следует перейти к следующему указателю и т. д. Активным будем называть указатель, на который установлен курсор. Указатель, используемый в конце выражения, предназначен для задания единиц измерений, если в этом имеется необходимость. В дальней-

шем, если это не вызывает недоразумений, дополняющий курсор и указатель будем идентифицировать соответственно как '┐' и '■'. При выводе курсора '┐' из области выражения один из указателей '■' (не определяющий единицу измерения) идентифицируется диагностическим сообщением 'Missing operand', требующим доопределения конструкции. В процессе ввода конструкций или вычисления выражений возможны ошибки, место и причина которых идентифицируются пакетом согласно прил. 5.

При выводе курсора '┐' из области математической конструкции он превращается в обычный курсор и наоборот. Если же в конструкции имелся только указатель для единиц измерений, то при выводе из ее области курсора '┐' удаляется и указатель, тогда как возвращение

X: = 19.42 Y: = 34.675 Z: = -67.98:

X: = 19.42

Y: = 34.675

Z: = -67.98

$$\frac{\sqrt{X + \exp(Y) \cdot \sin(Z)}}{\int \frac{Y \cdot \sin(X)}{x^2} \int \frac{Y}{x} [x^2 + z^2] dz dx} = 0.005$$

$$\frac{\sqrt{X + \exp(Y) \cdot \sin(Z)}}{\int \frac{Y \cdot \sin(X)}{x^2} \int \frac{Y}{x} [x^2 + z^2] dz dx} = 0.005$$

In the left side view of document is presented before "Ctrl + V", in the right side — afterwards

In the left side view of document is presented before "Ctrl + V", in the right side — afterwards

Рис. 2. Визуализация областей документа по клавишам 'Ctrl + V'

в нее курсора '┐' формирует в конце конструкции указатель для единиц измерений. Таким образом, сферой действия курсора '┐' являются только области математических конструкций, включая и графики, на остальной части документа он превращается в обычный курсор.

Позволяя помещать вычислительную конструкцию или текст в любое место документа, пакет, между тем, для каждой такой конструкции (вычислительной, текстовой, графической или иллюстративной) формирует невидимую прямоугольную область, содержащую ее. Для визуализации этих областей достаточно нажать клавиши 'Ctrl + V'; повторное нажатие этих же клавиш отменяет визуализацию областей документа. На рис. 2 изображен простой пример визуализации областей некоторого документа.

При редактировании той или иной конструкции ее область может изменяться в размерах либо вообще исчезать. Формально любой документ можно теперь определить как набор областей двух типов: **вычислительные** и **текстовые**. В свою очередь, вычислительные области подразделяются на области: выражений, плоских графиков, поверхностных графиков и рисунков. Тогда как текстовые подразделяются на *зоны* и просто *области*. Каждая из указанных областей создается и обслуживается в документе своими средствами, о которых подробно будет идти речь в соответствующих главах книги. Область

любого типа можно удалить из документа, для чего достаточно ввести в нее курсор и нажать клавишу 'F3' (удаление) или использовать эквивалентную ей команду пакета 'Cut'. В этом случае пакет помещает удаленную область в свою временную память, для большей надежности сохраняя ее копию на диске. Для возврата области в документ курсор устанавливается в свободную зону документа и нажимается клавиша 'F4' (возврат) или используется эквивалентная ей команда пакета 'Paste'. Следует помнить, что по клавишам 'F3' и 'F4' можно выводить и возвращать только по одной области документа. Если же по 'F3' вывести несколько областей, то по 'F4' можно будет вернуть в документ только последнюю из удаленных областей. Во избежание недоразумений при проведении подобных процедур рекомендуется визуализировать области документа. При попытке вернуть удаленную область на занятое место в документ выдается сообщение 'Would overlap existing region' и возврат не выполняется. Подробнее о проведении подобных процедур речь будет идти при рассмотрении вопросов редактирования документов пакета.

При формировании вычислительных конструкций следует учитывать, что пакет строго соблюдает приоритет выполнения операций согласно общепринятым математическим законам. Если закодирован в конце выражения знак равенства (=), то вычисляется значение этого выражения при работе пакета в режиме автоматических вычислений (о чем свидетельствует слово 'Auto' в ССП). В противном случае в правой части ССП появляются слово 'Calc' и клавиша 'F9', указывающие клавишу для выполнения вычислений. По умолчанию при загрузке пакета принимается автоматический режим вычислений (A-режим), поэтому появляющееся в документе выражение сразу же вычисляется, если оно корректно.

Для простого редактирования существующей конструкции можно воспользоваться КУК (прил. 6), а также клавишами 'Bksp' (удалить символ), 'Del' (удаление) и 'Insert' (вставить символ). Посредством КУК курсор '■' или '┐' подводится к нужному месту в конструкции и клавишами 'Bksp' или 'Del' удаляются элементы конструкции, стоящие слева от курсора. Для вставки символов следует просто ввести их, а пакет автоматически выполнит вставку нужных символов слева от курсора. При использовании клавиши 'Bksp' курсор '┐' заменяется на '└', указывая на то, что вставка символов или элементов будет вестись слева от курсора. Для возврата к курсору '┐' следует еще раз нажать клавишу 'Bksp'.

В результате корректировки конструкции удаляются и связанные с нею результаты вычислений, заменяясь на указатели '■', которые исчезают после вывода '┐' из области конструкции. Подробнее о возможностях редактирования документа пакета речь будет идти ниже. Переходим теперь к определению ряда основных понятий и обозначений, непосредственно связанных с процессом вычислений в среде пакета MathCAD.

В качестве входного алфавита документ использует все символы, допустимые расширенным кодом ASCII, буквы национальных алфавитов, представляемые кодами 128—175 ASCII, а также греческие

Таблица 5

Alt + X	Буква	Alt + X	Буква	Alt + X	Буква	Alt + X	Буква
A	α	G	Γ	N	η	R	ρ
B	β	H	Φ	O	Ω	S	σ
D	δ	I	∞	P	π	T	τ
E	ε	L	λ	Q	θ	U	μ
F	φ					W	ω

буквы, вводимые согласно табл. 5. Для ввода греческой буквы достаточно нажать клавишу 'Alt' и соответствующую латинскую букву (прописную или строчную). Для ввода национального символа достаточно нажать клавишу 'Alt' и ввести соответствующий ему трехзначный десятичный код ASCII на цифровой части клавиатуры. Из знаков, вводимых подобным образом и отсутствующих на некоторых клавиатурах ПК, можно отметить следующие:

'Alt + 124' → ' | '
 'Alt + 174' → ' « '
 'Alt + 158' → ' P_t '
 'Alt + 156' → 'знак фунта'
 'Alt + 175' → ' » '
 'Alt + 171' → ' 1/2 '

При наличии у ПК клавиатуры международного стандарта буквы национальных алфавитов можно вводить простым нажатием клавиш, если в MS DOS имеется соответствующий драйвер клавиатуры.

Используемые в среде пакета **переменные** имеют обычный математический смысл и определяются в вычислительном алгоритме своими уникальными именами. В качестве **имени** можно использовать любой идентификатор, представляющий собой буквенно-цифровой набор длиной не более 63 символов, в котором первой должна быть буква (латинская, греческая или из другого алфавита [коды 128—175]). Специальные знаки в имени не допускаются, кроме знаков подчеркивания (—), процента (%), штриха (') и точки (.), с которых не должно начинаться имя. В отличие от перечисленных знак бесконечности (∞) (клавиши 'Alt + I') может использоваться в имени переменной на любом месте. Использование в имени точки интерпретируется как литерный нижний индекс, суть которого состоит в следующем.

При наличии в имени переменной точки пакет отображает такое имя в виде **основной** его части (до точки) и **нижнего литерного индекса** (часть имени после первой точки). Литерный индекс не говорит о принадлежности переменной какому-либо массиву и тем отличается от обычного индекса переменной. Он просто является частью имени, давая возможность определенным образом идентифицировать переменную из общей группы переменных одинакового смысла. Например, определенные в некоторой конструкции переменные V.initial, V.average и V.final после вывода курсора '┐' из ее области принимают вид V_{initial}, V_{average} и V_{final}, определяя соответственно начальную, среднюю и конечную скорости некоторого физического объекта. При этом допускается и конструкция вида 'a.', к которой также следует

обращаться по 'a.', хотя при выводе курсора из области вычислений она и превращается визуально просто в 'a'. Например, правильным будет следующий фрагмент вычислений в документе: «a.: = 9 sin[a.] + $\sqrt{a. \cdot e^a.} = 24309.664''$ ».

В имени переменной в отличие от имен команд пакета различаются прописные и строчные буквы. В качестве имен переменных не следует использовать имена встроенных и пользовательских функций, которые будут рассмотрены ниже. Приведем примеры допустимых имен переменных любого типа (числовых, векторных, матричных и др.):

∞ ∞ . p. p = $\infty_{p.p}$ $\alpha\beta\delta$ A48VZ a1990
gG'S"Artur % S_____ Ar_____ G REAL % variable

Наряду с обычными переменными пакет допускает использование восьми так называемых **предопределенных** переменных, значения которых устанавливаются по умолчанию, если не указано противного. Предопределенные переменные можно использовать не только в вычислительных конструкциях, но и в качестве **пакетных** переменных, управляющих работой самого пакета. В табл. 6 представлены все предопределенные переменные, их значения по умолчанию и основное назначение.

Таблица 6

Переменная	Значение	Назначение и использование
π	3.1415...	Число π из 15 десятичных цифр
e	2.7182...	Число e из 15 десятичных цифр
∞	10^{307}	Бесконечность представляется максимальным числом в среде пакета; символ введен для удобства и его уточнение дано в прил. 3 п. 5
%	0.01	Используется в выражениях типа X*% или в качестве единицы масштабирования в конце выражений
TOL	10^{-3}	Величина допуска, используемая при приближенных вычислениях
ORIGIN	0	Задаёт значение индекса первого элемента произвольного массива
PRNCOLWIDTH	8	Ширина графы, используемая при записи в файл функцией WRITEPRN
PRNPRECISION	4	Число десятичных знаков, используемых в данных при записи их в файл функцией WRITEPRN

Значения всех (кроме '%') предопределенных переменных можно изменять на двух разных уровнях: *локальном* и *глобальном*. На локальном уровне (в рамках текущего документа) можно изменять путем присвоения переменным новых значений, а посредством команды 'Set' — последние четыре переменные пакета. На глобальном (на все время функционирования пакета) же уровне можно менять значения переменных TOL, ORIGIN, PRNCOLWIDTH и PRNPRECISION.

SION посредством включения в файл MCAD.MCC конфигурации пакета команды 'Set' с соответствующими параметрами, например:

```
SET ORIGIN 1
SET PRNCOLWIDTH 12
SET PRNPRECISION 6
SET TOL 0.00001
```

При описании математических конструкций, если не оговорено противного, будут использоваться следующие обозначения:

X — переменная или выражение любого типа;
Y — действительные переменная или выражение;
Z — комплексные переменная или выражение;
V — векторные переменная или выражение;
M — матричные переменная или выражение;
N — целочисленные переменная или выражение;
F — произвольная математическая функция;
i — диапазон значений произвольной переменной;
t — произвольное имя переменной любого типа;
file — имя файла или переменной, ассоциированной с ним;
var — произвольное имя переменной;
expr — произвольное математическое выражение.

В процессе создания документа и вычислений пакет отслеживает возникающие ошибки и аварийные ситуации и выдает соответствующие диагностические сообщения, полный перечень которых с рекомендациями по их устранению приведен в прил. 5. Однако по мере необходимости некоторые из них уже рассматривались и будут рассмотрены в соответствующих местах книги.

2.2. БАЗОВЫЕ ВЫЧИСЛИТЕЛЬНЫЕ КОНСТРУКЦИИ И ФОРМАТЫ РЕЗУЛЬТАТОВ ВЫЧИСЛЕНИЙ В СРЕДЕ ПАКЕТА MathCAD

В процессе организации вычислений в среде пакета пользователь имеет стандартный доступ к следующим типам чисел: десятичные действительные с плавающей точкой, комплексные, целые 8- и 16-ричные числа. Значения (d) действительных чисел должны находиться в диапазоне $|d| \leq 2.2 \times 10^{307}$, 8-ричных (o) в диапазоне $|o| \leq 1.074 \times 10^9$ и 16-ричных (h) в диапазоне $|h| \leq 2.147 \times 10^9$. Сказанное относится и к диапазонам значений действительных и мнимых частей комплексных чисел. Действительные числа вводятся в обычном формате $[-][\langle \text{целая часть} \rangle][.\langle \text{дробная часть} \rangle]$, тогда как 8-ричные числа имеют формат $[-]\langle \text{число} \rangle\{o|O\}$ и 16-ричные $[-]\langle \text{число} \rangle\{h|H\}$. Знак плюс (+) в представлении чисел любого типа не допускается, так как используется исключительно для обозначения операции сложения. Перед действительным числом X можно кодировать любую цепочку из n знаков (—) минус, присваивая тем самым ему знак, равный $(-1)^n$, а перед числом $|X| < 1$ кодирование нуля необязательно. Например: $.6 = 0.6$ и $-----7 = -0.7$. При этом в 8-ричном числе цифры лежат в диапазоне 0—7, а в 16-ричном — (0—9) & (A—F). Максимальная длина 8-ричного числа составляет 10 цифр,

а 16-ричного — только 8. Следует помнить, что любая конструкция, начинающаяся цифрой, рассматривается пакетом как число или константа.

Для ввода 16-ричного числа, начинающегося буквой, надо перед ним закодировать ведущий ноль. При вводе комплексного числа в качестве мнимой единицы используется буква 'i' (по умолчанию) или 'j', которая может устанавливаться пользователем. Отдельно буквы 'i' и 'j' в качестве обозначения мнимой единицы не используются; их нужно кодировать как 'li' или 'lj'. Хотя на экране после ввода они представляются как 'i' или 'j', например, $\sqrt{-1} = i$.

В обычном представлении можно ввести действительное число, равное 10^{64} , т. е. содержащее 63 цифры '9'. Максимальная длина действительного числа составляет 63 цифры (не считая знака), поэтому указанное выше число можно ввести обычным образом. В среде пакета минимальным положительным полагается число 10^{-15} , а меньше его полагается нулем; для любого выражения X выполняются соотношения $0 * X = X * 0 = 0$, $0/X = 0$ и $0/0 = 0$, при этом само значение X в таких конструкциях не вычисляется, но на наличие ошибок проверяется. Этот момент следует учитывать при организации вычислений. Для ввода чисел d, таких что $|d| \geq 10^{64}$, следует использовать экспоненциальную форму представления, в которой число имеет вид

$$[-]I \langle \text{Действительное число} \rangle \times I10^n \quad (n \leq 307)$$

Приведем примеры правильного кодирования чисел различных типов при их вводе:

Десятичные	8-ричные	16-ричные	Комплексные
19.42	—4567o	7fffffih	li
—2.2*10 ³⁰⁷	1777777o	OABCDEFH	25 + 64i
1234567.83547	7777777o	0A1B2C3D4h	6Ah + 77o*lj
3.8*10 ⁶⁴	—17432o	—23FFF456h	OABh*li
10 ⁶⁴ (—15)	72436o	0AB194290H	19.47*10 ⁶⁴ —5*lj
—19.47*10 ⁶⁴ (—3)	—1234567o	—1986ffah	—1942H—0FFh*li

В вычислениях разрешается произвольное сочетание всех типов чисел, допускаемых пакетом, но результат вычислений всегда будет того типа, который определен форматом результата (D — действительный с плавающей точкой, O — восьмеричный и H — шестнадцатеричный). По умолчанию в пакете определен D-формат результатов вычислений. Приведем примеры вычислений на сочетание различных типов чисел:

$$\begin{aligned} 0fff4H + 567o &= 6.59 \cdot 10^4: 19.42*10^6 6.4*(5ABh + 123o) = 7.483 \cdot 10^{10} \\ &—48.9*(5Ah — 43o)*(12o — OFH*li) = —2,69 \cdot 10^4 + 4.034 \cdot 10^4i \end{aligned}$$

Результаты вычисления данных выражений приведены в D-формате.

Пакет MathCAD позволяет представлять результаты вычислений в заданном виде, определяемом форматом результата. При этом имеется возможность задавать формат как для всего текущего документа

(глобальный), так и для отдельного вычисления (локальный). Для изменения формата результата следует использовать команду пакета

FORMAT | — | C F

После ввода команды 'Format' на экран выводится меню **глобального формата результата** (ГФР), содержащее его характеристики с их значениями по умолчанию и действия:

- | | |
|----------------------------|-------------------------|
| 1. Radix = d | 6. Complex Tolerance=10 |
| 2. Precision Displayed=3 | 7. Overall Default |
| 3. Exponential Threshold=3 | 8. Revert |
| 4. Imaginary Symbol=i | 9. Done |
| 5. Zero Tolerance=15 | |

Пользователь получает возможность изменить значения **характеристик 1—6** и затребовать **действия 7—9**. Для изменения значения требуемой характеристики формата посредством КУК или нажатия клавиши по первой букве характеристики она высвечивается, затем нажимается клавиша 'Enter', после чего можно изменить значение высвеченной характеристики и, нажав снова 'Enter', зафиксировать новое значение. Аналогично изменяются значения и других характеристик формата. Для отмены сделанных изменений следует высветить 'Revert' с последующим нажатием 'Enter' или нажать клавишу 'ESC'. В результате чего восстанавливается формат результата, использованный до применения команды 'Format'. Для пересчета результатов документа с учетом сделанных изменений в формате результата следует использовать действие 'Done' с последующим нажатием клавиши 'Enter'. В отличие от 'Revert' действие 'Overall Default' восстанавливает значения по умолчанию всех характеристик формата, а не только находящихся в команде 'Format' командного файла MCAD.MCC.

Если в процессе функционирования пакета была изменена команда 'Format' в файле MCAD.MCC (например, текстовым редактором по команде 'Dos Redactor'), то для возможности использования нового глобального формата результата можно применить команду 'Clear', которая не только удаляет текущий документ, но и перезагружает в память файл MCAD.MCC, определяя, в частности, текущий формат результата. Однако, наряду с файлом MCAD.MCC пакет имеет и формат результата по умолчанию, восстанавливаемый в качестве текущего действием 'Overall Default' меню глобального формата. Рассмотрим подробнее характеристики 1—6 меню глобального формата результата.

Характеристика 1 определяет формат представления результатов (D — действительный десятичный с плавающей точкой, O — восьмеричный и H — шестнадцатеричный). По умолчанию пакет использует D-формат, а в случае форматов 'O' и 'H' происходит округление до ближайшего целого соответственно 8- и 16-ричного числа. *Характеристика 2* задается целым числом $0 \leq n \leq 15$ (по умолчанию $n = 3$) и определяет количество выводимых цифр после десятичной точки. Для вывода результаты округляются согласно величине n , но в вычислениях участвуют с максимально допустимой точностью (эквивалентно $n = 15$). *Характеристика 3* задается целым числом $0 \leq$

$\leq p \leq 15$ (по умолчанию $p = 3$) и определяет условие экспоненциальной формы представления числа, а именно: если число M таково, что $10^p \leq |M| \leq 10^{-p}$, то оно выводится в экспоненциальной форме, а не в форме с плавающей точкой. *Характеристика 4* принимает только значения 'i' или 'j' и соответствует обозначению мнимой единицы (по умолчанию полагается 'i'). *Характеристика 5* задается целым числом $0 \leq p \leq 307$ и определяет степень *сравнимости с нулем*, т. е. все значения, меньшие 10^{-p} , выводятся нулевыми (по умолчанию $p = 15$). *Характеристика 6* задается целым числом $0 \leq p \leq 15$ и определяет степень *сравнимости* для комплексных чисел, а именно: если для числа $(A + Bi)$ имеет место $|A/B| < 10^{-p}$ или $|B/A| < 10^{-p}$, то число выводится соответственно как Bi или A (по умолчанию $p = 10$). При этом на внутренние расчеты в документе данная характеристика не влияет.

Таким образом, действуя на весь текущий документ и не меняя сути самих внутренних вычислений в документе, глобальный формат результата позволяет выводить окончательные результаты вычислений в требуемом для пользователя виде. Следует отметить, что в системе MINIDOS/MathCAD значения характеристик 1—6 формата результата соответственно равны: 'D', 5, 5, 'i', 20, 15.

При этих предположениях приведем пример вычислений (A) и изменение их результатов (B) для значений характеристик 1—6 соответственно 0; 3; 3; j; 2; 2:

$1942.1234567 = 1942.12346$ $10^4 = 10000$ $0.00048 = 0.00048$ $1 + 1990 \cdot i = 1 + 1990i$	$= 3626o$ $= 23420o$ $= 0$ $= 3706oj$
(A)	(B)

Таким образом, формат результата пакета влияет только на результаты вычислений документа, стоящие после знака равенства.

Наряду с глобальным пакет дает возможность использовать **локальный формат результата (ЛФР)**, позволяющий изменять представление отдельно выбранного результата в документе. Для этого необходимо в искомый результат (на любую из его цифр) установить курсор и нажать клавишу '{f | F}'. На экране появляется меню локального формата с его характеристиками и действиями, подобно рассмотренному выше ГФР. Но относительно глобального формата в локальном отсутствуют характеристики 4 и 5, а действие 7 (Global Default) позволяет для локального формата восстанавливать значения характеристик из глобального формата. После изменения результата согласно заданному для него локальному формату он помечается спереди знаком умножения (\cdot), который на печать не выводится. Приводимый ниже пример иллюстрирует применение локального формата (Н-формата) ко второму результату вычислений.

$19.42869 = 19.42869$ $19 + 48 \cdot i = 19 + 48i$	$= 19.42869$ $= .13h + 30hi$
Глобальный D-формат	Локальный H-формат

В остальном работа с ЛФР полностью аналогична случаю ГФР.

Здесь же уместно отметить еще одну важную унифицирующую черту пакета MathCAD. При дальнейшем изложении мы встретимся и с другими форматами глобального и локального назначения, но их структура полностью унифицирована и подобно ГФР и ЛФР содержит характеристики и действия. И, если набор характеристик определяется назначением формата, то действия для любого глобального формата идентичны по их набору и назначению. Сказанное справедливо и для локальных форматов разных видов. Поэтому, подробно описав действия ГФР и ЛФР, при рассмотрении других видов форматов будем ссылаться на эту информацию, являющуюся инвариантной к типу формата.

Пакет допускает в своей среде не только работу с весьма сложными математическими конструкциями, но и позволяет организовывать работу в режиме обычного электронного калькулятора. Для этого,

Таблица 7

Операция	Ввод	Изображение
Скобки	$X'; (X)$	(X)
Факториал	$N!$	$N!$
Сопряжение числа	Z''	\bar{Z}
Степень числа	$X1^{\wedge}X2$	$X1^{X2}$
Отрицание	$-X$	$-X$
Квадратный корень	$X\backslash$	\sqrt{X}
Модуль числа	$X/$	$ X $
Деление числа	$X1/X2$	$\frac{X1}{X2}$
Произведение	$X1 \times X2$	$X1 \cdot X2$
Сложение чисел	$X1 + X2$	$X1 + X2$
Вычитание чисел	$X1 - X2$	$X1 - X2$

кроме рассмотренных типов чисел, мы должны располагать некоторым набором операций над ними. В табл. 7 представлены в порядке убывания приоритета базовые операции пакета, способ их кодирования и изображения при выводе.

Следует иметь в виду, что в среде пакета имеют место следующие тождества $0! = 0! = 1! = 1$. Уже с помощью данных базовых операций можно формировать в общепринятом математическом виде и вычислять довольно сложные выражения, где под термином «выражение»

в дальнейшем будем понимать любую математически допустимую конструкцию, состоящую из чисел, переменных и функций, объединенных знаками математических операций.

Если пакет находится в автоматическом режиме вычислений (по умолчанию или по команде 'Auto'), то кодирование сразу же после выражения знака равенства (=) приводит к вычислению его результата и помещению его за знаком равенства. При формировании выражения кодирование любой из операций (табл. 7) без операндов порождает указатели, на места которых необходимо поместить требуемые операнды (числа или выражения). Приведем примеры вычислительных конструкций и результаты их вычислений:

Кодирование	Изображение	Результат
$19.42 \times 10^3 / (5 + 7 \times 1i)^2$	$19.42 \cdot \frac{10^3}{(5+7i)^2}$	$-85.113 - 248.247i$
$(3 + 4.5 \times 1i) / ((1942 \backslash + 5))$	$\frac{ 3+4.5i }{\sqrt{1942+5}}$	0.11

$$\begin{array}{lll}
91\sqrt{45}+90\times 1i+(12.3-6\times 1i) & \sqrt{\frac{91}{45+90i}}+|12.3-6i| & 64.769-31.572i \\
(34-6\times 1i)\times (12+1i)''/6i & (34-6i)\cdot \frac{12+i}{6i} & 0.575-0.053i \\
(34-6\times 1i)\times (12+1i)'''/6i & (34-6i)\cdot \left[\frac{12+i}{6i}\right] & 0.558-0.147i \\
\pi\times e^{-(5+1i)} & \pi\cdot e^{5+i} & 251.918+392.339i
\end{array}$$

Из приведенных примеров видно, что при кодировании выражений следует проявлять внимательность при использовании знаков операций и учитывать их приоритет, чтобы получить требуемое выражение. Читателю для приобретения достаточного навыка рекомендуется практически поработать с различного типа выражениями в среде пакета.

Однако, наряду с возможностью организации подобного типа вычислений пакет MathCAD, в первую очередь, ориентирован на решение сложных задач вычислительного характера, алгоритм которых описывается в терминах современной вычислительной математики. Для использования реальных возможностей пакета вводятся *переменные*, определением которых мы можем установить логические связи между математическими конструкциями и использовать промежуточные и окончательные результаты в других конструкциях. Для определения переменной следует ввести ее уникальное имя, нажать клавишу двоеточия (:), по которой помещается знак определения (:=), и после него закодировать число или более сложную конструкцию из чисел и предварительно определенных переменных. Например:

$$X:=1942 \quad Y1:=X+\sqrt{X^3+1}\cdot i \quad Z42:=2\cdot \pi\cdot X \quad AVZ:=Y1^{Z42}+X^i$$

$$PTIP_{Gos}:=\frac{|Y1|^{\overline{Y1}}\cdot Z42}{AVZ^2+X\cdot Y1} \quad PTIP_{Gos}=-5.881\cdot 10^{-12}-6.088\cdot 10^{-12}i$$

В приведенном примере определяются пять переменных и одно вычисление. В пакете принят естественный порядок вычислений слева направо и сверху вниз, поэтому операция присвоения (:=) носит в определенной мере локальный характер и ее область действия распространяется только на нижележащие конструкции, использующие ее. Наряду с локальным определением переменных в пакете можно использовать так называемое **глобальное определение** (\equiv), символ которого вводится по клавише (\sim). Пакет допускает глобальное определение любой конструкции, заданной локально. При этом в среде пакета MathCAD локально (:=) и глобально (\equiv) определяемы следующие конструкции:

- имя переменной, возможно, с верхним и/или нижним индексом;
- имя функции пользователя со списком аргументов в скобках;
- массивы (генерируемые клавишами 'Alt + M'), которые могут содержать в качестве элементов имена переменных с индексами обоих типов.

Любую другую определяемую конструкцию пакет рассматривает недопустимой и выдает соответствующее сообщение (прил. 5). Пакет MathCAD использует описания согласно следующим правилам:

при первом просмотре документа слева направо и сверху вниз вычисляются только глобальные определения;

при втором проходе документа используются глобальные определения из первого прохода и затем другие встречающиеся определения.

Глобальное определение переменной или функции имеет силу на протяжении всего документа независимо от его местоположения. Однако приоритет глобального определения ниже приоритета самого последнего определения (даже локального), а именно: если некоторая переменная X определена дважды сначала как локальная и затем как глобальная, то в своей области действия содержащие ее выражения используют локальное значение X , а выше ее области действия — глобальное. Следующий простой пример иллюстрирует сказанное:

$X:=1$	$2 \cdot Y=44$	$Y:=10$	$5 \cdot Y=50$	$5 \cdot X=5$	$Y+5=15$	$X+Y=11$	$X=42$
$X=48$			$Y=22$		$X+Y=70$		

Для области документа, лежащей за глобальной переменной, она совпадает по смыслу с локальной переменной. Глобальные определения используются для ввода в документ глобальных для него переменных, функций, систем измерений и других конструкций. При использовании глобальных определений следует иметь в виду, что для связанных с ними других ранее введенных конструкций пакет может потребовать их глобального определения. Подробнее об этом речь будет идти ниже.

Пакет позволяет снабжать математические конструкции и вычисления различного рода пояснениями текстового характера. Для этого в любом свободном месте документа устанавливается курсор и нажимается клавиша кавычек (") , по которой генерируются двойные кавычки ("—"), ограничивающие текст, курсор устанавливается на второй паре кавычек. В процессе ввода требуемого текста используется обычный курсор и происходит автоматическая раздвижка ограничивающих кавычек, например:

"Формула числа структур, обладающих неконструируемостью"

Длина строки текста (по умолчанию) равна 80 символам и при превышении ее происходит перенос текста на следующую строку со знаком переноса (-) (дефис). Для перехода на следующую строку достаточно нажать клавишу 'Enter', а для завершения формирования текста следует вывести курсор из области текста, что ведет к удалению ограничивающих текст кавычек и символов перевода строки (↵). Для резервирования места под текст достаточно по клавише 'Enter' ввести необходимое количество пустых строк. На печать ограничивающие символы и символы конца строки не выводятся, хотя и отображаются на экране. Подробнее о работе с текстовой информацией в среде пакета речь будет идти в гл. 8.

2.3. РАЗМЕРНОСТЬ, ЕДИНИЦЫ ИЗМЕРЕНИЙ И ЕДИНИЦЫ МАСШТАБИРОВАНИЯ

Наряду с числовыми безразмерными величинами пакет MathCAD дает возможность использовать единицы измерений и единицы масштабирования, позволяющие решать задачи из областей физики, химии,

техники, экономики и целого ряда других. Единицы измерений помогают идентифицировать ошибки и увеличивают наглядность вычисляемых результатов. Возможности пакета по работе с единицами измерений автоматически поддерживают многие из обычных преобразований, имеющих место при использовании систем измерений в научных и технических расчетах. Указывая пакету определяющие соотношения выбранной системы измерений, получаем возможность автоматических вычислений числовых размерных выражений, а также идентификации некорректных и несовместимых размерных величин. Рассмотрим вопросы использования в среде пакета единиц измерений, включая преобразования единиц, проверку размерности и изменение названий единиц.

По умолчанию пакет MathCAD предусматривает четыре компоненты размерности M, L, T и Q, представляющие соответственно массу, длину, время и заряд. Эти компоненты дополнительного определения не требуют и их физический смысл достаточно ясен. Переменные, определяемые в терминах компонент размерности, называются *единицами измерений*, с которыми можно производить соответствующие вычисления.

Для возможности использования системы измерений необходимо определить систему единиц, с которыми будут ассоциироваться вычисления. Рассмотрим в качестве примера международную систему мер СИ, основанную на базовых единицах измерений: метр (m), килограмм (kg), секунда (sec) и кулон (coul). В терминах компонент размерности пакета базовые единицы глобально определяются следующим образом:

$$m \sim 1L \quad kg \sim 1M \quad sec \sim 1T \quad coul \sim 1Q$$

На основе базовых определяются необходимые производные от них единицы измерений, например:

$$\begin{aligned} sm &\sim 0.01 \cdot m & km &\sim 1000 \cdot m & mm &\sim 0.001 \cdot m & gm &\sim 10^{-3} \cdot kg \\ \text{tonne} &\sim 1000 \cdot kg & \text{min} &\sim 60 \cdot sec & \text{hr} &\sim 3600 \cdot sec & \text{day} &\sim 24 \cdot hr \\ v &\sim \frac{m}{sec} & a &\sim \frac{m}{sec^2} & g &\sim 9.80665 \cdot \frac{m}{sec^2} \end{aligned}$$

Как видно, операции над единицами измерений производятся аналогично случаю обычных переменных. Пакет MathCAD стандартно включает три файла систем измерений USCUSTOM.MCD, MKS.MCD и CGS.MCD, назначение которых приведено в прил. 2. Пользователь может загружать их поочередно в документ по команде 'Load' пакета и просматривать их содержимое. Наибольший интерес для отечественного пользователя может представить файл MKS.MCD, содержащий международную систему мер СИ. Однако пользователь имеет возможность создавать и собственные файлы систем мер с учетом конкретных классов вычислительных задач.

Включать файл с системой мер в документ можно следующими основными способами:

при создании нового документа в память, очищенную по команде

'Clear', загрузить командой 'Load' файл с искомой системой мер и продолжить формирование документа;

командой 'Append' дописать в конец текущего непустого документа файл с искомой системой мер.

Команды 'Load' и 'Append' пакета по работе с файлами будут рассмотрены ниже. Команда же 'Load' достаточно проста и легко используется без предварительного ознакомления. Так как переменные в подобного рода файлах определяются глобально, то местоположение их в документе не играет роли.

Пример работы с системой мер СИ. По команде 'Load' загружается файл MKS.MCD и в обычных физических терминах описывается решение следующей задачи: Определить энергию E колебания груза, подвешенного к пружине, при условии, что груз в начальный момент оттянут на $A = 8$ см из положения равновесия и затем предоставлен самому себе; относительно пружины известно, что она вытягивается под влиянием силы $F = 2$ кГ на $L = 1$ см. Фрагмент документа, решающего данную задачу, имеет следующий вид:

[H] Так как в начальный момент груз не имеет скорости, то амплитуда его колебаний равна $A := 8 \cdot \text{cm}$

Сила растяжения и расстояние соответственно равны

$$kG := g \cdot 10^3 \cdot \text{gm} \quad F := 2 \cdot kG \quad L := 1 \cdot \text{cm}$$

коэффициент упругости K и полная энергия E определяются как

$$K := \frac{F}{L} \quad E := \frac{K \cdot A^2}{2} \quad \text{Откуда:}$$

$E = 6.276 \cdot 10^7 \cdot \text{mass} \cdot \text{length}^2 \cdot \text{time}^{-2}$ ■ Помещая теперь вместо указателя единицы измерений 'joule' и 'erg', соответственно получаем результаты в этих единицах:

$$E = 6.276 \cdot \text{joule} \quad E = 6.276 \cdot 10^7 \cdot \text{erg}$$

Система измерений: $\text{cm} \equiv 1L$ $\text{gm} \equiv 1M$ $\text{sec} \equiv 1T$

$$g = 980.665 \cdot \frac{\text{cm}}{\text{sec}^2} \quad \text{dyne} = \text{gm} \cdot \frac{\text{cm}}{\text{sec}^2} \quad \text{erg} \equiv \text{dyne} \cdot \text{cm}$$

[K]

$$\text{joule} \equiv 10^7 \cdot \text{erg}$$

При использовании той или иной системы мер пакет проверяет выражения на соответствие размерности и проводит все необходимые преобразования единиц измерений и вычисления. По мере возникновения ошибок, связанных с размерностью, как правило, сообщения бывают следующих трех видов:

некорректное преобразование единиц;

переменная с неверной единицей;

использование переменных с несовместимыми размерностями.

Для выявления ошибки рекомендуется проверять алгоритм расчетов с использованием выбранной системы мер.

По мере вычисления результатов пакет указывает и их размерность в терминах базовых единиц $1L$, $1M$, $1T$ и $1Q$ (по умолчанию соответственно 'length', 'mass', 'time' и 'charge'). Это хорошо видно на предыдущем примере. При необходимости получить результат в конкретных единицах измерений необходимо вместо последнего указателя в окончательном результате указать нужную единицу, например,

в предыдущем примере указан 'joule' (джоуль). После ввода допустимой единицы или их комбинации пакет производит необходимые преобразования и отображает результат в искомым единицах. Если комбинация единиц выражения допустима, то из результата исчезают базовые единицы измерений (mass, length, time, charge). Для перехода к новым единицам измерений достаточно в конечном результате удалить размерность и вместо ее указателя ввести новые единицы.

Для изменения наименований размерности используется команда пакета

DIMENSION | — | C D

По команде высвечиваются (подобно команде 'Format') 4 характеристики и действия *меню наименований размерности*. Действия по названию и своему назначению полностью соответствуют случаю ГФР. Характеристики позволяют изменять наименования базовых единиц размерности пакета M, L, T и Q. Например, для базовой единицы Q можно определить имя 'МТОС', введя вместо наименования 'charge' новое имя 'МТОС'.

Наряду с размерностью пакет допускает использование безразмерных единиц вычисления (например, π , e, % и т. д.), в которых будут выводиться результаты. Такой прием называется **масштабированием**. Для этого достаточно определить некоторую переменную и поместить ее имя вместо указателя размерности. Например, основной результат вычисления одного и того же выражения

$$\begin{aligned} \text{avz} &:= 19.42 \quad \text{rag} := 1 \quad \text{dd} := \frac{\pi}{180} \cdot \text{rad} \\ 19.42 \cdot [\sqrt{34} - 2.8^3] &= -313.071 \blacksquare \\ &= -99.654 \cdot \pi \\ &= -115.172 \cdot e \\ &= -3.131 \cdot 10^4 \cdot \% \\ &= -16.121 \cdot \text{avz} \\ &= -1.794 \cdot 10^4 \cdot \text{dg} \end{aligned}$$

представлен в предопределенных переменных пакета π , e и % и в определенных единицах 'avz' и радианах. Наряду со сказанным, в указатель можно помещать не только единицы измерений и масштабирования, но и произвольные выражения, результаты вычисления которых будут масштабировать исходный результат вычисления. Аппарат размерности и масштабирования позволяет существенно расширить вычислительные возможности пакета и круг его приложений, в первую очередь, в физико-технических расчетах.

2.4. ВСТРОЕННЫЕ И ПОЛЬЗОВАТЕЛЬСКИЕ ФУНКЦИИ ПАКЕТА MathCAD

Для повышения эффективности организации вычислительных алгоритмов и работы с данными пакет располагает богатым набором (в количестве 75) встроенных функций, охватывающих различные разделы современной математики. Встроенную функцию (за определенным исключением функций доступа к данным) подобно переменной можно

использовать в любом выражении документа или в качестве аргумента других функций. **Встроенная функция** пакета кодируется в виде

⟨Имя функции⟩ (⟨Список аргументов⟩)

в котором 'Имя функции' кодируется согласно его написанию и является буквозависимым, а 'Список аргументов' содержит перечень аргументов, разделенных запятой. В качестве аргументов могут использоваться любые допустимые выражения. Число аргументов для каждой встроенной функции фиксировано и должно кодироваться согласно ее описанию. Полный перечень встроенных функций пакета с их кратким описанием приведен в прил. 7. Многие из функций достаточно хорошо известны и особых пояснений не требуют, однако целый ряд функций будет подробно рассмотрен в гл. 6.

При использовании встроенной функции пакет вычисляет значения ее аргументов и возвращает результат вычисления функции. Приведем простые примеры использования встроенных функций пакета:

$$X:=30.059 \quad Y:=10+30i$$

$$\sin(X) \cdot \cos(X) = -0.207 \quad \sqrt{\sinh(Y)} = 79.725 - 68.244i$$

$$\operatorname{Re}(Y)^2 + \operatorname{Im}(Y)^2 = 1000 \quad \operatorname{root}[X^2 + 3 \cdot X + 1, X] = -0.382$$

$$\operatorname{floor}(X) \cdot \operatorname{ceil}[\sqrt{X}] + \Gamma(X) = 1.08 \cdot 10^{31} \quad J1(X) + Y1(X) = -0.046$$

Наряду с встроенными, пакет допускает функции **пользователя**, применение которых полностью идентично случаю **встроенных функций**, но с областью действия только в рамках данного документа. Перед своим использованием **пользовательская функция** должна быть предварительно определена локальным или глобальным способом

⟨Имя функции⟩ (⟨Список аргументов⟩) {:= | ≡} ⟨Выражение⟩

При этом при глобальном определении пользовательской функции все входящие в нее переменные (кроме аргументов) должны быть предварительно определены глобальными либо перед каждым использованием функции они должны быть определены любым способом, в противном случае они идентифицируются пакетом как неопределенные. Глобально определенные пользовательские функции имеют своей областью действия весь документ независимо от их местоположения в нем. В качестве 'Имени функции' может быть использовано любое допустимое имя; 'Список аргументов' содержит имена переменных, разделенных запятыми; 'Выражение' может быть любым допустимым в среде пакета. Более сложные конструкции в списке аргументов при определении функции пользователя не допускаются. После своего определения пользовательская функция используется в документе подобно встроенной функции с **отмеченными отличиями**. Например:

$$F_{avz}(X, Y, Z) \equiv \frac{\sqrt{\cos(X) \cdot \cos(Y)}}{(X+Y)^{\operatorname{Re}(Z)}} + \sinh(Z) \cdot \ln(Z)$$

$$X:=19.9 \quad Z:=13.2+9 \cdot i \quad F_{avz}[X, \sqrt{X}, Z^2] = 5.15 \cdot 10^{40} - 7.177 \cdot 10^{40}i$$

$$F_{avz}[\sin(X), \ln(Z), \operatorname{Re}[Z^2]] = 7.066 \cdot 10^{40}$$

$$F_{avz}[F_{avz}(1, 4, 0), F_{avz}(1, 2, 3), 1.9] = 2.099 + 0.001i$$

В качестве аргументов пользовательской функции обычно используются имена переменных, но можно и имена векторов, матриц или других функций. Переменные, кроме аргументов, в пользовательской функции должны быть предварительно определены. При этом следует иметь в виду, что при вычислении пользовательских функций их переменные, не являющиеся аргументами, принимают значения в точке определения функции. Сказанное иллюстрирует следующий простой пример:

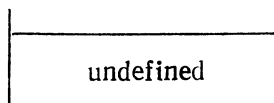
$$G := 90 \quad S(X) := G \cdot X \quad S(1) = 90 \quad S(5) = 450 \quad G := 10 \quad S(1) = 90 \\ S(5) = 450$$

При определении пользовательской функции пакет не вычисляет ее до момента использования в документе. Поэтому ошибки использования такой функции ассоциируются с местом ее применения, хотя причина их может состоять в самом определении функции. Для переопределения встроенной функции пакета достаточно под ее именем ввести новое определение, которое действует только в рамках текущего документа. Наиболее часто используемые функции пользователя можно, глобально определив, поместить в специальный файл, скажем, USERLIB.MCD и при необходимости включать его в начало или конец текущего документа. Для системы MINIDOS/MathCAD пользовательский файл с указанным именем описан в гл. 10 и в прил. 13.

2.5. ВЫЧИСЛЕНИЕ ВЫРАЖЕНИЙ И ДИАГНОСТИКА ОШИБОК

Пакет определяет переменные, вычисляет выражения и решает системы в документе слева направо и сверху вниз, как это имеет место в обычных математических документах. Данный порядок определяет и область действия переменных и определений. Для определения порядка предшествования двух выражений пакет анализирует взаимные расположения верхних левых углов соответствующих им областей. Если оба выражения находятся на одной и той же линии, то выражение слева влияет на выражение справа, но не наоборот. В связи с использованием в среде пакета многоуровневых конструкций принцип порядка предшествования выражений, основанный на взаимном расположении верхнего левого угла их областей, требует особого внимания при определении переменных в документе и местоположения выражений и их составных частей по строкам документа. Например, нахождение на одной строке определения переменных X , Y и двухуровневого выражения

$$X := 3 \quad Y := 10 \quad X^Y = \blacksquare \blacksquare$$



не исключает неопределенности компоненты Y второго уровня выра-

жения, тогда как помещение выражения на строку ниже удовлетворяет принципу предшествования:

$$X: = 3 \quad Y: = 10$$

$$XY = 59049$$

Как уже отмечалось, пакет производит вычисления в А-режиме или в М-режиме, первый из которых используется пакетом по умолчанию. При редактировании существующего выражения вычисления не производятся, они реализуются только после вывода курсора из области выражения, кодирования знака равенства в конце выражения либо нажатия клавиши вычислений 'F9' (в М-режиме). После осуществления любой из этих операций пакет выполняет следующие действия:

производит синтаксический контроль выражения и в случае обнаружения ошибки идентифицирует ее местоположение указателем из прямоугольника, содержащего диагностику;

производит логический контроль выражения (неопределенные переменные, несовместимые единицы измерений, деление на ноль), а также в случае наличия ошибки идентифицирует ее местоположение в выражении и выдает диагностику;

при отсутствии ошибок вычисляет выражение, руководствуясь приоритетом входящих в него операций; при этом для определения пакет вычисляет переменную или функцию слева от знака $\{ : = | \equiv \}$, а при наличии знака равенства — справа от него помещает результат вычисления выражения.

Если выражение является частью определения, то пакет перевычисляет любое другое выражение в документе, которое связано с данным определением. Для перевычисления конкретного выражения необходимо в его область установить курсор и нажать клавишу 'F9' (Calc).

При необходимости прервать процесс вычисления следует использовать клавиши 'Ctrl + Break'. Пакет запрашивает санкцию на прекращение вычисления выражения и по ответу 'Y' прекращает его, помечая выражение сообщением 'Interrupted'. Для повторения прерванного вычисления следует поместить курсор в соответствующую область и нажать клавишу 'F9' (Calc).

При выводе курсора из области определения в А-режиме пакет производит следующие действия: выполняет заданное определение и отмечает все другие выражения в документе, связанные с данным определением, а также актуализирует каждое из связанных выражений по мере их визуализации на экране монитора.

Невидимые связанные выражения документа помечаются как «задержанные вычисления», которые со временем должны быть сделаны. По мере перлюстрации документа на экране все ранее «задержанные выражения» вычисляются, а выводятся уже актуализированные результаты. При выводе на печать или полной перлюстрации документ полностью актуализируется согласно последним сделанным в нем изменениям.

Такая оптимальная организация позволяет всегда видеть только актуальные результаты, но не заставляет ждать пользователя из-за вычисления невидимой для него части документа. Слово 'WAIT' в ССП идентифицирует не только процесс вычисления конкретного выражения, но и процессы печати или перлюстрации документа, когда пакет производит «задержанные вычисления» согласно сделанным предыдущим изменениям.

При желании формировать документ без проведения вычислений следует перевести пакет командой 'Manual' в М-режим, о чем будет говорить отсутствие слова 'Auto' в ССП. В этом режиме формирование документа или его перлюстрация не сопровождаются вычислениями входящих в него выражений. Если в М-режиме формируется или корректируется документ, то в ССП появляется сообщение 'Calc F9', уведомляющее пользователя о том, что для обновления документа следует использовать команду пакета

CALCULATE | F9 | C C

При выводе документа на печать в М-режиме результаты его могут оказаться неактуализированными, поэтому перед его выводом следует произвести актуализацию документа с помощью команды пакета

PROCESS | — | C P

В М-режиме вычисления в документе можно осуществлять одним из следующих способов:

для проведения всех задержанных вычислений, связанных с текущим состоянием экрана, следует использовать клавишу 'F9'; тогда при дальнейшей перлюстрации документа в ССП снова появляется сообщение 'Calc F9', предоставляя возможность провести актуализацию видимой части документа;

для проведения всех задержанных вычислений или актуализации всего документа следует использовать команду 'Process';

для переключения пакета в А-режим вычислений ввести команду 'Auto'; после чего в ССП появляется слово 'Auto', пакет актуализирует весь документ и продолжает его актуализацию автоматически по мере ввода новых изменений и дополнений.

Путем совмещения обоих режимов вычисления пользователь может достаточно гибко совмещать процессы создания или редактирования документа и вычислений в нем.

Как уже отмечалось, при обнаружении пакетом ошибки она локализуется и идентифицируется. Полный перечень идентифицируемых пакетом ошибок, их описание и рекомендации приведены в прил. 5. Пакет не вычисляет выражение, содержащее ошибку, а ошибочное определение игнорирует, что может породить сопутствующие данному определению ошибки. Поэтому причиной локализованной пакетом ошибки может быть некоторое предшествующее ему выражение или определение. Исправление возникающих в документе ошибок зависит от опыта пользователя, его навыков работы с пакетом MathCAD,

а также от специфики самого документа. В качестве общих рекомендаций можно сформулировать следующие:

после обнаружения ошибки перевести пакет в М-режим и попытаться определить и исправить ошибку; это особенно удобно при значительном объеме вычислений в документе, требующих, порой, значительных временных затрат;

причина возникшей ошибки может лежать в определениях (переменных, функций пользователя) и выражениях, связанных с ошибочной конструкцией;

для более точной локализации и идентификации ошибки можно использовать выключение ошибочной конструкции из вычислительного процесса в документе (команда 'Equation').

Для возможности по тем или иным причинам исключать какую-либо вычислительную область или блок областей (выражение, определение, график, решающий блок и т. д.) из процесса вычислений следует воспользоваться командой пакета

EQUATION | — |—

предварительно поместив курсор в область выключаемого объекта. В качестве выключаемого объекта может быть выбрана любая область документа, отличная от текстовой области или зоны. Решающий блок содержит несколько вычислительных областей и о его выключении будет идти речь в гл. 5.

В результате **выключенный** объект помечается справа маленьким квадратиком (\square), идентифицирующим исключение области из процесса вычислений в документе. При наличии в ней ошибок информация о них сохраняется при выводе курсора из области объекта. Например, выключенное определение имеет вид $X := 19.62\square$ и не влияет на все последующие в документе выражения, но может служить причиной ошибочных ситуаций типа 'Undefined'. С выключенными объектами можно проводить различного рода корректировки, не вызывая ошибок в других местах документа. При этом изменения в выключенном объекте совершенно не отражаются на других выражениях документа и не корректируют их, т. е. выключенный объект относительно вычислительного процесса в документе рассматривается пакетом как отсутствующий. Если документ, содержащий выключенную таблицу или график, записать в дисковый файл, то при загрузке такого файла обратно в память в этих объектах сохраняется только та информация, которая была занесена непосредственно пользователем, а не была получена в результате вычислений. Например, если в области графика были определены только аргумент и функция, то в результате вычислений пакет дополнит его диапазонами изменения аргумента и значений функции, а также построит сам график. Если же теперь данный график выключить командой 'Equation' и записать документ на диск, то после повторной загрузки документа в области выключенного графика останутся на осях координат только имена аргумента и функции. Для включения (активизации) выключенного объекта следует поместить курсор в его область и снова ввести команду 'Equation'. Меха-

низм выключения областей оказывается весьма эффективным и в случае подготовки различного рода рукописей в среде пакета, когда требуется без соблюдения требований пакета вводить различные математические конструкции, не предназначенные для участия в вычислительном процессе документа-рукописи.

2.6. РАНЖИРОВАННЫЕ ПЕРЕМЕННЫЕ, ТАБЛИЦЫ И ИТЕРАТИВНЫЕ ВЫЧИСЛЕНИЯ

Наряду с обычными однократными вычислениями выражений пакет MathCAD позволяет осуществлять итеративные вычисления, выбирая для вычисления значений одного и того же выражения различные значения входящих в него переменных. Достигается это использованием ранжированных переменных, таблиц и векторов. **Ранжированная переменная** — это переменная, которой приписан диапазон изменения значений и каждое использование такой переменной воспринимается пакетом как необходимость произвести вычисления по всем ее значениям. В дальнейшем ранжированную переменную будем называть просто *p-переменной*. При наличии в выражении *p-переменных* такие выражения будем называть ранжированными или просто *p-выражениями*. Пакет вычисляет выражение для каждого значения всех входящих в него *p-переменных* и в процессе вычисления *p-выражения* в ССП высвечивается, как правило, слово 'WAIT'. Для прерывания *p-вычисления* следует использовать клавиши 'Ctrl + Break', а для рестарта прерванного вычисления установить в его область курсор и нажать клавишу 'F9'. Результаты таких вычислений можно либо сохранить в массивах, либо представить в виде векторов, таблиц или графиков. В разделе рассматриваются *p-переменные* и вектора, а также использование их для организации итеративных вычислений.

В общем случае при определении *p-переменной* кодируется и представляется соответственно следующим образом:

$$\langle \text{Имя} \rangle \{ : | \sim \} \langle \text{AB1} \rangle [, \langle \text{AB2} \rangle] ; \langle \text{AB3} \rangle$$

$$\langle \text{Имя} \rangle \langle : = | = \rangle (\text{AB1}) [, (\text{AB2})] .. (\text{AB3})$$

За именем *p-переменной* кодируется символ локального или глобального определения, за которым указываются в общем случае три арифметических выражения (AB), значения которых соответственно задают:

(AB1) — начальное значение *p-переменной*;

(AB2) — следующее значение *p-переменной*, определяющее шаг (Ш) ее изменения как $\text{Ш} = (\text{AB2}) - (\text{AB1})$; при отсутствии шага он полагается равным единице;

(AB3) — конечное значение *p-переменной*, которое из-за величины Ш может быть и не достижимым.

Знак шага *p-переменной* определяется знаком разности $(\text{AB3}) - (\text{AB1})$, т. е. $\text{Ш} = [(\text{AB3}) - (\text{AB1})] \times [(\text{AB2}) - (\text{AB1})] / |(\text{AB3}) - (\text{AB1})|$ при наличии в определении *p-переменной* выражения (AB2). В этом же случае для значений указанных выражений должно

соблюдаться условие $(AB1) < (AB2) \leq (AB3)$, иначе возникает ошибочная ситуация. Таким образом, очередное значение p -переменной образуется как измененное на величину Π предыдущее значение (начиная с начального), не выходящее из диапазона $(AB1) - (AB3)$. Количество значений p -переменной будем называть ее рангом. При определении p -переменной также должно соблюдаться ограничение на ее ранг, состоящее в необходимости выполнения соотношения

$$\left| \frac{(AB3) - (AB1)}{[(AB2) - (AB1)]} \right| \leq 10^9,$$

в противном случае возникает аварийная ситуация с диагностикой 'Illegal range'.

Нельзя определять одну p -переменную в терминах какой-либо другой p -переменной. При работе с p -переменной следует помнить, что если ее значения не являются целыми, то такую переменную нельзя использовать в качестве верхнего или нижнего индекса. Приведем простые примеры определения p -переменных и отображения их значений:

[H] a:=2 b:=3 c:=6 x:=a,b..c-1 y:= \sqrt{a} , \sqrt{b} .. \sqrt{c}

$$z:=\sin(a), \sqrt{b} \dots \frac{c}{b} \quad v:=\sqrt{a} \dots b \cdot \frac{5}{c}$$

x	y	z	v
2	1.414	0.909	1.414
3	1.732	1.732	2.414
4	2.05		
5	2.368		

[K]

Определенную таким образом p -переменную в некотором смысле можно назвать **циклической**, исходя из определения циклических конструкций, скажем, в языке программирования БЕЙСИК [10].

Другим способом ранжирования выражений является определение его значений на основе ранжированного индекса. Для этого можно поступить следующим образом. Определяется некоторая p -переменная, скажем, 'K' в качестве нижнего индекса и с помощью операции 'I' (левая квадратная скобка) нижнего индекса определяется вектор значений искомого p -выражения. Например, кодируя

$$K:0; 4 \quad X [K:48 * \sin(K)]$$

получаем эквивалентное ему представление

$$K_i = 0 \dots 4 \quad XK_i = 48 \cdot \sin(K)$$

После такого определения переменная 'X' становится векторной и ее значения (по количеству, равные K) помещаются в одноименный

вектор X. Введя теперь имя вектора со знаком равенства, получаем изображение вектора-столбца значений p -выражения

$$X = \begin{bmatrix} 0 \\ 40.391 \\ 43.646 \\ 6.774 \\ -36.327 \end{bmatrix}$$

В общем виде этот способ ранжирования выражений можно определить следующим образом:

$\langle \text{Имя} \rangle \{ \langle \text{Имя1} \rangle \{ : | \sim \} \langle \text{Выражение} \rangle \langle \text{Имя} \rangle_{\langle \text{Имя1} \rangle} \{ : = | \equiv \} \langle \text{Выражение} \rangle$

где представлены соответственно способы его кодирования и представления. При этом 'Имя' — имя вектора, 'Имя1' — имя p -индекса, а 'Выражение' может быть любым допустимым выражением, включающим и p -переменную 'Имя1'. В случае использования глобального определения все входящие в него переменные должны быть также определены глобально.

Для форматирования значений элементов вектора можно использовать локальный формат результата (ЛФР), для чего следует курсор поместить между скобками элементов вектора и нажать клавишу '{f | F}'. После этого на экране появится меню ЛФР, работа с которым рассмотрена в разделе 2.2. Вектор значений для p -выражения обычно легче определить посредством матричной команды 'Matrix' (клавиши 'Alt + M'). Например, кодирование конструкции

$\langle \text{Имя} \rangle \{ : | \sim \} \text{'Alt + M'}$

предоставляет возможность задать элементы вектора, определяющего некоторую p -переменную.

Еще одним способом ранжирования переменных является использование **входных таблиц**. Входная таблица для переменной определяется следующим образом:

$\langle \text{Имя} \rangle \{ \langle p\text{-переменная} \rangle : \langle \text{Список значений} \rangle$

где 'Список значений' представляет собой перечень значений переменной 'Имя', разделенных запятыми. По данному определению пакет генерирует вертикальную таблицу, заполняемую элементами списка значений сверху вниз по мере их ввода. Например, кодирование конструкции 't := 0..3 t := 2 A [i := 1, 2, 3, 4 B [i := t, t², t³, t⁴' генерирует таблицы вида

$$\begin{array}{|c|} \hline A_i := \\ \hline 1 \\ \hline 2 \\ \hline 3 \\ \hline 4 \\ \hline \end{array} \quad A = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \quad \begin{array}{|c|} \hline B_i := \\ \hline t \\ \hline t^2 \\ \hline t^3 \\ \hline t^4 \\ \hline \end{array} \quad B = \begin{bmatrix} 2 \\ 4 \\ 8 \\ 16 \end{bmatrix}$$


Если ввести теперь 'A=' и 'B=' без индекса, то получим векторные представления входных таблиц. Из данного примера следует, что элементы-выражения во входной таблице не вычисляются, тогда как в соответствующем ей векторе элементы представляют уже результаты таких вычислений. После создания в документе входной таблицы пакет предоставляет следующие возможности по ее корректировке:

увеличить число элементов таблицы, для чего поместить курсор в конец последнего элемента и нажать клавишу запятой (,); пакет добавляет пустой элемент с указателем в таблицу, вместо которого можно поместить новый элемент (число или выражение);

ввести новый элемент внутрь таблицы, для чего поместить курсор в конец какого-либо элемента таблицы и нажать клавишу запятой (,); пакет добавляет пустой элемент с указателем после этого элемента;

заменить или удалить элемент таблицы, для чего поместить курсор в конец нужного элемента и нажать клавишу '{Del | Bksp}' до появления указателя; после этого для замены ввести новое значение элемента, а для его удаления еще раз нажать '{Del | Bksp}'.

При выполнении указанных корректировок входной таблицы в пределах ее p -индекса пакет актуализирует результаты всех связанных с ней последующих вычислений, что не имеет места при увеличении размера таблицы сверх предельного значения ее p -индекса. Все элементы входной таблицы должны либо быть безразмерны, либо иметь одинаковую размерность. Обычно входная таблица имеет один вход для каждого значения из области p -индекса. При наличии слишком большого количества входов для значений индекса лишние игнорируются. Размер входной таблицы ограничен 50 элементами, поэтому при необходимости следует использовать набор связанных между собой входных таблиц. Входные таблицы предоставляют достаточно удобный способ ввода в документ больших массивов исходных для вычислений данных. Приведем простые примеры работы со входными таблицами:

[H]		k:=0..3 sec:=1T	
	Z _k :=	Z _k :=	Z _k :=
	19.42	19.42	19.42
	$\sqrt{19.47}$	$\sqrt{19.47}$	$\sqrt{19.47}$
	19.67	19.67	19.67
			19.89
(1)		(2)	(3)

$$Z^T = (19.42 \ 4.412 \ 19.67 \ 19.89) \quad ZA^T = (19.42 \ 19.47 \ 23.213 \ 1.145) \cdot time$$

[K]

В данном примере определяется p -индекс 'k' и входные таблицы Z и ZA; (1), (4) — исходные входные таблицы; (2), (3) — в таблицу Z в ее конец добавляется новый элемент; в таблице ZA используются размерные элементы; Z^T и ZA^T представляют соответствующие табли-

цам Z и ZA вектора-строки. Таким образом, имеется два способа представления элементов вектора: при вводе имени вектора с нижним индексом и знаком равенства (например, ' $A_k=$ ') выводится таблица его элементов, а при вводе имени без индекса (например, ' $A=$ ') производится вывод элементов в векторных обозначениях, о которых подробнее речь будет идти в гл. 4.

Множество результатов вычисления p -выражения оформляется согласно сказанному в виде **выходной таблицы** или вектора, например:

[H] $x:=1..4$ $Y_x:=\ln(x) + \sqrt{x}$

x		Y_x
1		1
2		2.107
3		2.831
4		3.386

$$Y = \begin{bmatrix} 1 \\ 2.107 \\ 2.831 \\ 3.386 \end{bmatrix}$$

[K]

Данный подход можно применять и для вывода значений массивов (например, матриц), для чего следует использовать p -переменную в качестве нижнего индекса. При выводе на экран выходной таблицы в ее строках могут находиться несущественные пробелы, которые игнорируются при выводе на печать. Для их удаления на экране достаточно нажать клавиши ' $\text{Ctrl} + R$ ' (Redraw Screen), по которым восстанавливается также искаженная в процессе работы с документом информация экрана. Рекомендуется периодически использовать данные клавиши для восстановления содержимого экрана.

Размер выходной таблицы подобно случаю входной не превышает 50 элементов, поэтому для возможности увеличения ранга p -выражения следует использовать несколько p -переменных, входных таблиц или векторов. Например:

[H] $i:=1..50$ $j:=51..100$

i	j
1	51
2	52
...	...
50	100

[K]

Для изменения формата чисел в выходной таблице достаточно использовать ЛФР, о котором говорилось выше. Если в исходном p -выражении использовались единицы размерности, то в его выходной

таблице каждому элементу пакет приписывает соответствующую размерность, одинаковую для всех элементов. Приведем простой пример: [H]

$$\text{sec}:=1\text{T } t:=1\cdot\text{sec}..25\cdot\text{sec } m:=1\text{L } V_0:=101.5\cdot\frac{\text{m}}{\text{sec}} g:=9.8\cdot\frac{\text{m}}{\text{sec}^2}$$

t	$V_0 \cdot t - g \cdot \frac{t^2}{2}$
1.time ¹	96.6.length ¹
2.time ¹	183.4.length ¹
...	...
10.time ¹	525.length ¹
...	...
25.time ¹	-525.length ¹

[K]

Данный пример иллюстрирует решение задачи о движении тела в поле тяжести, брошенного вертикально вверх со скоростью V_0 . Расчеты пройденного телом расстояния даются ежесекундно и позволяют установить максимальную высоту подъема тела (525 м, 10-я с) и время его возвращения на землю (25-я с).

В p -выражениях допускается использование так называемых **условных операций ранжирования** или просто **условных p -операторов**. Условный p -оператор определяется одним из следующих способов:

$$\langle \text{Имя} \rangle_{p\text{-var}} \{ := | \equiv \} \{ \langle p\text{-expr } 1 \rangle | \langle \text{Expr } 1 \rangle \} \langle \text{ОП} \rangle \{ \langle p\text{-expr } 2 \rangle | \langle \text{Expr } 2 \rangle \}$$

$$\langle \text{Имя} \rangle (p\text{-var}) \{ := | \equiv \} \{ \langle p\text{-expr } 1 \rangle | \langle \text{Expr } 1 \rangle \} \langle \text{ОП} \rangle \{ \langle p\text{-expr } 2 \rangle | \langle \text{Expr } 2 \rangle \}$$

где по обе стороны от 'ОП' (оператора) могут использоваться p -выражения (необходимо присутствие по крайней мере с одной стороны) или обычные скалярные выражения. В качестве оператора должны быть указаны конструкции $\{ \approx | < | \leq | > | \geq | \neq \}$. Способ и суть использования таким образом определенного условного p -оператора состоят в том, что p -выражение, соединенное с ним знаком умножения, ранжируется только по тем значениям p -переменной ($p\text{-var}$), на которых p -оператор (по данной p -переменной) принимает значение «истина». При тех же значениях p -переменной, на которых p -оператор принимает значение «ложь», p -выражение не вычисляется и получает нулевые значения.

Вместе с p -выражением можно использовать несколько условных p -операторов, образующих список **p -операторов**. При этом элементы списка по отношению к самому p -выражению и друг к другу могут занимать произвольное положение. Список p -операторов принимает значение «истина» только тогда, когда истинны все входящие в него p -операторы. Следующий фрагмент документа иллюстрирует ска-

[H] $\text{ORIGIN}:=1 \quad j:=1..5 \quad i:=1..3 \quad k:=1..3$

$A_j := (j \approx 2) \cdot j \quad A^T = (0 \ 2 \ 0 \ 0 \ 0)$

$A_j := (j \approx 2) \cdot j \cdot (j \approx 3) \quad A^T = (0 \ 0 \ 0 \ 0 \ 0)$

$A_j := (j \approx 2) \cdot j \cdot (j \geq 2) \quad A^T = (0 \ 2 \ 0 \ 0 \ 0)$

$A_j := (j \geq 2) \cdot j \cdot (j \leq 3) \quad A^T = (0 \ 2 \ 3 \ 0 \ 0)$

$M := \begin{bmatrix} 1 & 4 & -7 \\ 2 & -5 & 8 \\ -3 & 6 & -9 \end{bmatrix}$ Реорганизация матрицы

$N := \begin{bmatrix} 0 & 0 & -3 \\ 0 & -5 & 0 \\ -7 & 0 & -9 \end{bmatrix}$

$N_{k,i} := [i \neq \text{if } [M_{i,k} \geq 0, 1, \text{ORIGIN} - 1]] \cdot M_{i,k}$

$[j^2 \neq 3 \cdot j] \cdot j \quad [j^2 \neq 3 \cdot j] \cdot \left[\frac{j}{2} - \text{ceil} \left[\frac{j}{2} \right] \approx 0 \right] \cdot j$

1
2
0
4
5

0
2
0
4
0

Способы определения условных p -операторов

$A_j := (j \neq 2) \cdot (j \leq 4) \quad B(j) := (j \neq 2) \cdot (j \leq 4)$

$B(j) \cdot j^2$	$A_j \cdot j^2$	$B(j) \cdot A_j$	$B(j)$	$A = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$
1	1	1	1	
0	0	0	0	
9	9	1	1	
16	16	1	1	
0	0	0	0	

Использование условных p -операторов позволяет обойти при вычислениях особые ситуации, вызванные наличием сингулярности:

$M:=10 \quad A:=-1 \quad B:=1 \quad j:=0..M-1 \quad Y(X):=X$

$H := \frac{B-A}{M} \quad F(X) := Y(X)^{-1} \quad X_j := A + H \cdot j$

$Z_j := [j \neq \text{if } [Y[X_j] \approx 0, j, \text{ORIGIN} - 1]] \cdot F[X_j]$

$Z^T = (-1 \ -1.25 \ -1.667 \ -2.5 \ -5 \ 0 \ 5 \ 2.5 \ 1.667 \ 1.25)$

[K] 'if' — встроенная функция пакета MathCAD }

Из приведенного фрагмента нетрудно убедиться, что при определении условного p -оператора по первому или второму способу он представляет собой соответственно массив (вектор или матрицу) или выходную таблицу. Механизм условного ранжирования позволяет существенно расширить возможности пакета по организации ранжиро-

ванных и итеративных вычислительных алгоритмов. В гл. 3 будут приведены и другие интересные примеры использования условного ранжирования. Следует отметить допустимость определения условных операторов и для конструкций непрерывного типа (производные, интегралы), но в этом случае подобные операторы не имеют особого смысла.

Рассмотрим теперь способы организации *итеративных* вычислений. Простейшим типом таких вычислений является естественное обобщение одинарного вычисления некоторого выражения. В этом случае определяются некоторая p -переменная и p -выражение как функция этой переменной. Например: $k := 1..50$ $Y1_k := k^2$ $Y2_k := k^3$ $Y3_k := k^4$ $Y4_k := k^5$. Тут же уместно отметить, что данный тип вычислений весьма удобен в векторном представлении и позволяет отказаться от использования нижних индексов. При этом p -выражения в векторном обозначении, как правило, вычисляются значительно быстрее, о чем подробнее будет идти речь в гл. 4. Другие интересные примеры подобного типа будут рассмотрены ниже.

Пакет наряду с обычной допускает множественную ранжировку выражений. Так, если в p -выражении используется m p -переменных X_j рангов r_j , то пакет вычисляет его для Πr_j ($j = \overline{1, m}$) комбинаций значений всех входящих в него p -переменных. Такой подход, например, весьма удобен для определения матриц. Так для определения (3×3) -матрицы A , элемент (i, j) которой равен $(i + j)$, достаточно ввести определения: « $i:0; 2$ $j:0; 2$ $A[i, j]: i + j$ », отображаемые как « $i := 0..2$ $j := 0..2$ $A_{i, j} := i + j$ ». Вводя теперь имя матрицы со знаком равенства, получаем ее содержимое

$$A = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 2 & 3 \\ 2 & 3 & 4 \end{bmatrix}$$

Следует отметить, что пакет не допускает использования массивов размерности более двух. При использовании m p -переменных рангов r_j в выходной таблице она представляется в виде одноходовой таблицы длины $L = \Pi r_j$; при использовании же этих p -переменных для построения графика пакет выводит одну точку для каждого из L результатов ($j = \overline{1, m}$).

Другой разновидностью обычной итерации является так называемая *рекуррентная* итерация. Суть такой итерации состоит в том, что некоторая переменная вычисляется при некотором начальном значении, а все последующие вычисления определяются как некоторая функция предыдущих ее значений. Будут рассмотрены рекуррентные итерации одной переменной, нескольких переменных и вектора. При определении таких итераций индексные выражения следует вводить в скобках, чтобы сохранить требуемую структуру p -выражения. Первый тип рекуррентной итерации иллюстрирует простой пример порождения некоторой последовательности чисел:

[H]

$A:=3 \quad B:=4 \quad C:=5 \quad i:=1..5 \quad N_i:=i \quad N_{i+1}:=A \cdot N_i^2 - B \cdot N_i + C$

i	N_i
1	1
2	4
3	37
4	3964
5	$4.7124 \cdot 10^3$

[K]

Данный пример иллюстрирует типичную рекуррентную итерацию, в которой каждый элемент числовой последовательности зависит от предыдущего элемента, т. е. $N_{i+1} = F(N_i)$ ($i = 0, 1, 2, \dots$).

Обобщением предыдущего типа является *множественная* рекуррентная итерация, когда значение каждой из совокупности переменных зависит от предыдущих значений всех или части переменных, т. е.

$$x_{i+1}^j = F_j(x_i^1, x_i^2, \dots, x_i^m) \quad (j = \overline{1, m}; i = 0, 1, \dots)$$

Следующий простой пример иллюстрирует сказанное:

[H] $t:=1..5 \quad A_{t+1}:=34.5 \cdot A_t^2 - \sqrt{A_t \cdot B_t} + 1.7 \cdot C_t$
 $B_{t+1}:=0.004 \cdot A_t^2 - 1.2 \cdot C_t^3$
 $C_{t+1}:=7.2 \cdot \sqrt{A_t} + 0.3 \cdot \ln[B_t]$

t	A_t	B_t	C_t
1	1	1	1
2	35.2	-1.196	7.2
3	42741.89	-4.644	$42.771 + 0.942i$
4	$6.303 \cdot 10^{10}$	$7.307 \cdot 10^6$	$1488.277 + 0.942i$
5	$1.37 \cdot 10^{23}$	$1.589 \cdot 10^{19}$	$1.808 \cdot 10^6$

[K]

Рекуррентную итерацию можно осуществлять на векторной основе, начиная с начального вектора и вычисляя на каждом шаге итерации новый вектор. Данный подход использует понятие итеративного вектора и операцию верхнего индекса. Под *итеративным* будем понимать вектор, получающийся из некоторого начального вектора путем конечного числа определенного типа итераций.

Операция верхнего индекса определяется для произвольной матрицы A ($A^{(k)}$) и состоит в выделении ее k -го вектора-столбца. Вводится данная операция по клавишам 'Alt + ()'. Например, кодирование « $A'Alt + ()'k$ » отображается как $A^{(k)}$, где k — должно быть

целым скалярным выражением. Весьма интересными как с теоретической, так и с прикладной точек зрения являются векторные итерации, описываемые следующей матричной моделью:

$V^{(0)}$ — начальный вектор-столбец; A — матрица итерации;

$V^{(k)}$ — вектор k -й итерации, $V^{(k)} = AV^{(k-1)}$ ($k = \overline{1, p}$; $1 < p \leq \infty$).

В частности, данная модель охватывает хорошо известные стохастические дискретные процессы Маркова, имеющие многочисленные приложения в различных областях науки и техники. Приведем пример реализации описанной модели в среде пакета:

$$[H] \quad k:=1..20 \quad V^{(0)}:= \begin{bmatrix} 0.43 \\ 0.48 \\ 0.23 \end{bmatrix} \quad A:= \begin{bmatrix} 1 & 0.3 & 0.2 \\ 0.7 & 1 & -0.5 \\ -0.4 & -0.1 & 1 \end{bmatrix}$$

$$V^{(k)}:=A \cdot V^{(k-1)}$$

$$V^{(1)}= \begin{bmatrix} 0.62 \\ 0.666 \\ 0.01 \end{bmatrix} \quad V^{(10)}= \begin{bmatrix} 15.483 \\ 39.679 \\ -19.565 \end{bmatrix} \quad V^{(20)}= \begin{bmatrix} 1005.939 \\ 2588.623 \\ -1275.548 \end{bmatrix}$$

[K]

Из материалов данного раздела следует, что итеративные вычисления можно описывать в терминах нижних и верхних индексов, p -переменных, условных p -операторов, таблиц, а также векторов и матриц. Подробнее работа с векторами и матрицами будет рассмотрена в гл. 4. Здесь же в общих чертах отметим некоторые различия между этими подходами. Нижние индексы ассоциируются с конкретными элементами массива и при использовании p -переменной в качестве таких индексов пакет осуществляет просмотр всех или части (например, при условном ранжировании) элементов массива.

При отсутствии нижних индексов имя относится к целому массиву (вектору или матрице). Для работы с конкретными элементами массива следует использовать нижние индексы (например, $M_{i,j} := 19.9$), а для большинства матричных или векторных операций рекомендуется использовать массив как единый объект (например, по 'A=' на экран выводится содержимое матрицы A). При этом следует помнить, что представление итеративных вычислений в векторно-матричной форме не только более компактно и наглядно, но и, как правило, требует значительно меньших временных затрат при вычислениях. Однако, не каждое итеративное вычисление представимо в такой форме. Ниже будут более детально рассмотрены вопросы итеративных вычислений в векторно-матричной форме.

При вычислении любого выражения в среде пакета его результат может быть только одного из следующих видов:

единственное значение допустимого пакетом типа;

выходная таблица для p -выражения;

вектор или матрица для p -выражений, векторных и/или матричных выражений.

Таким образом, из трех допустимых видов результатов вычислений мы довольно подробно ознакомились с первыми двумя и в некото-

рой степени с третьим. Подробнее третий тип вычислений будет рассмотрен в гл. 4, тогда как вопросы ранжированных операций суммирования и произведения будут отдельно рассмотрены в следующей главе.

2.7. РЕДАКТИРОВАНИЕ ВЫРАЖЕНИЙ ДОКУМЕНТА

Для редактирования выражений, представляемых в документе в наиболее близком к математическому виде, пакет располагает развитыми средствами. Сам принцип редактирования выражений достаточно прост, имеет много общего с обычными текстовыми редакторами и не требует специальных знаний для его освоения и применения. В разделе будет использован ряд понятий, которые обсуждались в предыдущих разделах главы: число, имя, операция (оператор), указатель, выражение, определение и др. Понятия операнда, бинарного и унарного операторов (операций) достаточно хорошо известны из математической литературы и особых пояснений не требуют. Как при формировании выражения в момент его кодирования (ввода) и вычисления, так и при его редактировании существенную роль играет приоритет составляющих данное выражение операций (операторов). Вопрос приоритета основных поддерживаемых пакетом операций рассмотрен в прил. 8.

Для редактирования нам понадобится понятие **оператора выражения высшего уровня** (или просто **ВУ-оператора**), представляющего собой оператор (или операторы) с максимальным для данного выражения приоритетом. Это связано с тем, что многие вопросы редактирования используют ВУ-оператор в качестве основы всего выражения. Рассмотрим теперь основные принципы редактирования выражений.

Для редактирования существующего выражения необходимо поместить курсор в позицию, где требуется произвести корректировку. После этого для дополнения буквы, числа или оператора достаточно их ввести, а для удаления этих же элементов используются клавиши {Del | Bksp}. В процессе редактирования выражения пакет постоянно интерпретирует все вводимые изменения и сразу же отображает на экране результат редактирования. Важным моментом редактирования является установка курсора в подходящую позицию выражения, зависящую от типа редактирования (вставка, удаление) и самого редактируемого элемента выражения (имя, число, оператор). При проведении редактирования следует руководствоваться следующим правилом:

при редактировании числа или имени курсор устанавливается на любую его букву или цифру;

при редактировании оператора курсор устанавливается на любую его видимую часть.

Следует помнить, что при редактировании операторов возведения в степень, верхнего и нижнего одинарного индекса курсор можно устанавливать как в конец оператора, так и в его левую пустую позицию. Тогда как в случае нижнего двойного индекса курсор следует устанавливать только в левую от него пустую позицию.

Редактируемое выражение может содержать пробелы, однако устанавливать курсор на них не следует, так как пакет выдает сообщение 'Can't edit blank space' и редактирования не происходит. Исключение составляет пробел перед именем или числом, куда можно устанавливать курсор для вставки соответствующих символов. Подробнее остановимся на основных элементах редактирования, иллюстрируя их простыми примерами и пояснениями. В приводимых примерах, как правило, левая часть представляет исходное состояние редактирования, а правая — результат редактирования после вывода курсора из области редактируемого выражения.

Редактирование имен и чисел. В этом случае курсор устанавливается в нужную позицию редактируемого имени или числа и для вставки слева от курсора вводятся требуемые символы, раздвигая выражение; нажатие же клавиш {Del | Bksp} приводит к удалению (по числу нажатий) символов слева от курсора. Простой пример иллюстрирует вставку цифр и удаление букв имени:

$$\begin{array}{ll} Y: = X + _ | 9.42 & Y: = X + 459.42 \\ \text{Variable } _ | 48: = 19.62 & \text{Var } 48: = 19.62 \end{array}$$

Вставка оператора после числа или имени. При вставке бинарного оператора курсор устанавливается в конец имени или числа. Символ оператора вводится согласно правилу его кодирования. В результате этого пакет вставляет нужный оператор, раздвигая выражение, и помещает соответствующий ему указатель, вместо которого для завершения редактирования следует поместить нужное выражение. При необходимости пакет реинтерпретирует отредактированное выражение с учетом приоритета вставленного оператора. Простой пример иллюстрирует сказанное:

$$Y: = X^3 + 2 _ | \cdot X \quad Y: = X^3 + \frac{2}{\blacksquare} \cdot X \quad Y: = X^3 + \frac{2}{\sqrt{X}} \cdot X$$

Вставка оператора перед числом или именем. В отличие от предыдущего случая курсор устанавливается на начало числа или имени и переводится в режим вставки нажатием клавиши 'Ins'. Визуально курсор '┐' заменяется на '└'. После этого вводится требуемый оператор и, если оператор бинарный, то пакет помещает перед ним указатель, вместо которого для завершения редактирования следует поместить нужное выражение. Относительно режима коррективы курсор '┐' называется *дополняющим*, а курсор '└' — *вставляющим*. Вторичное нажатие клавиши 'Ins' возвращает курсор из режима вставки в основной дополняющий режим. Простой пример иллюстрирует сказанное:

$$Y: = X^2 + _ | 5 \quad Y: = X^2 + \blacksquare \cdot _ | 5 \quad Y: = X^2 + \sqrt{_ | X} \cdot 5$$

Вставка оператора после (перед) выражения. В этом случае корректура имеет ряд специфических особенностей. Прежде всего курсор устанавливается на оператор выражения с максимальным приоритетом, а затем вводится новый оператор. При этом пакет вставляет новый оператор в конец выражения, добавляя (при необходимости

сохранения существующего выражения) скобки, а также помещает указатель, связанный с новым оператором. Для завершения корректировки следует вместо указателя поместить нужное выражение. Если курсор переведен до вставки нового оператора в режим вставки (клавиша 'Ins'), то пакет помещает новый оператор перед существующим выражением. Все остальное соответствует уже описанному. Следующие простые примеры иллюстрируют сказанное:

$$\begin{aligned} y: &= x^3 + a \cdot _ x + F(x) & y: &= x^3 + (a \cdot x) \blacksquare _ + F(x) \\ & & y: &= x^3 + (a \cdot x)^4 + F(x) \\ y: &= \frac{(a \cdot x + b)}{(c \cdot x + d)} _ & y: &= \frac{(a \cdot x + b)}{(c \cdot x + d)} \cdot \blacksquare _ & y: &= \frac{a \cdot x + b}{c \cdot x + d} \cdot F(x) \\ y: &= x^3 + a \cdot _ x & y: &= x^3 + \frac{(a \cdot x)}{\blacksquare _} & y: &= x^3 + \frac{a \cdot x}{\sin(x)} \end{aligned}$$

Вставка унарного оператора. В этом случае курсор устанавливается в нужную позицию выражения (на имя, число, имя функции, оператор с максимальным приоритетом), и вводится требуемый унарный оператор, применяемый соответственно к переменной, числу, функции или выражению. Следующие примеры иллюстрируют сказанное:

$$\begin{aligned} y: &= x _ + 5 & y: &= |x| + 5 & y: &= \frac{(a \cdot x + b)}{(c \cdot x + d)} _ & y: &= \sqrt{\frac{a \cdot x + b}{c \cdot x + d}} \\ y: &= F(x) + \sin _ (x) & y: &= F(x) + \sqrt{\sin(x)} \end{aligned}$$

Удаление и замена оператора. В этом случае курсор устанавливается на заменяемый оператор, нажимается клавиша {Del | Bksp} и на месте оператора появляется указатель оператора (\square). Вместо данного указателя можно ввести новый оператор, в результате чего пакет реинтерпретирует выражение по правилам приоритета с учетом нового оператора. Если указатель оператора удалить, то пакет при выводе курсора из области выражения, как правило, идентифицирует ошибочные ситуации 'Missing operator', 'Undefined' и др. Однако возможно проведение данной процедуры и корректно, например:

$$a: = 43 \quad aw: = 48 \quad w: = 23 \quad y: = a \cdot _ w \quad y: = a \cdot w \quad y: = aw \quad y = 48$$

В данном примере удаление оператора не приводит к ошибке, так как в результате просто переопределяется переменная. С помощью замены бинарного оператора на унарный к выражению вида <выражение1><оператор><выражение2> легко можно применять унарные операторы. Следующие простые примеры иллюстрируют сказанное:

$$\begin{aligned} y: &= F(x) + _ \sin(x) & y: &= F(x) \square _ \sin(x) & y: &= \frac{F(x)}{\sin(x)} \\ y: &= F(x) + _ \sin(x) & y: &= F(x) \square _ \sin(x) & y: &= \sqrt{F(x) \square _ \sin(x)} \\ y: &= \sqrt{\sqrt{F(x) \square _ \sin(x)}} & y: &= \sqrt{\sqrt{\sqrt{F(x) \square _ \sin(x)}}} \\ y: &= \sqrt{\sqrt{\sqrt{\sqrt{F(x) + \sin(x)}}}} \\ Z: &= M1 + _ M2 & Z: &= M1 \square _ M2 & Z: &= (M1 \square _ M2)^{(K)} \\ & & Z: &= (M1 + M2)^{(K)} \end{aligned}$$

При удалении унарного оператора курсор устанавливается на удаляемый оператор и нажимается клавиша {Del | Bksp}, например:

$$Y: = _ \sqrt{\sin(x)} \quad Y: = \sin(x)$$

Удаление оператора и одного из его операндов. В этом наиболее часто используемом случае курсор устанавливается в конец удаляемого операнда и нажимается клавиша {Del | Bksp}. В результате чего удаляется требуемый операнд и вместо него появляется указатель. Нажав еще раз клавишу {Del | Bksp}, удаляем указатель и ассоциированный с ним оператор. Следующий простой пример иллюстрирует сказанное:

$$X: = 5 \frac{\sin(X)}{X_} + X^2 = 24.808$$

$$\frac{\sin(X)}{_} + X^2 = \blacksquare \blacksquare \quad \sin(X) + X^2 = 24.041$$

Одним из важнейших способов редактирования является помещение в выражение дополнительных скобок, а также их удаление. Использование скобок в режиме редактирования преследует две основные цели: перегруппировка операций в существующем выражении по новому и сохранение структуры выражения при дополнении его новыми операциями. При вводе выражения использование скобок позволяет изменять стандартные правила приоритета операций в нужном для пользователя виде. Выбранные для перегруппировки части выражения выделяются парой скобок (левой и правой). Однако перегруппировка существующего выражения имеет ряд особенностей.

Вставка новых скобок. В этом случае курсор переводится клавишей 'Ins' в режим вставки и помещается в нужную позицию выражения (начало первого имени, числа или оператора выделяемой группы), куда помещается левая скобка «(» для перегруппировки. Левая скобка помещается пакетом перед вставляющим курсором. После этого новое нажатие клавиши 'Ins' переводит курсор в обычный режим дополнения. Затем курсор устанавливается на конец группируемого выражения и вводится правая скобка «)», в результате чего пакет перегруппировывает выражение согласно введенным скобкам. При попытке ввести правую скобку без соответствующей левой пакет игнорирует заказ и вместо этого переводит курсор на следующий уровень скобок, если таковой имеется. Этот прием может оказаться полезным для организации быстрого перехода с одного уровня скобок на другой в выражении, содержащем большое количество вложенных скобок. Следует отметить, что при группировке элементов выражения, находящихся на нескольких строках документа, пакет вместо круглых сначала использует квадратные скобки (хотя вводятся круглые), которые при выводе курсора из области выражения либо переводятся в круглые, либо (при очевидной группировке) удаляются вообще. При возврате же курсора в область выражения скобки восстанавливаются, позволяя проводить необходимое редактирование. Следующие простые примеры иллюстрируют сказанное:

$$\begin{aligned}
 X: &= 5 \frac{5}{\lfloor X^2 \rfloor} + X^3 - 1 = 124.2 \quad \frac{5}{(X^2)} + X^3 = \blacksquare \blacksquare \\
 &\frac{5}{\lfloor X^2 + X^3 \rfloor} - 1 = \blacksquare \blacksquare \quad \frac{5}{X^2 + X^3} - 1 = -0.967 \\
 \frac{5}{\lfloor X^2 \rfloor} + \frac{1}{X} - 1 &= -0.6 \quad \frac{5}{(X^2)} + \frac{1}{X} - 1 = \blacksquare \blacksquare \quad \frac{5}{\lfloor X^2 + \frac{1}{X} \rfloor} - 1 = \blacksquare \blacksquare \\
 &\frac{5}{X^2 + \frac{1}{X}} - 1 = -0.802
 \end{aligned}$$

Вставку новых скобок можно использовать не только для перегруппировки существующего выражения, но и для сохранения структуры его или его частей при редактировании. Например, выделив посредством скобок в выражении некоторое подвыражение (не меняющее структуры исходного выражения), можно его редактировать, не затрагивая структуру остальной части выражения, и наоборот. Такой подход позволяет производить частичное редактирование выражений. Для быстрого ввода пары скобок можно использовать оператор скобок (клавиша апострофа (')), по которому пара скобок выделяет требуемое выражение. В этом случае вставка пары скобок аналогична вставке унарного оператора и состоит в следующем.

Курсор устанавливается на имя, число, функцию, указатель или оператор выражения с наивысшим приоритетом и нажимается клавиша апострофа ('), в результате чего перечисленные элементы выражения с относящимися к ним компонентами берутся в скобки. Простые примеры иллюстрируют сказанное:

$$\begin{aligned}
 Y: &= F(X) + 5 \cdot \lfloor X + 48 \rfloor & Y: &= F(X) + (5 \cdot X) + 48 \\
 Y: &= \sin(X) + \ln \lfloor (Z) + 43 \rfloor & Y: &= \sin(X) + (\ln(Z)) + 43 \\
 Y: &= \cos(X) + \blacksquare \lfloor \blacksquare \rfloor & Y: &= \cos(X) + (\blacksquare \lfloor \blacksquare \rfloor)
 \end{aligned}$$

Описанный подход можно использовать для расширения отдельных частей существующего выражения. В этом случае расширяемая часть с помощью клавиши апострофа (') берется в скобки, а затем нужным образом расширяется, не нарушая структуру всего выражения в целом. Например:

$$\begin{aligned}
 y: &= \frac{5}{\sin(x)} + \ln(z) & y: &= \frac{5}{(\sin(x) \lfloor \rfloor)} + \ln(z) & y: &= \frac{5}{(\sin(x) + \blacksquare \lfloor \blacksquare \rfloor)} + \ln(z) \\
 & & y: &= \frac{5}{\sin(x) + \cos(x)} + \ln(z)
 \end{aligned}$$

Редактирование и удаление скобочной структуры. После корректного ввода пары скобок пакет рассматривает их как единое целое, поэтому (редактируя выражение в скобках в целом) курсор можно устанавливать на любую из скобок пары. Так, для удаления пары скобок достаточно установить курсор на любую скобку пары и нажать клавишу {Del | Bksp}. Пакет позволяет удалять только связанные пары скобок, реинтерпретируя в случае необходимости выражение.

Аналогично добавляется оператор после пары скобок. Простые примеры иллюстрируют сказанное:

$$\begin{aligned} x: &= 5 \quad 5 \cdot [x^3 + \ln(x)] _ = 633.047 \quad 5 \cdot x^3 + \ln(x) _ = \blacksquare \blacksquare \\ & \quad 5 \cdot x^3 + \ln(x) = 626.609 \\ 5 \cdot _ [x^3 + x^2] &= 750 \quad 5 \cdot (x^3 + x^2) + \blacksquare _ = \blacksquare \blacksquare \quad 5 \cdot (x^3 + x^2) + x = 755 \end{aligned}$$

Команды перекомпоновки. Для перекомпоновки существующего выражения пакет располагает тремя командами

INCOPY | Ctrl + F2 | I C
INCUT | Ctrl + F3 | I C
INPASTE | Ctrl + F4 | I C

сферой действия которых является область выражения. При этом команда 'Incory' позволяет копировать указанный элемент выражения во внешнюю память пакета (ВПП), не изменяя самого выражения; команда 'Incute' — удалять из выражения заданную конструкцию, помещая ее в ВПП и формируя на месте удаления указатель, который можно наполнить новым содержанием или удалить; команда 'Inpaste' — помещать конструкцию из ВПП в указанное свободное место документа или указатель любого его выражения.

Перед выполнением команд 'Incory' или 'Incute' курсор устанавливают на требуемую конструкцию выражения (имя, число, функцию, оператор подвыражения с максимальным приоритетом) и нажимают соответственно клавиши 'Ctrl + F2' или 'Ctrl + F3'. В первом случае указанная конструкция копируется в ВПП без изменения самого выражения (в ССП появляется сообщение 'Expression copied'), а во втором случае конструкция после копирования в ВПП удаляется из выражения и вместо нее устанавливается указатель. Для использования команды 'Inpaste' курсор помещают на свободное место документа между областями или в указатель существующего выражения и нажимают клавиши 'Ctrl + F4', по которым конструкция из ВПП помещается по местоположению курсора. Данную команду можно использовать многократно, копируя одну и ту же конструкцию из ВПП в различные места документа. Простой пример иллюстрирует использование всех перечисленных команд:

$y: = \sin(x) + \cos(x) _ \cdot \ln(x)$	Исходное состояние
$y: = \sin(x) + \blacksquare _ \cdot \ln(x)$	После применения 'Incute'
$y: = \sin(x) + x \cdot \ln(x)$	После заполнения указателя
$z: = \blacksquare _$	После ввода определения
$z: = \cos(x)$	После применения 'Inpaste'
$y: = \sin(x) + x \cdot \ln(x) _$	После применения 'Incory'
$z: = \cos(x) + \blacksquare _$	После ввода оператора (+)
$z: = \cos(x) + \ln(x)$	После применения 'Inpaste'

Для перекомпоновки выражения соответствующую его часть можно брать в скобки и оперировать указанными командами со всем подвыражением, находящимся в скобках, как с целым. Из сказанного следует, что команды перекомпоновки предоставляют достаточно широкие возможности по перегруппировке вычислительной части текущего документа.

Относительно рассмотренных команд необходимо сделать ряд уточнений, не вытекающих непосредственно из фирменной документации [19—22], но имеющих существенное значение. Для команд 'Incopy', 'Incute' и 'Inpaste' (группа команд 'In-region') пакет отводит различные типы ВПП в зависимости от обрабатываемых ими объектов (помеченных текстовых блоков или выражений). Согласно этому будем различать ВПП текста и ВПП выражений. Более того, для подобных команд 'Copy', 'Cut' и 'Paste' пакет выделяет еще один тип ВПП (ВПП области / зоны), являющийся общим для всех команд данной группы, но отличный от ВПП текста и ВПП выражений. При этом содержимое всех трех типов ВПП сохраняется при очистке памяти от текущего документа по команде 'Clear'. Такая организация ВПП позволяет не только весьма гибко организовывать редактирование текущего документа, но может служить основой для организации интерфейса между различными документами. Наконец, следует иметь в виду, что любой вид копирования типа 'In-region' выражения в ВПП с последующим возвращением в документ не сохраняет операции векторизации, если она имелаась в исходном выражении, а при копировании его в ВПП операция не связывалась прямо с копируемым выражением, например, путем взятия векторизованного выражения в скобки. Таким образом, при копированиях типа 'In-region' все области вычислений (выражения, графики, рисунки) используют один и тот же тип ВПП, а текстовые помеченные блоки — другой. Тогда как по команде 'Copy' используется единая ВПП (отличная от ВПП для 'In-region') для всех типов областей и зон документа.

Оператор продолжения выражения. Для возможности использования длинных выражений, а также вставки внутрь их новых конструкций пакет располагает оператором **продолжения**, до некоторой степени аналогичным по действию оператору сложения. Для продолжения существующего выражения курсор устанавливают в его конец и нажимают клавиши 'Ctrl + Enter'. В результате этого в конце выражения появляются три точки (...), а на следующей строке документа знак плюс (+) с указателем, вместо которого можно помещать необходимое продолжение выражения.

Для разбиения выражения курсор устанавливается на нужный знак плюс (и только на него) и нажимается клавиша {Del | Bksp}, по которой знак плюс удаляется и вместо него появляется указатель оператора (□). После этого нажимаются клавиши 'Ctrl + Enter' и выражение разбивается аналогично уже описанному. Если же после установки курсора на знак плюс (+) сразу же нажать клавиши 'Ctrl + Enter', то пакет реализует так называемое **вложенное прерывание**, позволяющее осуществлять прерывание внутри самого выражения. Суть такого приема хорошо иллюстрирует следующий простой пример:

X:=5	Y:= X + X + _ X + X + X	Исходное состояние
	Y:= [X + X + X ...] + X	После нажатия клавиш
	[+ ■]	'Ctrl+Enter'

Из данного примера следует, что пакет прерывает выражение после следующей за отмеченным знаком (+) конструкции и дает возможность с этого места организовать вложенное продолжение. Следующий пример иллюстрирует основные возможности оператора продолжения:

$Y := X + X + X + X + X _$	Исходное состояние
$Y := X + X + X + X + X \dots$ $+ \blacksquare$	После нажатия клавиш 'Ctrl+Enter'
$Y := X + X + X + X + X \dots$ $+ X + X + _ X + X + X$	После продолжения выражения и установки курсора
$Y := X + X + X + X + X \dots$ $+ X + X \square X + X + X$	После нажатия клавиши {Del Bksp}
$Y := X + X + X + X + X \dots$ $+ X + X \dots$ $+ X + _ X + X$	После нажатия клавиш 'Ctrl+Enter' и установки курсора
$Y := X + X + X + X + X \dots$ $+ X + X \dots$ $+ [X + X \dots] + X$ $+ \blacksquare$	После нажатия клавиш 'Ctrl+Enter'
$Y := X + X + X + X + X \dots$ $+ X + X \dots$ $+ [X + X \dots] + X$ $+ [X + _ X + X + X]$	После продолжения выражения внутри его и установки курсора
$Y := X + X + X + X + X \dots$ $+ X + X \dots$ $+ [X + X \dots] + X$ $+ [+ [X + X \dots] + X + X]$ $+ \blacksquare$	После нажатия клавиш 'Ctrl+Enter'

Приведенный пример на использование оператора продолжения выражений достаточно прозрачен и особых пояснений не требует.

Из настоящего раздела следует, что наряду с простыми пакет предоставляет и достаточно развитые средства редактирования выражений, под которыми понимаются любые вычислительные конструкции документа (определения, уравнения, неравенства, числа, переменные, функции и т. д.). Для получения навыка по редактированию выражений читателю рекомендуется практически поработать с пакетом MathCAD на ПК, совместимых с IBM PC/XT/AT.

2.8. СОЗДАНИЕ ДВУХМЕРНЫХ ГРАФИКОВ В СРЕДЕ ПАКЕТА

Продолжая материал раздела 2.6, рассмотрим вопросы создания простых графиков с использованием r -переменных. Пакет позволяет представлять значения r -выражений в виде двухмерных графиков: одна ось отводится под значения r -переменной и другая — под значения r -выражения, являющегося функцией этой переменной. Для вывода графика необходимо произвести следующие действия.

Определяются p -переменная и p -выражение и для создания области графика курсор устанавливается в свободную область документа (находящуюся в сфере действия локального определения p -выражения, либо в любое свободное место документа при глобальном определении данного p -выражения) и нажимается клавиша 'д' (эт). После этого в указанном месте документа появляется пустая графическая область с 6 указателями, имеющая вид, изображенный на рис. 3.

Для удобства рассмотрения указатели на рис.3 пронумерованы, а оси графической области условно отмечены подобно декартовой системе координат (ось X и ось Y). На экране перечисленные элементы отсутствуют, а графическая область содержит только прямоугольную рамку с указателями и курсором, установленным на один из указателей.

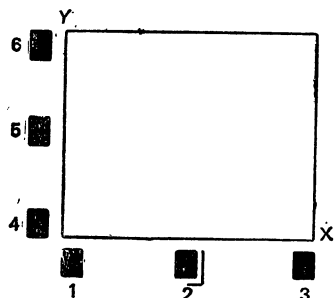


Рис. 3. Структура области двумерного (плоского) графика

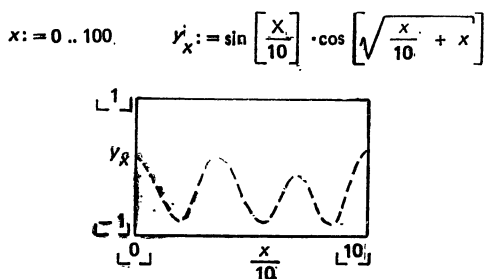


Рис. 4. График функции в стандартном формате пакета MathCAD

После создания по клавише 'д' графической области курсор устанавливается в указатель 2 оси X. В этой позиции, как правило, задается имя p -переменной, а в позициях 1 и 3 соответственно начало и конец диапазона изменения ее значений. По желанию пользователя назначение осей графической области можно менять, например, ось X на ось Y и наоборот. В позиции 5 задается имя с нижним индексом выходной таблицы соответствующего p -выражения (или просто левая часть определения этого выражения), а в позициях 4 и 6 соответственно начало и конец диапазона изменения его значений. При этом, если в какой-либо из позиций 2, 5 помещены имена p -переменной и выходной таблицы, то диапазоны изменения их значений можно не указывать — пакет расставляет их автоматически (помечая каждое такое значение снизу уголками, которые на печать не выводятся) после вывода курсора из графической области. Можно чередовать частичное определение значений диапазонов с их автоматическим доопределением. В качестве диапазонов значений должны быть скалярные действительные предварительно определенные выражения.

После вывода курсора из графической области или нажатия клавиши 'Enter' пакет вычисляет точки искомого графика (идентифицируя этот процесс в ССП сообщением 'WAIT'), выводит их на экран

(соединяя по умолчанию прямыми линиями) и формирует недостающие значения диапазонов на осях графика (позиции 1, 3, 4 и 6). Приведем простой пример создания двухмерного графика (рис. 4).

В данном примере после определения p -переменной x и p -выражения u_x курсор устанавливается ниже области выражения и нажимается клавиша 'ⓐ', иницилирующая на экране графическую область стандартного формата, который устанавливается пакетом по умолчанию. Курсор устанавливается на средний указатель оси X графической области, куда помещается выражение ' $x / 10$ ' для p -переменной. Затем курсор устанавливается в средний указатель оси Y и вместо него вводится имя выходной таблицы p -выражения. После этого нажимается клавиша 'Enter' и пакет начинает производить вычисления, связанные с доопределением области и построением в ней искомого графика. Весь этот процесс сопровождается сообщением 'WAIT' в ССП. После построения графика курсор визуализируется ниже полностью определенной графической области. Задавая различные значения для диапазонов осей графической области, можно выбирать для просмотра искомого графика различные его участки и/или производить его растяжение или сжатие по осям координат. Каждое такое изменение требует от пакета пересчета всех необходимых значений перед выводом нового изображения графика.

В приведенном примере (рис. 4) для вывода графика использовался стандартный формат графической области: ее размер полагался 6 строк в высоту и 15 столбцов документа в ширину при соединении вычисленных точек графика прямыми отрезками. Однако пакет дает возможность изменять графический формат, подобно тому, как это делалось в случае локального формата результата (ЛФР). Для этого достаточно поместить курсор в искомую графическую область и нажать клавишу {f | F}, по которой на экран выводится меню **локального графического формата** (ЛГФ). Работа с ЛГФ полностью аналогична случаю ЛФР, а сам локальный формат содержит следующие характеристики и действия:

1. Size
2. Trace Types
3. Log Cycles
4. Subdivisions
5. Global Default
6. Revert
7. Done

Смысл и назначение действий 5—7 полностью соответствуют случаю ЛФР, а характеристики 1—4 определяют следующие параметры ЛГФ: 1 — размер графической области в строках и столбцах документа; 2 — спецификация способа соединения точек выводимого графика; 3 — число логических циклов соответственно по осям Y и X; 4 — разбиение соответственно осей Y и X. Подробнее о графическом формате в 2 и 3 измерениях речь будет идти в гл. 9. Здесь же наряду с вводом основных понятий проиллюстрируем использование ЛГФ только для увеличения размера графической области, содержащей график предыдущего примера (рис. 4).

Для этого достаточно установить курсор в графическую область и нажать клавишу {f | F}, в результате чего в правой части экрана появляется меню ЛГФ с высвеченной характеристикой 1. Нажав клавишу 'Enter', требуем изменения этой характеристики. Пакет подтверждает возможность ее изменения установкой курсора в конец ее текущего значения и пользователь может ввести новое значение в виде «п, m» (п — высота области в строках и m — ширина ее в столбцах документа; $p, m \leq 127$). После этого следует последовательно нажать клавиши 'Enter' — 'D' — 'Enter', тогда пакет перевычисляет искомый график в новой графической области. Для нашего примера новый вид графика изображен на рис. 5.

Как следует из сказанного, предельный размер графической области (127×127) позволяет выводить двухмерный график целиком на любой принтер. Для перлюстрации же такой области можно посред-

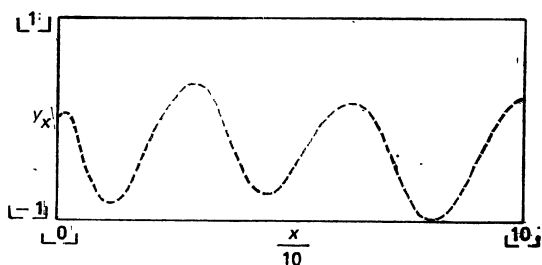


Рис. 5. График функции в измененном локальном формате пакета

ством перемещения курсора обозревать любую видимую часть графика, что в ряде случаев не совсем удобно. Поэтому при первоначальном графическом анализе рекомендуется использовать графическую область, помещающуюся на экране, и только в случае вывода графика на печать или при его сложном поведении можно прибегнуть к увеличению графической области. Не следует опасаться существенного увеличения размера дискового файла, куда помещается документ, содержащий графические области максимальных размеров. Так, например, документ, содержащий только определения p -переменной, p -выражения и графической области предыдущего примера, требует под выгружаемый файл только 531 и 537 байтов соответственно с графической областью стандартного и максимального размеров, т. е. увеличение объема незначительно.

Для проведения необходимых корректировок в графической области можно использовать команды 'Incscr', 'Incscr' и 'Inpaste', рассмотренные выше. С их помощью можно удалить и копировать с сохранением в ВПП выражений элементы графической области, находящиеся на местах ее указателей подобно случаю обычных элементов выражений. Кроме того, указатели графической области по команде 'Inpaste' можно заполнять выражениями из ВПП. Саму кривую графика копировать таким способом нельзя, но ее можно уда-

лать из области по команде 'Incuit' (не нарушая содержимого ВПП), при этом все остальные элементы области остаются без изменения.

Подобно выражениям и определениям графики можно выключать по команде 'Equation', помещая перед этим курсор в соответствующую графическую область. В результате этого она помечается справа квадратиком (□) аналогично случаю выражений. Подобным образом можно поступать и с выходными таблицами. Данные выключенные объекты (графики и таблицы) не участвуют во всех последующих изменениях и вычислениях документа, пока вновь по команде 'Equation' не будут включены. При копировании документа, содержащего выключенные графики и выходные таблицы, с последующей загрузкой его в память такие объекты восстанавливаются не полностью: от выходной таблицы остается только ее имя со знаком равенства, а графическая область сохраняет все свои элементы, кроме самой кривой графика. Отменив командой 'Equation' выключенность данных объектов, можно получить их актуализированный на данный момент вид.

Приведенные в разделе основы работы с двухмерными графиками позволяют легко оформлять в графическом виде результаты вычислений, носящих табличный или векторный характер. Подробнее о работе с графической информацией будет идти речь в гл. 9.

2.9. СОХРАНЕНИЕ ДОКУМЕНТА НА ДИСКЕ И ВЫВОД ЕГО НА УСТРОЙСТВО ПЕЧАТИ

Для последующего использования текущий документ рекомендуется сохранить в дисковом файле, что позволяет сделать команда пакета

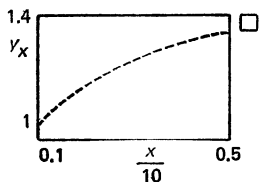
SAVE [⟨СФ⟩] | F6 | F S

x: = 1..5

$$y_x := \cos \left[\frac{x}{10} \right] + \sin \left[\frac{x}{10} \right]$$

Definition of r-variable.

Definition of r-function



Plot of r-function

This example illustrates the saving of MCD-file by command "Save" and its following loading in memory by command "Load".
Disk MCD-file is ASCII-format text file!

Рис. 6. Пример документа, сохраняемого в дисковом MCD-файле

Параметр 'СФ' команды определяет спецификатор файла, куда будет осуществляться выгрузка текущего документа. По умолчанию имя файла с сохраняемым документом получает расширение '.MCD',

поэтому такие файлы для краткости будем называть *MCD-файлами*. Если параметр у команды 'Save' опущен, то пакет запрашивает спецификатор файла по запросу 'Save as:', на который следует ввести необходимый ответ. После выгрузки документа в дисковый файл он сохраняется в памяти, его спецификатор высвечивается в левой половине ССП и с ним можно продолжать работу. При попытке сохранить текущий документ в уже существующем файле пакет запрашивает санкцию на обновление данного файла 'File exists: «Y» to overwrite?'. Ответ «Y» санкционирует обновление, в противном случае команда 'Save' игнорируется и пакет переходит в состояние документа. При выгрузке документа в MCD-файл возможны аварийные и особые ситуации, с которыми можно ознакомиться в прил. 3 и 5.

Дисковый файл, содержащий выгруженный документ, является обычным текстовым ASCII-файлом в MCD-формате, имеющим следующую структуру. Для примера рассмотрим простой документ, представленный на рис. 6, в котором имеются 6 областей и одна текстовая зона.

Данный пример иллюстрирует фрагмент некоторой рукописи, содержащей определения ранжированных переменной и выражения (функции), график функции, текстовые области, содержащие комментарии к перечисленным конструкциям, и часть основного текста рукописи. Выгружая по команде 'Save C : file' данный документ, создаем на устройстве 'C' MCD-файл с именем 'file.mcd'. А так как этот файл является текстовым, то его можно анализировать и редактировать обычным текстовым редактором. Вывод данного файла на принтер имеет следующий вид:

```
.MCD 25000 0 (Стандартный заголовок MCD-файла)
.CMD SURFACEFORMAT rot = 10 tilt = 35 vScale = 20 size = 15, 30
.CMD SKETCHFORMAT mag = 1.0, 1.0 center = 0.50, 0.50 size = 15, 30 box = y
.CMD PLOTFORMAT logs = 0, 0 subdivs = 1, 1 size = 5, 15 type = 1
.CMD FORMAT rd = d ct = 15 im = i et = 5 zt = 20 pr = 5 mass length
time charge
.CMD SET ORIGIN 0
.CMD SET TOL 0.0001000000000000
.CMD MARGIN 0
.CMD LINELENGTH 78
.CMD SET PRNCOLWIDTH 8
.CMD SET PRNPRECISION 4

.EQN 0 0 1 12 (Описания областей и зон документа)
x : 1; 5 ◀
.TXT 0 32 1 27
al, 26, 46, 25
Definition of r-variable
.EQN 2 —32 3 25
y/x : cos (x / 10) + sin (x/10) ◀
.TXT 1 32 1 27
al, 26, 46, 25
Definition of r-function
.EQN 6 —32 9 21
▼&&y {x {1, 1, 6, 15, 1} @&x/10 ◀
.TXT 2 33 1 21
al, 20, 45, 19
```


Plot of r-function
.TXT 8 —33 3 32003
b3, 32002, 78, 165

This example illustrates the saving of MCD-file by command 'Save' and its following loading in memory by command 'Load';
disk MCD-file is ASCII-format text file!

Из приведенной распечатки MCD-файла хорошо видно, что он состоит из стандартного заголовка (11 строк), содержащего, по сути дела, весь файл конфигурации MCAD.MCC пакета, и набора описаний всех областей и зон сохраненного документа. Информация заголовка файла позволяет настроить основные характеристики пакета на работу с документом в той среде, которая была на момент его сохранения в дисковом MCD-файле. Каждое описание содержит его тип [EQN — область выражения, .TXT — текстовая область (a) или зона (b)], координаты описываемой области или зоны и само их содержимое; при этом текст представлен так же, как и в документе на экране, а выражения представляются в том виде, как они вводятся в документ. Для графической области указывается не только ее местоположение, но и текущий формат, а также идентификаторы для осей Y и X. Каждое выражение, т. е. вычисляемая конструкция документа (поэтому и графические области рассматриваются пакетом как EQN-области), завершается символом (\blacktriangleleft), а выключенные выражения слева дополнительно помечаются символом (\blacktriangledown). Все вычисляемые пакетом конструкции дополнительно справа также помечаются знаком вопроса (?), а их значения вычисляются только после загрузки MCD-файла в память. Подобная организация MCD-файла имеет целый ряд преимуществ, из которых отметим следующие.

Текстовый формат MCD-файла позволяет, зная его структуру, производить редактирование посредством любого текстового редактора, не обращаясь к услугам самого пакета. Сам файл при такой организации содержит весьма ограниченный объем избыточной информации, что позволяет эффективно использовать дисковую память, а использование средств сжатия файлов (прил. 12) дает возможность существенно улучшить эту характеристику. Структура MCD-файла (во многом аналогичная представлению документа в памяти ПК) позволяет применять достаточно простые и эффективные алгоритмы восстановления из нее того документа, который визуализируется на экране и выводится на устройства печати. Более того, структура MCD-файла объясняет многие свойства пакета, рассматриваемые в настоящей книге (неполная восстанавливаемость выключенных графических областей и выходных таблиц, необходимость строгого соблюдения взаимного расположения зон и областей в документе, наличие различных типов ВПП, заметные временные задержки при визуализации частей документа и ряд других). Для лучшего усвоения принципиальных вопросов организации пакета читателю рекомендуется самостоятельно поэкспериментировать с MCD-файлами, используя вышесказанное и полученную в процессе работы новую информацию.

В определенной мере обратной к 'Save' является команда пакета

LOAD [CФ)] | F5 | F L

вводимая подобно команде 'Save' с очевидными изменениями. С параметром 'СФ' команду можно ввести только в командном режиме пакета, инициируемом клавишей 'ESC'. В этом случае пользователь сразу же точно идентифицирует загружаемый документ. Если расширением имени файла является '.MCD', то в параметре 'СФ' указывать его не обязательно. При вводе команды 'Load' без параметра пакет просит определить спецификатор загружаемого файла, выдавая запрос 'Load as:'. В зависимости от полученного ответа пакет производит следующие действия:

Ответ

Действия пакета

<СФ>	Загружает MCD-файл согласно его спецификатора
'Enter'	Выводит меню всех MCD-файлов активного УВВ
<УВВ:>	Выводит меню всех MCD-файлов заданного УВВ
[УВВ:] **	Выводит меню файлов заданного каталога тома

После этого (кроме первого вида ответа) требуемый MCD-файл можно ввести через текущее меню файлов, выводимое на экран. При этом в ответ на запрос меню MCD-файлов пакет выводит имена только тех файлов, которые имеют расширение имени '.MCD'. Таким образом, следует, вообще говоря, различать MCD-файлы и на уровне расширений их имен, так как по отношению к пакету они в общем случае неэквивалентны.

Перед вводом команды 'Load' рекомендуется очистить память от текущего документа по команде 'Clear'. Если же команда 'Load' вводится при наличии текущего документа в памяти, то пакет требует подтверждения необходимости удаления текущего файла. Получив положительный ответ 'Y', пакет удаляет текущий файл и загружает новый, указанный в команде 'Load'. В противном случае команда 'Load' игнорируется, а пакет переходит в состояние документа, не изменяя содержимого памяти. Следует отметить, что при загрузке файла по команде 'Load' предыдущее содержимое всех типов ВПП сохраняется, что позволяет передавать некоторые конструкции и результаты между документами.

Важной командой работы с текущим документом является команда пакета

APPEND [<СФ>] | — | F A

Все сказанное о команде 'Load' можно отнести и к команде 'Append', но в отличие от первой она дописывает указанный (либо явно, либо в ответ на запрос пакета) файл в конец текущего документа. При этом команду 'Append' не следует применять к пустому документу. В противном случае пакет в ССП выводит диагностическое сообщение 'No file loaded' и команда игнорируется, а пакет возвращается в состояние документа. Команду 'Append' рекомендуется использовать при последовательном объединении нескольких документов в единый текущий документ либо при необходимости дополнения текущего документа недостающими компонентами (глобально определенные системы единиц измерений, функции пользователя, определения и т. д.).

позволяет вывести текущий документ на печать или в указанный дисковый файл в PRN-формате (в дальнейшем именуемый *PRN-файл*). При этом во избежание негативных последствий от возможных аварийных и особых ситуаций перед выводом документа на печать его рекомендуется по команде 'Save' сохранить в дисковом MCD-файле. Если тип принтера не специфицирован, то в ССП появляется сообщение 'No printer specified', в ответ на которое следует воспользоваться командой 'Selectprinter', позволяющей специфицировать принтер или плоттер. Так как ваш ПК имеет постоянный тип принтера, то его рекомендуется включить в файл конфигурации MCAD. МСС пакета, тогда не потребуется каждый раз определять принтер.

После успешного ввода команды 'Print' пакет запрашивает размер выводимой области документа («Print area 0, 0 to n, m»; по умолчанию указывается весь текущий документ). Ответ клавишей 'Enter' требует вывести весь текущий документ, иначе следует указать размер выводимой области документа в виде «n1, m1 n2, m2», где (n1, m1) и (n2, m2) — координаты в терминах (строка, столбец) соответственно верхнего левого и нижнего правого углов выводимой прямоугольной области текущего документа.

После получения ответа пакет запрашивает выводное устройство или файл 'Printer device or filename: PRN', предполагая по умолчанию основной принтер (PRN). Если принтер приписан порту вывода 'PRN', то в ответ достаточно нажать клавишу 'Enter'. Как правило, этот ответ достаточен для большинства эксплуатирующихся ПК. Если же принтер приписан другому порту вывода, то в ответ следует ввести его логическое имя, например, 'COM1'. При отсутствии устройства или его неисправности команда 'Print', как и все остальные команды, отменяется клавишей 'ESC'. После успешного начала операции печати текущий документ выводится в том виде, как он отображается на экране. Ответив на запрос устройства вывода спецификатором файла, можно поместить выводимый документ или его часть в указанный PRN-файл, который впоследствии можно распечатать. Для срочного прекращения вывода на печать или в файл следует нажать клавиши 'Ctrl + Break' или 'Ctrl + A', при этом в первом случае после прекращения печати пакет производит прогон бумаги на принтере. Прекратив вывод, пакет переходит в состояние документа, не изменив текущий документ в памяти.

Если выводимый на печать документ шире строки принтера (80 или 132 колонки), то пакет печатает документ полосами. Например, при попытке распечатать документ с шириной строки 250 символов на 132-колонном принтере пакет будет выводить сначала полосу текущего документа шириной в 132 (столбцы 0—131), а затем полосу документа шириной 128 (столбцы 132—250). Для получения единого документа эти две выходные полосы достаточно соответствующим образом состыковать и склеить.

В мультипрограммном режиме при использовании печати другими программами пакет может встретить затруднения при установлении связи с принтером в момент инициации печати по команде 'Print'. Во избежание этого рекомендуется выполнять следующие действия. Для параллельных принтеров следует специализировать логические имена 'PRN' или 'LPT1' ('LPT2', если принтер связан со вторым портом вывода). Для последовательных принтеров предварительно в командном режиме пакета следует выполнить следующие команды MS DOS:

MODE COM1: 9600, N, 8, 1, P (или другие параметры, отвечающие конкретному принтеру ПК)

MODE LPT1: = COM1

Затем после запроса типа принтера ответить логическим именем 'PRN'. При использовании плоттеров в качестве устройства вывода предварительно в командном режиме пакета следует выполнить следующую команду MS DOS:

MODE COM1: 9600, N, 8, 1, P (или другие параметры, отвечающие конкретному плоттеру ПК)

И в ответ на запрос имени устройства ответить логическим именем 'COM1', а не 'PRN' или 'LPT1', даже в случае переназначения их к 'COM1' (например, по системной команде 'Assign').

Если после описанных процедур пакет не выводит документ на печать, то прежде всего следует проверить совместимость имеющихся устройств вывода ПК с типами соответствующих устройств, поддерживаемых пакетом (см. файлы MCAD.MCP, MORE.MCP и документацию [22]). При невозможности прямого вывода документа на печать рекомендуется поступать следующим образом. Ввести команду 'Print' пакета и на запрос размера выводимой области документа нажать клавишу 'Enter', а на последующий запрос имени устройства вывода ответить спецификатором файла, куда будет помещен весь документ в PRN-формате (например, 'C:Grodno.doc'). После этого следует сохранить текущий документ по команде 'Save' в дисковом файле и выйти из пакета по клавишам 'Ctrl + Q' или другим описанным способом в состояние MS DOS. Затем в состоянии MS DOS системной командой, например:

COPY C : Grodno.doc PRN

вывести (на ПК, имеющем подходящую периферию устройств печати) требуемый файл на печать.

Еще о некоторых моментах вывода текущего документа на печать следует сказать особо. В случае наличия в выводимом документе графической информации на принтере не всегда обеспечивается полная достоверность. Поэтому, по возможности, рекомендуется использовать в качестве устройства вывода плоттер. Описанная выше процедура вывода текущего документа на печать полностью соответствует принтеру, но для плоттера в качестве устройства вывода она имеет несколько особенностей, которые будут рассмотрены в разделе 8.4.

При сохранении результата вывода текущего документа в PRN-файле следует учитывать две особенности. Во-первых, с ростом размера документа существенно увеличивается требуемое время для его записи на диск. Во-вторых, существенно (относительно MCD-файла) растет объем занимаемого им места на диске. В качестве примера приведем оценки размеров MCD- и PRN-файлов, полученных на основе документа, рассмотренного в начале раздела (рис. 6). Для анализа было взято 6 документов, полученных из исходного документа путем дописывания к нему командой 'Append' определенного количества его же копий. Полученные результаты сведены в следующую интересную табл. 8.

Таблица 8

Документ (N×D)	Размер MCD-файла (M, байт)	Размер PRN-файла (P, байт)	P/M
1 × D = D	890	1844	2.1
3 × D	2567	7676	3.0
6 × D	3275	15341	4.7
9 × D	4709	23006	4.9
12 × D	6137	30671	5.0
15 × D	7568	38336	5.1

Из табл. 8 следует, что с ростом документа D относительно к предыдущему рост как величины M, так и величины P постепенно снижается, практически, с одинаковой скоростью. Тогда как величина P/M имеет тенденцию к росту, в определенной мере характеризую требуемые ресурсы внешней памяти ПК под MCD- и PRN-файлы для одного и того же выводимого документа. Еще более существенными оказываются временные издержки по созданию соответствующих MCD- и PRN-файлов с ростом размера выводимого документа. Таким образом, сохранять документ в PRN-файле следует только в случае крайней необходимости в связи с возникающими проблемами вывода его на печать, заранее подготовив для него место на томе (как показывает опыт, не менее чем в 6 раз большее относительно его MCD-файла), так как в случае нехватки места пакет «зависает» и потребуются повторная перезагрузка системы MS DOS, да и сами временные затраты на создание PRN-файла для большого документа достаточно велики. К тому же следует одновременно с PRN-файлом сохранять документ и в MCD-файле, так как обратная загрузка PRN-файла в память по команде 'Load' не восстанавливает исходный документ.

Таким образом, при выводе документа на печать в общем случае пользователь имеет возможность вывести его копию на принтер (PRN по умолчанию), в дисковый файл (указав в ответ на запрос устройства «СФ»), либо на экран дисплея (CON). Однако последнюю возможность использовать не рекомендуется, так как файл выводится в PRN-формате, что не только малоприспособно для анализа, но в ряде случаев может привести к «зависанию» пакета, требующего перезагрузки системы MS DOS, например, по клавишам 'Ctrl + Alt + Del'.

Настоящая глава представила только базовые возможности пакета MathCAD, которые наряду с основами работы в его среде (организация многих типов вычислений, использование текстовой информации, создание простых графиков, вывод документов на печать, сохранение их на дисках и т. д.) являются хорошим введением в изучение более сложных и глубоких возможностей пакета, рассматриваемых в последующих главах книги.

Глава 3

СУММЫ. ПРОИЗВЕДЕНИЯ, ПРОИЗВОДНЫЕ И ОПРЕДЕЛЕННЫЕ ИНТЕГРАЛЫ

При организации вычислений пакет MathCAD поддерживает работу с 29 математическими операциями, полный список которых приведен в прил. 8. В предыдущей главе были рассмотрены 13 из этих операций, а здесь будут изучены еще четыре и приведены примеры и особенности их использования.

3.1. СУММА И ПРОИЗВЕДЕНИЕ РАНЖИРОВАННЫХ ВЫРАЖЕНИЯ

Операция суммирования (произведения) суммирует (умножает) выражение, стоящее под знаком операции, по всем значениям p -переменной. Для ввода знака операции суммирования (произведения) следует нажать клавишу '\$' ('#'), после чего на экране генерируется знак соответствующей операции с двумя указателями (рис. 7, а, б).

Стоящий под знаком операции указатель требует определения имени p -переменной, а активный указатель (на котором устанавливается курсор) требует определения выражения (возможно функции от данной p -переменной), над которым будет произведена соответствующая ранжированная операция. Перемещение к очередному определяемому указателю можно осуществлять либо посредством КУК, либо клавишей табуляции 'Tab'. После полного определения соответствующей операции, нажав клавишу равенства (=), получаем результат вычисления.

На период вычисления в правой части ССП высвечивается, как правило, сообщение 'WAIT' с мигающим индикатором-точкой, идентифицирующее наличие процесса вычисления. Это не является во многих случаях избыточной информацией, так как ряд вычислений требует значительных временных затрат и пользователь в этом случае должен иметь возможность определенного контроля ситуации. Время вычисления выражений, рассматриваемых ниже, может быть весьма значительным и определяется рядом факторов. Во-первых, на время вычислений существенно влияет класс и тип используемого

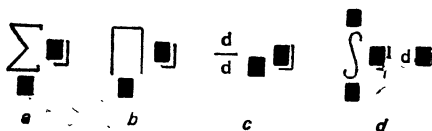


Рис. 7. Конструкции пакета MathCAD для операций ранжированных суммирования (а) и произведения (б), а также численных дифференцирования (с) и интегрирования (д)

ПК. Например, на отечественном ПК ИСКРА 1030 временные издержки при работе с пакетом весьма заметны. Так, например, четвертое выражение приводимого ниже примера вычисляется 15 с, а последнее определение требует для своего вычисления уже 49 мин. Во-вторых, данная ситуация объясняется не только быстродействием ПК и вычислительной сложностью выражений, но, в ряде случаев, и недостаточной эффективностью реализуемых пакетом вычислительных алгоритмов. В качестве простого примера, подтверждающего сказанное, приведем два фрагмента вычисления суммы 10^4 единиц в среде пакета (1) и в среде языка БЕЙСИК (2) ПК ИСКРА 1030:

$$(1) X := 1 \dots 10000 \sum_X 1 = 10000$$

$$(2) \text{FOR } K = 1 \text{ TO } 10000 : S = S + 1 : \text{NEXT } K : \text{PRINT } S$$

В первом случае потребовалось 47,8 с для вычисления, тогда как для эквивалентного вычисления во втором случае только 39,6 с, т. е. на 8,2 с меньше; в случае суммирования 10^5 единиц эта разница составляет уже 62 с, а это довольно значительная величина, учитывая простоту вычисляемой конструкции. В этом плане к недостаткам пакета следует отнести отсутствие в его среде службы времени, основанной на работе таймера. И вместо (или совместно) сообщения 'WAIT' было бы предпочтительнее видеть затрачиваемое на работу время.

При работе с требующими больших временных затрат вычислениями может оказаться полезным использование клавиш 'Ctrl + + Break', по которым пакет запрашивает санкцию 'Interrupt calculation?' на прекращение текущего вычисления в документе. По ответу 'Y' вычисление прекращается и пакет переходит в состояние документа, позволяя сделать необходимые изменения, для чего рекомендуется перевести предварительно пакет в М-режим командой 'Manual'. В противном случае вычисления в документе продолжаются и запрос на прерывание аннулируется. В случае прерывания вычисления пакет идентифицирует операцию выражения, на которой было прервано вычисление, сообщением 'Interrupted'.

Операции ранжированных суммирования и произведения можно использовать в тех местах выражений, где допустимо использование любого другого выражения, включая определения. Приведем фрагмент документа, иллюстрирующий использование обоих ранжированных операций:

$$\{H\} \text{MIN} := 0.1 \text{ H} := 0.2 \text{ MAX} := 1 \text{ min} := 0.2 \text{ h} := 0.3 \text{ max} := 2 \\ X := \text{MIN}, H \dots \text{MAX} \text{ Y} := \text{min}, h \dots \text{max} \text{ } \infty := 10^8 \text{ N} := \dots \infty \\ n := 5 \text{ t} := 0.5 \text{ Z} := 0.3, 0.5 \dots 5 \text{ A} := 1, 1.1 \dots 10$$

$$\sum_A 1 = 91 \prod_A 1 = 1 \sum_A A = 500.5 \prod_A A = 2.572 \cdot 10^{61}$$

$$\sum_X (X^2 + X^3) = 6.875 \prod_X (\sin(X) + \cos(X)) = 15.434$$

$$\sum_X \sum_Y (X^2 + X \cdot Y + Y^2) = 475 \prod_X \left[\sum_Y \sqrt{Y} \cdot \sin(Y) + \sum_X \ln(X) \right] = 1.092 \cdot 10^9$$

$$\prod_X X^{\sum Y} + \sum_Y Y^{\prod X} = 18.999 \quad \pi := 4 \cdot \sum_N (-1)^{N+1} \cdot \frac{1}{2 \cdot N - 1} = 3.141$$

$$N := 1 \dots 10 \quad M := \sum_N \frac{n + N - 1}{(n - 1)! \cdot N!} \cdot t^N \quad M = 0.142 \quad \ln \left[\sum_X \prod_Y X^Y \right] = 0.114$$

$$[K] \quad L := \frac{\prod_Y Y^2 + \prod_X \sqrt{X} + \prod_Z Z^2}{\sum_X \sum_Y \sum_Z (X \cdot Y \cdot Z) \cdot \prod_X \prod_Y \prod_Z (X + Y + Z)} \quad L = 0$$

Приведенный фрагмент документа достаточно прозрачен и особых пояснений не требует. Однако следует помнить, что в качестве индекса операции (располагающегося под ее знаком) следует использовать только p -переменную, а не скаляр или p -выражение, при этом p -переменная не обязательно является целочисленной.

Совместно с операциями ранжированных суммирования и произведения можно использовать и условные p -операторы, позволяющие производить ранжирование по выбранным значениям p -переменных. Однако, если для ранжированного суммирования данная возможность в целом ряде случаев оказывается весьма полезной, то для ранжированного произведения она особого смысла не имеет. Причина этого состоит в том, что как только p -оператор исключает какое-либо значение p -переменной из вычисления, то выражение под знаком произведения на данном значении становится нулевым и все произведение в целом также превращается в нуль. В операции же суммирования это может иметь определенный смысл. Например, следующий фрагмент иллюстрирует суммирование только положительных значений некоторой функции с использованием условного p -оператора и встроенной функции 'if':

$$[H] \quad X := 0, 0.2 \dots 3.5 \quad A := -48 \quad Y(X) := \cos(X)$$

$$S_1 := \sum_X (X \neq i \text{ if } (Y(X) > 0, A, X)) \cdot Y \quad S_2 := \sum_X Y(X)$$

$$[K] \quad S_1 = 5.496 \quad S_2 = -1.257$$

Таким образом, операции ранжированных суммирования и произведения могут оказаться весьма полезными при организации вычислений, связанных с представлениями функций различного рода числовыми или функциональными рядами, специальными функциями, приближенными вычислениями и т. д. Вопросы использования операций векторного суммирования будут рассмотрены в следующей главе.

3.2. ЧИСЛЕННОЕ ВЫЧИСЛЕНИЕ ПРОИЗВОДНЫХ

В среде пакета производится численное вычисление значений m -кратных ($m \geq 1$) и частных производных функций в заданных точках. Реализованный в пакете метод численного дифференцирования базируется на аппроксимации, использующей значения функции в четырех точках, прилегающих к точке дифференцирования. Следует отметить, что итерационный процесс приближенного вычисления

производной имеет ограничение на число итераций и если оно пре-
взойдено без получения результата, то вычисление прекращается
с идентификацией ошибочной ситуации 'Not converging', т. е. итера-
ционный процесс не сходится. В конце раздела о такой ситуации будет
сказано несколько подробнее.

Для определения операции дифференцирования следует нажать
клавишу вопроса (?), по которой на экране генерируется знак опера-
ции с двумя указателями (рис. 7, с). Стоящий в знаменателе указатель
требует определения переменной дифференцирования, тогда как
активный указатель — дифференцируемой функции. Переменная
дифференцирования и все входящие в дифференцируемую функцию
константы и переменные должны быть предварительно определены
локально или глобально, при этом допускается сочетание обоих ти-
пов определений. Переменная дифференцирования идентифицируется

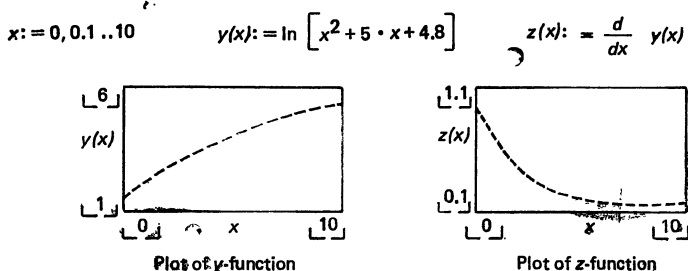


Рис. 8. Использование p -переменных для вывода графиков
функции и ее производной

уникальным именем, удовлетворяющим соглашениям пакета. Диф-
ференцируемая функция может быть как действительной, так и ком-
плексной. Приведем примеры на численное дифференцирование
функций в среде пакета:

[H] $X := 1.5$ $Y := 2$ $Z := 19.42$ $V := 5 + 3.4i$ $UV := 5$

$F(X) := 5 \cdot X^2 + 3 \cdot X + 8$ $G(X, Y) := (X + Y)^3$ $W(X, Y, Z) := X^2 \cdot Y + Y^3 \cdot X$
 $\times Z + Z^3 \cdot X \cdot Y$

$\frac{d}{dX} F(X) = 18$ $\frac{d}{dUV} \sqrt{UV} = 0.224$ $F(X) := \frac{d}{dY} G(X, Y)$ $\frac{d}{dX} F(X) = 21$

$\frac{d}{dX} G(X, Y) + \frac{d}{dY} G(X, Y) = 73.5$ $\frac{d}{dV} F(V) = 53 + 34i$ $R(X) := \frac{F(X)}{\cos(X)}$

$\frac{d}{dX} W(X, Y, Z) + \frac{d}{dY} W(X, Y, Z) + \frac{d}{dZ} W(X, Y, Z) = 1526.427$

$\frac{d}{dV} R(V) = 5.528 + 10.534i$ $\frac{d}{dX} [e^{\cos(x)} + \sin(X)] = -1$

[K] $\frac{d}{dX} \left[e^{\tanh(x)} + \sinh(x) \cdot \frac{d}{dX} \sqrt{\ln(x) + x^2} \right] = 2.45574$

В качестве переменных дифференцирования, констант и пере-
менных дифференцируемой функции могут выступать и p -переменные.
В этом случае результатом дифференцирования является выходная

таблица или вектор, содержащая значения производной по всей совокупности значений p -переменных. Данная возможность пакета позволяет, например, вывести график производной любой плоской функции или частной производной в заданной секущей плоскости. Фрагмент документа (рис. 8) иллюстрирует использование p -переменных для построения графиков исходной функции и ее производной.

При проведении приближенных вычислений (производные, интегралы и т. д.) пакет существенно использует значение предопределенной переменной TOL (по умолчанию $TOL = 0.001$). Так при вычислении производной процесс последовательных приближений к искомому результату прекращается, когда разность между двумя последовательными значениями итераций по модулю меньше значения TOL. Естественно, при уменьшении значения TOL точность вычислений и требуемое для этого время растут, и наоборот. Следующий простой пример иллюстрирует влияние значения TOL на результат вычисления в точке $X = 1.48$ производной функции

$$Y(X) = \frac{1}{X} \cdot \cos(X) + \frac{X^3}{\ln(X)} + \ln(X) \cdot e^X,$$

который отражает табл. 9.

Естественно, на точность вычислений влияет не только величина значения переменной пакета TOL, но и сам характер поведения функции в точке ее дифференцирования: чем сильнее изменения функции в области дифференцирования, тем больше сказывается на результате вычисления значение переменной TOL пакета MathCAD.

В следующих трех случаях пакет не может вычислить значение производной и выдает диагностическое сообщение 'Not converging':

1) в заданной точке производной не существует;

2) при сильно колебательном поведении функции в окрестности точки дифференцирования;

3) значение переменной TOL так мало, что ошибки округления делают невозможным вычисление производной с требуемой точностью.

В первых двух случаях следует произвести анализ дифференцируемой функции математическими методами либо грубо оценить ее поведение с помощью графика, построенного тут же в документе. Иногда такой подход дает довольно быстрый результат. Тогда как в третьем случае следует увеличить значение переменной пакета TOL, введя ее новое определение в текущий документ, например, 'TOL: = 0.01'.

Таблица 9

Значение TOL	Значение производ- ной $\frac{d}{dX} Y(X)$
10^{-5}	6.486187
10^{-4}	6.486188
10^{-3}	6.486208
10^{-2}	6.488348
10^{-1}	6.891320

3.3. ВЫЧИСЛЕНИЕ ОПРЕДЕЛЕННЫХ ИНТЕГРАЛОВ

Пакет позволяет вычислять как обычные m -кратные ($m \geq 1$) определенные, так и криволинейные интегралы. Вычисление определенных интегралов осуществляется методом Ромберга в сочетании с его модификацией. Как и в случае вычисления производных, пакет устанавливает предельное значение количества итераций, требуемых для приближенного вычисления интеграла. Если этот предел достигнут без получения результата или если подынтегральная функция имеет особенности на одном или обоих концах отрезка интегрирования, то пакет переходит на использование модифицированного метода Ромберга [21]. Для модифицированного метода пакет также устанавливает предельное количество итераций и, если оно будет превзойдено без получения искомого результата, то вычисление прекращается, а интеграл диагностируется сообщением 'Not converging', т. е. итерационный процесс вычисления расходится.

На самом деле пакет позволяет вычислять многие интегралы, подынтегральные функции которых имеют особенности на концах отрезка интегрирования. Вычисление таких интегралов в целом ряде случаев требует больших временных затрат, чем для случая хорошо определенных подынтегральных функций. Подобно случаю вычисления производных при очень маленьких значениях предопределенной переменной TOL процесс интегрирования может завершиться аварийно с диагностикой 'Not converging', даже если интеграл имеет конечное значение. Наряду с этим пакет не может интегрировать функции, имеющие особенности в точках внутри отрезка интегрирования. Подынтегральные функции, имеющие большое число конечных разрывов (например, ступенчатые функции), также могут привести к диагностическому сообщению 'Not converging'. Если особенности интегрируемой функции известны, то рекомендуется в ряде случаев вычисляемый интеграл представлять в виде суммы интегралов с особыми точками как пределами интегрирования. При этом чертеж подынтегральной функции может оказать существенную помощь при анализе ее поведения. Сделать его можно описанными средствами пакета тут же в текущем документе, что повышает оперативность принятия решения.

Для ввода знака операции интегрирования следует нажать клавишу амперсанда (&), по которой на экране появляется знак операции с четырьмя указателями (рис. 7, *d*). Для определения и вычисления интеграла необходимо вместо активного указателя задать подынтегральную функцию (подынтегральная функция должна быть указана явно, в случае равенства ее константе кодируется данная константа, а не выносится за знак интеграла), после знака дифференциала (*d*) следует определить переменную интегрирования, а вместо указателей снизу и сверху знака интеграла указать действительные выражения, определяющие соответственно значения нижнего и верхнего пределов интегрирования. После завершения определения интеграла и нажатия клавиши равно (=) пакет производит вычисление интеграла и возвращает после знака равенства его значение (если оно существует

и может быть получено пакетом). На период вычисления в правой части ССП появляется сообщение 'WAIT' с мигающим индикатором-точкой, идентифицирующее наличие процесса вычисления, который для случая сложных в вычислительном отношении интегралов может быть достаточно длительным. При этом, если определение некоторой переменной или функции содержит интегральное выражение, то оно вычисляется уже в рамках данного определения, а не при первом использовании переменной или функции. В ряде вычислительных алгоритмов это является не совсем удобным свойством пакета.

В качестве пределов интегрирования должны выбираться только действительные выражения, тогда как подынтегральная функция может быть как действительной, так и комплексной. Кроме переменной интегрирования (имя которой должно удовлетворять соглашениям пакета) остальные переменные и константы, используемые в конструкции, должны быть определены локально или глобально. В качестве пределов интегрирования могут использоваться и предопределенные переменные пакета, включая символ бесконечности (∞). Однако при использовании в этом качестве символа бесконечности в ряде случаев рекомендуется с учетом специфики вычисления переопределять для данной переменной значение в сторону уменьшения (например, $\infty := 10^{100}$), ибо при стандартном значении возможны аварийные ситуации с диагностикой 'Overflow'. Это может иметь место для подынтегральных функций, вызывающих при значениях переменной (∞) по умолчанию ситуацию 'Переполнение'. Возможность использования переменной пакета (∞) позволяет не только вычислять, но и оформлять несобственные интегралы в общепринятом виде. Приведем фрагмент документа, содержащего вычисления различного типа интегралов в среде пакета:

$$\begin{array}{ll}
 [H] & Y:=1 \quad Z:=2 \quad \infty:=10^{100} \\
 F(X):=5 \cdot X^2 + 4 \cdot X - 8 & \int_{42}^{90} \ln(AVZ) \, dAVZ = 200.001 \quad \int_Y^Z F(X) \, dX = 9.667 \\
 \int_{\ln(Y)}^{\sqrt{Z}} \ln(X) \cdot F(X) \, dX = 0.409 & \int_Y^Z \frac{\sin(Y)}{\cos(Y) + i \cdot \sin(Y)} \, dY = 0.059 - 0.917i \\
 \int_0^1 \ln(\Gamma(X+1)) \, dX = -0.081 & \int_{\sin(Y)}^{1+\cos(Z)} \frac{\cos(X)^2}{1-4 \cdot \cos(X) + 48} \, dX = -0.003 \\
 \int_{Y+Z}^{Z^2} F(X) \cdot \operatorname{atan}(X) \, dX = -40.275 & \int_1^2 \operatorname{atan}(t) \cdot \operatorname{atanh}(t) \, dt = 0.891 - 1.525i \\
 \int_0^{\infty} \left[\frac{1}{1+X^2} - \frac{1}{1+X} \right] \cdot \frac{1}{X} \, dX = 0 & \int_0^{\infty} \frac{X}{\tanh(X)} \, dX = 5 \cdot 10^{199}
 \end{array}$$

$$\int_0^{\pi} \cos(48 \cdot t) \cdot \cos(43 \cdot t) dt = -9.173 \cdot 10^{-15} \quad \pi := 2 \cdot \int_0^{\pi/2} \frac{\sinh(Z)}{\sin(Z)} dZ \quad \pi = 3.142$$

$$\int_0^1 e^{i \cdot X} dX = 0.841 + 0.46i \quad \int_0^1 (\cos(X) + i \cdot \sin(X)) dX = 0.841 + 0.46i$$

$$\int_0^{\pi/2} \sin(X) \cdot \cos(X)^3 dX = 0.25 \quad \int_0^1 \sqrt{[1 - X^2] \cdot [2 - X^2]} dX = 1.036$$

$$\int_0^{\pi/3} e^{\cos(X) + i \cdot \sin(X)} \cdot \sin(X) dX = 0.872 + 0.58i$$

$$\int_2^5 [e^X + e^{-X} - i \cdot \cos(2 \cdot X)] dX = 141.153 - 0.106i \quad \text{TOL} := 0.1$$

$$\int_0^1 \left[e^{\tanh(X)} + \sinh(X) \cdot \frac{d}{dX} \sqrt{\ln(X) + X^2} \right] dX = 2.40559 - 0.71989i$$

[K]

Подобно случаю вычисления производных пакет в качестве пределов интегрирования, а также констант и переменных подынтегрального выражения допускает использование p -переменных, что дает возможность выводить результаты вычисления интеграла в виде выходных таблиц (например, для табуляции интеграла в зависимости от значений пределов интегрирования и/или значений констант и переменных, входящих в его подынтегральную функцию). Данная возможность позволяет также получать достаточно точные графики первообразных функций. Следующий простой фрагмент документа иллюстрирует построение графика первообразной для функции $Y(X)$ на основе табуляции значений интеграла по его верхнему пределу интегрирования:

$$[H] \quad X := 0, 0.1 \dots 1 \quad Y(X) := \frac{1}{1+X} \quad Z(X) := \int_0^X Y(X) dX$$

[K]

Рис. 9

Следующие примеры иллюстрируют использование p -переменных в качестве пределов интегрирования и констант подынтегрального выражения:

$$[H] \quad Y := 1 \quad N := 1 \dots 4 \quad \varphi := 1 \dots 2 \quad K := 1 \dots 2 \quad M := 1 \dots 4$$

$$F(\varphi, K) := \int_0^{\varphi} \cos(K \cdot \varphi) d\varphi \quad R(M) := \int_Y^{Z \cdot M} \sum_N \cos(X)^N dX$$

K	φ	F (φ, K)	M	R (M)
1	1	0.841	1	0.18
2	2	0.455	2	-0.174
		0.909	3	1.612
		-0.378	4	5.739

Среднеквадратичные значения тока в цепи выпрямителя:

$$K := 0 \dots 4 \quad J_{mK} := 50 \cdot (K + 1) \quad T := \pi \quad \omega := 50$$

$$J_K := \left[\frac{1}{T} \cdot \int_0^T J_{mK}^2 \cdot \sin(\omega \cdot t) dt \right]^{\frac{1}{2}}$$

$$[K] \quad J^T = (5.64 \quad 11.28 \quad 19.93 \quad 22.57 \quad 28.21)$$

Приведенные примеры использования *p*-переменных достаточно прозрачны и особых пояснений не требуют.

При интегрировании функций, имеющих внутри отрезка интегрирования особые точки (например, сингулярность, разрывы и т. д.), в ряде случаев можно использовать прием, состоящий в специальном определении подынтегральной функции (например, устраняющей сингулярность или «сглаживающей» разрыв). Пусть, например, требуется проинтегрировать модуль функции, заданной на интервале $(-\infty, 0]$ уравнением $Y(X) = X$, а на интервале $(0, \infty)$ уравнением $Z(X) = 8$. Очевидно, в точке $X = 0$ определенная таким образом функция имеет конечный разрыв (скачок). Следующий фрагмент решает данную задачу:

$$[N] \quad Y(X) := X \quad Z(X) := 8 \quad U(X) := \text{if}(X \leq 0, Y(X), Z(X))$$

$$[K] \quad J := \int_{-1}^1 |U(X)| dX \quad J = 8.4997$$

В некоторых случаях может оказаться эффективным сочетание данного подхода с уменьшением значения переменной TOL пакета, например, в случае устранения сингулярности подынтегральной функции.

Следует отметить возможность использования условных операторов в среде пакета для конструкций непрерывного типа, которая, однако, при своей допустимости не влияет на результаты вычислений. Простые примеры иллюстрируют сказанное:

$$[N] \quad \int_0^4 (0 < X \leq 1) \cdot (2 < X \leq 4) \cdot X^3 dX = 64$$

$$[K] \quad \int_0^4 \frac{0 \leq X \leq 1}{2 \leq X \leq 4} dX = 4 \quad \int_0^4 (0 \leq X \leq 1) \cdot (2 \leq X \leq 4) dX = 4$$

Точность вычисления интеграла, как и в случае ⁷ производных, определяется значением предопределенной переменной TOL пакета (по умолчанию TOL = 0.001). Однако из-за сглаживающего характера операции интегрирования в большинстве случаев влияние изменения величины TOL не столь существенно, как при вычислении производных. Однако и здесь следует иметь в виду, что значительное уменьшение величины переменной TOL в целях повышения точности вычислений не только ведет к увеличению временных затрат (которые для многих видов и типов интегралов могут быть весьма значительными), но и может служить в ряде случаев причиной прекращения вычислений с идентификацией интеграла сообщением 'Not converging'. Это же сообщение наряду с перечисленными причинами может возникнуть и при использовании интервала интегрирования $[0, \infty)$ для несобственных интегралов.

Наряду с однократными пакет позволяет вычислять также m -кратные ($m > 1$) интегралы, для чего в конструкцию вводятся подряд знаки интеграла по числу (m) кратности вычисляемого интеграла. Однако к m -кратным интегралам следует прибегать только в случае необходимости, так как с ростом кратности резко растет время их вычисления. Например, уже для вычисления двухмерного интеграла

$$\int_0^1 \int_0^1 e^{x+y} dx dy = 2.95249$$

в среде пакета требуется 97 с, тогда как для эквивалентной ему конструкции

$$\left[\int_0^1 e^x dx \right] \cdot \left[\int_0^1 e^y dy \right] = 2.95249$$

требуется при прочих равных условиях только 6 с. Для более сложных интегралов разница во времени вычислений может быть значительно существеннее. Поэтому (если это возможно) при оформлении документа в М-режиме можно использовать произвольные допустимые интегральные конструкции, однако перед их вычислением рекомендуется произвести над ними эквивалентные упрощающие преобразования, что позволит в ряде случаев резко снизить время вычислений. Основным направлением в этом отношении является сведение исходной интегральной конструкции к одной или нескольким связанным конструкциям, содержащим интегралы, по возможности, меньшей кратности. Приведем фрагмент документа, содержащего вычисления кратных интегралов:

[H] 1. Вычисление сложной интегральной конструкции X:=1 Y:=2

$$\frac{\int_{\sin(X)}^{\sin(Y)} \sqrt{1-X^2} dX}{\int_{\sqrt{Y}}^X X \cdot Y dX dY + \int_{\ln(X)}^{X+Y} \int_{\frac{X}{2}}^{\sqrt{Y}} \int_X [X^2 + Y^2 + Z^2] dX dY dZ} = 0.03379$$

Время вычисления такой конструкции составляет 47 мин 34 с.

2. Вычисление объема шара радиуса $R:=1$

$$V_{\text{Sphere}} := 8 \cdot \int_0^1 \int_0^{\sqrt{1-X^2}} \int_0^{\sqrt{1-X^2-Y^2}} 1 \, dZ \, dY \, dX \quad V_{\text{Sphere}} = 4.18879$$

3. Вычисление площади круга радиуса R

$$[K] \quad S := 4 \cdot \int_0^1 \int_0^{\sqrt{1-X^2}} 1 \, dY \, dX \quad S = 3.14156$$

В данном фрагменте наряду с областями вычислений продемонстрировано использование текстовых областей и зон. Интегралы по поверхности и по объему путем сведения их к двойным и тройным интегралам также можно вычислять в среде пакета MathCAD.

Наряду с m -кратными ($m \geq 1$) интегралами пакет позволяет вычислять и криволинейные интегралы путем сведения их к однократным интегралам, подынтегральная функция которых содержит производные. Действительно, криволинейный интеграл является обобщением однократного, когда отрезок интегрирования заменяется на кривую линию (L). В этом случае различают криволинейные интегралы 1-го и 2-го родов. Криволинейные интегралы 1-го рода определяются как интегралы по длине кривой

$$J^1 = \int_{(L)} F(X, Y) \, dS,$$

тогда как 2-го рода — как интегралы по проекции на оси координат

$$J_1^2 = \int_{(L)} F(X, Y) \, dX \quad \text{и} \quad J_2^2 = \int_{(L)} F(X, Y) \, dY.$$

Предположим, что отрезок интегрирования плоской кривой задан параметрически

$$X = X_1(t); \quad Y = Y_1(t) \quad (t_1 \leq t \leq t_2).$$

Тогда имеют место следующие соотношения:

$$\begin{aligned} \int_{(L)} F(X, Y) \, dS &= \int_{t_1}^{t_2} F(X_1(t), Y_1(t)) \sqrt{X_1'^2(t) + Y_1'^2(t)} \, dt \\ \int_{(L)} F(X, Y) \, dX &= \int_{t_1}^{t_2} F(X_1(t), Y_1(t)) X_1'(t) \, dt \\ \int_{(L)} F(X, Y) \, dY &= \int_{t_1}^{t_2} F(X_1(t), Y_1(t)) Y_1'(t) \, dt \end{aligned}$$

Правые части этих соотношений представляют собой обычные однократные интегралы по переменной интегрирования t , подынтегральная функция которых содержит обычные производные по той же переменной t . Трехмерный случай рассматривается аналогично. Приведем

фрагмент документа, содержащего вычисления криволинейных интегралов 1-го и 2-го родов с учетом сказанного:

[Н] 1. Вычисление криволинейного интеграла 1-го рода

$$J_1 = \int_{(L)} (X^2 + Y^2 + Z^2) dS \square$$

где L — один виток винтовой линии, заданной параметрически

$$X = A \cos(t), \quad Y = A \sin(t) \quad \text{и} \quad Z = Bt \quad (0 \leq t \leq 2\pi)$$

$$A := 3 \quad B := 4 \quad t_1 := 0 \quad t_2 := 2 \cdot \pi \quad X(t) := A \cdot \cos(t)$$

$$Y(t) := A \cdot \sin(t) \quad Z(t) := B \cdot t \quad F(X, Y, Z) := X^2 + Y^2 + Z^2$$

$$J_1 := \int_{t_1}^{t_2} F(X(t), Y(t), Z(t)) \cdot \sqrt{\left[\frac{d}{dt} X(t)\right]^2 + \left[\frac{d}{dt} Y(t)\right]^2 + \left[\frac{d}{dt} Z(t)\right]^2} dt$$

(Результат вычисления интеграла J_1) $J_1 = 6897.4157$

2. Вычисление криволинейного интеграла 2-го рода

$$J_2 = \int_{(L)} XY dX + YZ dY + ZX dZ \square$$

где L — кривая, заданная параметрически $X = t \quad Y = 2t^2 \quad Z = 3t^3 \quad (0 \leq t \leq 1)$;

$$t_1 := 0 \quad X(t) := t \quad Y(t) := 2 \cdot t^2 \quad Z(t) := 3 \cdot t^3$$

$$J_2 := \int_{t_1}^{t_2} \left[X(t) \cdot Y(t) \cdot \frac{d}{dt} X(t) + Y(t) \cdot Z(t) \cdot \frac{d}{dt} Y(t) + Z(t) \cdot X(t) \cdot \frac{d}{dt} Z(t) \right] dt$$

[К] (Результат вычисления интеграла J_2) $J_2 = 7.78571$

Другие интересные примеры численного дифференцирования и интегрирования в среде пакета будут рассмотрены в гл. 10. Следует иметь в виду, что операции дифференцирования и интегрирования не векторизуются (клавиши 'Alt + (—)'), что не позволяет, в частности, применять их к матричным или векторным функционалам.

Глава 4

ВЫЧИСЛЕНИЯ С ВЕКТОРАМИ И МАТРИЦАМИ В СРЕДЕ ПАКЕТА

Наряду со скалярными объектами пакет позволяет производить обработку массивов. Основными объектами изучения в главе будут одномерные (векторы) и двумерные (матрицы) массивы. Пакет для работы с данными объектами предоставляет широкий набор операций и ветроенных функций. Как правило, пакет имеет дело с векторами-столбцами и квадратными матрицами, но допускаются также вектора-строки и прямоугольные матрицы. Работу с массивами размерности больше двух пакет не поддерживает, что в определенной мере ограничивает его применение. При необходимости работы с массивами большей размерности рекомендуется использовать другие программные средства.

4.1. СОЗДАНИЕ ВЕКТОРОВ И МАТРИЦ В СРЕДЕ ПАКЕТА

Для ввода и определения вектора или матрицы следует использовать команду

MATRIX | Alt + M | C G

по которой пакет выдает запрос 'Array size (rows columns):' на определение числа строк и столбцов массива. Ответ на данный запрос вводится в виде «N M» и определяет не только размер массива, но и его тип согласно табл. 10.

Таблица 10

Rows	Columns	Тип и размер создаваемого массива
N	M	Прямоугольная (N×M)-матрица; N, M > 1
N	[1]	Вектор-столбец из N элементов
1	M	Вектор-строка из M элементов
—	—	Повтор запроса на размер массива

После получения ответа на запрос пакет выводит на экран макет массива, состоящий из ограничивающих скобок и указателей для его элементов. Например, по ответу «3 5» выводится макет

$\left[\begin{array}{ccccc} \blacksquare & \blacksquare & \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare & \blacksquare & \blacksquare \end{array} \right]$	$\left[\begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \end{array} \right]$
--	---

Для завершения определения массива необходимо все входящие в его макет указатели заменить соответствующими скалярными выражениями, допустимыми пакетом. Для более быстрого перехода от одного указателя к другому рекомендуется использовать клавишу табуляции 'Tab', а не КУК. Так для заполнения приведенного выше макета (3 × 5)-матрицы достаточно, например, произвести следующие действия:

1 'Tab' 6 'Tab' 11 'Tab' 2 'Tab' ... 10 'Tab' 15

в результате чего сформированная матрица принимает вид, указанный справа от ее макета. Каждый массив идентифицируется уникальным именем, удовлетворяющим соглашениям пакета.

Элементы массива можно определять в терминах p -переменных, например, $M_{i,j} := i \cdot j$, где i, j — ранжированные целочисленные переменные. Следует помнить, что, определяя массив через его элементы, обязательно задавать значения для всех его элементов. В этом случае неопределенные элементы получают по умолчанию нулевые значения, например, определяя четыре элемента матрицы M

$M_{1,1} := 5 \quad M_{1,2} := 4 \quad M_{2,1} := 6 \quad M_{2,2} := 1,$

получаем матрицу

$$M = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 5 & 4 \\ 0 & 6 & 1 \end{bmatrix}.$$

Данный подход можно использовать и для определения в документе векторов и матриц, не прибегая к основной команде 'Matrix', а также массивов максимально допустимого пакетом размера, однако это требует больших временных затрат и внимательности. Например, по определению « $A_{88,88} = 0$ » создается нулевая квадратная (89×89)-матрица. После создания массива он может принимать участие во многих векторных и матричных операциях, допускаемых пакетом для данного типа объектов. При этом пакет рассматривает вектор как $(N \times 1)$ -матрицу. Подробнее операции с векторами и матрицами рассматриваются в следующем разделе.

Наряду с созданием пакет позволяет изменять размер существующего массива и даже его тип. Для этого достаточно поместить курсор на элемент массива, где необходимо сделать изменения, и нажать клавиши 'Alt + M'. В результате этой процедуры пакет запрашивает требуемые действия 'Insert / delete (+/-) rows cols:' по вставке / удалению строк и столбцов существующего массива. В качестве ответа следует ввести два числа соответственно для строк и столбцов. При этом положительные числа добавляют соответственно строки и столбцы, а отрицательные удаляют их; нулевые значения не изменяют числа соответствующих компонент редактируемого массива. Пакет вставляет строки ниже позиции курсора и столбцы правее ее. Для вставки строки выше первой и столбца левее первого следует курсор поместить на левую скобку массива и нажать клавиши 'Alt + M'. Пакет удаляет строки от позиции курсора вниз и столбцы от позиции курсора вправо, включая содержащие сам курсор. Подобные корректировки размерности массива можно осуществлять и над векторами, добавляя к ним либо строки, либо столбцы. При этом корректным является преобразование таким способом матриц в вектора и наоборот. Элементы вновь добавляемых в массив строк и/или столбцов идентифицируются указателями, требующими своего определения. Например, (3×4) -матрица A, стоящая слева с курсором на $(0,1)$ -элементе, преобразуется в квадратную (2×2) -матрицу A после ответа '—1 —2' на запрос 'Insert/delete (+/-) rows cols:', тогда как ответ '2—1' превращает вектор-строку B в (3×3) -матрицу B:

$$A: = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{bmatrix} \rightarrow A: = \begin{bmatrix} 5 & 8 \\ & \\ 9 & 12 \end{bmatrix} \quad B: = (1 \ 2 \ 3 \ 4) \rightarrow B: = \begin{bmatrix} 1 & 3 & 4 \\ \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare \end{bmatrix}$$

Для определения массива следует ввести его имя с символом $\{ : | \sim \}$, а затем использовать клавиши 'Alt + M'. После этого произвести описанные выше действия. Не следует присваивать массиву имя, совпадающее с именем скалярной переменной и наоборот, так

как пакет использует описания переменных с одинаковыми именами в соответствии с их областями действия. В терминах определенного массива можно определять и другие массивы, например, $M := A$, где A — ранее определенный массив.

Для обращения к элементу существующего массива или при создании массива через определение его элементов следует использовать операцию нижнего индекса, вводимую по клавише 'I', например:

$$M[(i, j)] \rightarrow M_{i,j} := \blacksquare \quad V[j] \rightarrow V_j := \blacksquare$$

Индексированную переменную можно использовать в вычислениях как обычную скалярную переменную. Для корректного определения двойных индексов их следует кодировать в скобках, которые можно указывать либо явно, либо с помощью клавиши апострофа ('), например: $M[(k, j)]$ или $M[']$. Во втором случае в генерируемые скобки вместо указателя следует помещать требуемые индексы или их значения. После вывода курсора из области выражения, содержащего индексированные переменные, индексные скобки аннулируются и визуализируются только по возвращении курсора в область либо в случае возникновения ошибки.

По умолчанию значения индексов массива начинаются с нуля и до максимально допустимого пакетом. Изменить начальное значение для индексов можно с помощью предопределенной переменной `ORIGIN` (по умолчанию `ORIGIN = 0`), присвоив ей соответствующее значение. При этом во избежание возможных недоразумений определять переменную `ORIGIN` пакета следует глобально (например, `ORIGIN ≡ 1`), что позволяет в рамках всего документа унифицировать использование индексов массивов. При этом можно использовать для разных массивов документа различные значения переменной `ORIGIN`, так как она выступает как обычная переменная и в зависимости от уровня определения (локальный или глобальный) имеет области действия в документе согласно соглашениям пакета. Значения переменной `ORIGIN` могут быть как положительными, так и отрицательными, но обязательно целочисленными. Например, при `ORIGIN := -20` значения индексов массивов будут начинаться с «-20». При определении нового значения для переменной `ORIGIN` пакет при каждом использовании элементов массивов, находящихся в области ее действия, перенумеровывает их согласно этому значению. Попытка использования индекса со значением, меньшим указанного последней переменной `ORIGIN`, идентифицируется диагностическим сообщением 'Index out of bounds'.

Еще об одном важном моменте нумерации элементов массива следует сказать особо. При глобальном определении массива его индексы также требуют глобального определения. Более того, если для изменения начального значения индексов глобально определенного массива использована переменная пакета `ORIGIN`, то она также должна быть определена глобально, в противном случае для него действует значение первого в документе глобального определения (или по умолчанию) переменной `ORIGIN`. Для локально определен-

ных массивов указанное замечание не имеет силы. Следующий пример иллюстрирует сказанное:

[H] $\text{ORIGIN} \equiv -1$ $\text{ORIGIN} := 1$ $i := 1..2$ $j := 1..2$

$M_i := i$ $M^T = (0 \ 0 \ 1 \ 2)$ $B_j := 2 \cdot j$ $B^T = (2 \ 4)$

$\text{ORIGIN} \equiv 1$ $i := 1..2$ $M_i := i$ $M^T = (1 \ 2 \ 1 \ 2)$

$B^T = (2 \ 4)$ $\text{ORIGIN} \equiv 3$ $i := 3..4$ $M_i := i$

$M^T = (3 \ 4 \ 1 \ 2)$ $j := 3..4$ $b_j := 3 \cdot j$ $B^T = (9 \ 12)$

[K] Влияние определения ORIGIN на индексацию массивов

При использовании массивов пакет накладывает ряд ограничений на их размеры в зависимости от типа их использования: определения, отображения на экран или вывода на печать. В среде пакета массив не может быть определен размером более, чем в 8000 элементов. Например, квадратная матрица имеет максимальный размер (89 × 89). Пакет не позволяет по команде 'Matrix' создавать новую матрицу (вектор) или расширять существующую до размера большего, чем в 100 элементов. При попытке ввести на запрос 'Array size:' недопустимые размеры массива пакет их игнорирует и предлагает повторить ответ. Связано это с тем, что определять массив с большим числом элементов непосредственно в документе поэлементно весьма трудоемко и для этих целей следует использовать другие подходы и возможности пакета (например, определение массива через его элементы, являющиеся функцией индексов: $i := 0..88$ $j := 0..88$ $M_{i,j} := F[i,j]$). Вопрос ввода больших массивов данных в документ будет рассмотрен подробно в гл. 7, посвященной работе в среде пакета с файлами данных различной организации.

При попытке вывода на экран массива, содержащего более 200 строк или 200 столбцов, пакет оставляет для вывода только 200 строк и 200 столбцов. Это происходит в том случае, когда выводится на экран большой массив, являющийся результатом вычисления выражения, стоящего до знака равенства ($=$). В этом случае вывод элементов массива усекается следующим образом: если массив имеет слишком много строк, то пакет отделяет снизу угловыми скобками и тремя пунктирными строками выводимые 200 строк; если же массив имеет слишком много столбцов, пакет отделяет справа выводимые 200 столбцов правой скобкой с тремя пунктирными столбцами. При этом пакет продолжает внутренний просмотр всех элементов массива, даже если они на экран не выводятся.

4.2. ОПЕРАЦИИ НАД ВЕКТОРАМИ И МАТРИЦАМИ

Пакет предоставляет весьма широкий набор операций для работы с массивами. Использование недопустимой операции над массивами приводит к диагностическому сообщению 'Illegal array operation' или 'Non-scalar value'. Для обеспечения ряда операций над матрицами пакет использует их специальные представления. Алгоритм вычисления определителя квадратной матрицы M и ее обращения базируется на специальном представлении $M = L \cdot U$, где L и U —

соответственно нижняя и верхняя треугольные матрицы. Такое LU-представление позволяет решать систему линейных уравнений $MX = Y$ путем простых подстановок элементов из L , U и Y . Определитель матрицы M представляет собой произведение диагональных элементов матрицы U из LU-представления. Указанный подход позволил существенно упростить алгоритмы реализации многих матричных операций. Наряду с широким набором векторных и матричных операций пакет располагает целым рядом встроенных функций для работы с данными объектами, предоставляя, в свою очередь, пользователю возможность определять и свои собственные функции векторного и матричного аргументов. В табл. 11 приведен полный перечень операций над векторными и матричными выражениями с правилами их кодирования, представления и описанием. В качестве обозначений конструкций в табл. 11 используются: S , V , M и N — соответственно скалярное, векторное, матричное и целочисленное выражения.

Таблица 11

Представление	Кодирование	Операция и ее назначение
$S \cdot \{V M\}$	$S \times \{V M\}$	Умножение скаляра на вектор или матрицу
$\frac{\{V M\}}{S}$	$\{V M\} / S$	Деление вектора или матрицы на скаляр
$\{V1 M1\} \cdot \{V2 M2\}$	$\{V1 M1\} \times \{V2 M2\}$	Умножение вектора на матрицу, матрицы на матрицу; скалярное умножение векторов возвращает скаляр
$\{V M\} \pm S$	$\{V M\} \pm S$	Сложение (вычитание) вектора и матрицы со скаляром
$\{V1 \pm V2 M1 \pm M2\}$	$\{V1 \pm V2 M1 \pm M2\}$	Сложение. (вычитание) векторов и матриц
$\{-V -M\}$	$\{-V -M\}$	Отрицание вектора или матрицы (умножение на -1)
M^N	M^N	Степень матрицы, при $N = -1$ — обращение матрицы
$\begin{Bmatrix} V \\ M \end{Bmatrix}$	$\begin{Bmatrix} V \\ M \end{Bmatrix}$	Норма вектора — скаляр
$\{V^T M^T\}$	$\{V M\}' \text{Alt} + (I)'$	Определитель матрицы
$V1 \times V2$	$V1' \text{Alt} + (\times)' V2$	Транспонирование вектора или матрицы
$\{\bar{V} \bar{M}\}$	$\{V'' M''\}$	Векторное умножение трехэлементных векторов
ΣV	$* \text{Alt} + (\$)' V$	Сопряжение элементов вектора или матрицы
$\{\vec{V} \vec{M}\}$	$\{V M\}' \text{Alt} + (-)'$	Векторная сумма; возвращает сумму элементов вектора
$M^{(N)}$	$M' \text{Alt} + ()' N$	Векторизация; поэлементное выполнение операций над массивами
V_N	$V [N]$	Возвращает в качестве вектора N -й столбец матрицы
$M_{N1, N2}$	$M [(N1, N2)]$	Возвращает элемент вектора
		Возвращает элемент матрицы

В табл. 11 во второй графе конструкция вида 'Alt + (X)' определяет необходимость одновременного нажатия клавиш 'Alt' и 'X' для ввода соответствующей операции, где $X = \{ | \mid * | \$ | - | \wedge \}$, тогда как символ '1' определяет альтернативность выражений. Приведенные в табл. 11 векторные и матричные операции соответствуют общепринятым операциям линейной алгебры и особых пояснений не требуют. В качестве примера приведем фрагмент документа, иллюстрирующий использование всех перечисленных в табл. 11 операций над массивами:

[H]

1. Определения переменных, векторов и матриц

$$N := 1 \dots 10 \quad Z := 0,3, 0,5 \dots 5 \quad S := 19,42$$

$$V1 := \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \quad V2 := \begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix} \quad M1 := \begin{bmatrix} 1 & 0,3 & 0,4 \\ 0,1 & 1 & 0,2 \\ 0 & 0,6 & 1 \end{bmatrix} \quad M2 := \begin{bmatrix} 2 & 1,1 & 0,5 \\ 1,2 & 2 & 0,1 \\ 1 & 2 & 2 \end{bmatrix}$$

$$R := \begin{bmatrix} \int_0^1 X dX \frac{d}{dS} \ln(S) \\ \sum_N \prod_1 \end{bmatrix} \quad MK := \begin{bmatrix} 1+5i & 0,42 \\ 0,47 & 8-3i \end{bmatrix}$$

2. Иллюстрация векторных и матричных операций

$$\begin{aligned} S \cdot V1 &= \begin{bmatrix} 0 \\ 19,42 \\ 38,84 \end{bmatrix} \quad S \cdot M1 = \begin{bmatrix} 19,42 & 5,826 & 7,768 \\ 1,942 & 19,42 & 3,884 \\ 0 & 11,652 & 19,42 \end{bmatrix} \quad |M1| = 0,874 \\ & \quad |V2| = 7,071 \quad V1 \cdot V2 = 14 \\ M1 \cdot M2 &= \begin{bmatrix} 2,76 & 4,1 & 1,33 \\ 1,6 & 3,31 & 0,55 \\ 1,72 & 7,2 & 2,06 \end{bmatrix} \quad M1 \cdot V2 = \begin{bmatrix} 6,2 \\ 5,3 \\ 7,4 \end{bmatrix} \quad V1 + S = \begin{bmatrix} 19,42 \\ 20,42 \\ 21,42 \end{bmatrix} \\ \frac{M2}{S} &= \begin{bmatrix} 0,103 & 0,057 & 0,026 \\ 0,062 & 0,103 & 0,005 \\ 0,051 & 0,309 & 0,103 \end{bmatrix} \quad M1 + M2 = \begin{bmatrix} 3 & 1,4 & 0,9 \\ 1,3 & 3 & 0,3 \\ 1 & 6,6 & 3 \end{bmatrix} \quad |R| = -2,332 \\ & \quad V1^T = (0 \ 1 \ 2) \quad \sum V2 = 12 \\ M1 + S &= \begin{bmatrix} 20,42 & 19,72 & 19,82 \\ 19,52 & 20,42 & 19,62 \\ 19,42 & 20,02 & 20,42 \end{bmatrix} \quad M1^T = \begin{bmatrix} 1 & 0,1 & 0 \\ 0,3 & 1 & 0,6 \\ 0,4 & 0,2 & 1 \end{bmatrix} \quad M1^{(2)} = \begin{bmatrix} 0,4 \\ 0,2 \\ 1 \end{bmatrix} \\ M1 - M2 &= \begin{bmatrix} -1 & -0,8 & -0,1 \\ -1,1 & -1 & 0,1 \\ -1 & -5,4 & -1 \end{bmatrix} \quad V1 \times V2 = \begin{bmatrix} -3 \\ 6 \\ -3 \end{bmatrix} \quad MK = \begin{bmatrix} 1-5i & 0,42 \\ 0,47 & 8+3i \end{bmatrix} \\ M1^2 &= \begin{bmatrix} 1,03 & 0,84 & 0,86 \\ 0,2 & 1,15 & 0,44 \\ 0,06 & 1,2 & 1,12 \end{bmatrix} \quad (M1 \cdot M2) = \begin{bmatrix} 2 & 0,33 & 0,2 \\ 0,12 & 2 & 0,02 \\ 0 & 3,6 & 2 \end{bmatrix} \\ M1^{-1} &= \begin{bmatrix} 1,007 & -0,069 & -0,389 \\ -0,114 & 1,144 & -0,183 \\ 0,069 & -0,686 & 1,11 \end{bmatrix} \quad |MK| = 22,803 + 37i \\ & \quad V1_2 + M1_{2,2} = 3 \\ - (M1 - M2) &= \begin{bmatrix} 1 & 0,8 & 0,1 \\ 1,1 & 1 & -0,1 \\ 1 & 5,4 & 1 \end{bmatrix} \end{aligned}$$

[K]

Использование операции ранжированного суммирования (произведения) над массивами без индексов эквивалентно их p -сложению

(скалярному p -произведению). В таких конструкциях также можно использовать условные p -операторы. Следующий фрагмент документа иллюстрирует сказанное:

{H} i:=0..3 j:=0..2 k:=0..2

$$V := \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \quad M := \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix} \quad \sum_i (i \neq 2) \cdot V = \begin{bmatrix} 3 \\ 6 \\ 9 \\ 12 \end{bmatrix} \quad \sum_i V = \begin{bmatrix} 4 \\ 8 \\ 12 \\ 16 \end{bmatrix}$$

$$\sum_j \sum_k M = \begin{bmatrix} 9 & 36 & 63 \\ 18 & 45 & 72 \\ 27 & 54 & 81 \end{bmatrix} \quad \sum_j \sum_k (j \leq k) \cdot M_{j,k} \cdot (j \neq 1) = 10$$

$$k:=1..2 \quad p:=1..2 \quad \sum_i (i \neq 2) \cdot V_i = 7$$

$$M := \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} \quad \prod_k \prod_p M = \begin{bmatrix} 199 & 435 \\ 290 & 634 \end{bmatrix} \quad M^4 = \begin{bmatrix} 199 & 435 \\ 290 & 634 \end{bmatrix}$$

Вычисление суммы элементов $X_{k,j}$ массива, удовлетворяющих условию $A \leq X_{k,j} \leq B$: $A:=43 \quad B:=48$

$X := \text{READPRN}(\text{Grodno})$ Чтение массива данных из файла

$$k:=0..rows(X)-1 \quad j:=0..cols(X)-1$$

$D'(A, B, C) := \text{if}(C \geq A, \text{if}(C \leq B, 1, 0), 0)$ Функция определения принадлежности C отрезку $[A, B]$

$$\{K\} \quad R' := \sum_k \sum_j D'[A, B, X_{k,j}] \cdot X_{k,j} \quad R' = 1991$$

Наряду с перечисленными операциями пакет содержит ряд встроенных функций векторных и матричных аргументов. Так как функции этой группы, применяемые к векторному или матричному выражению, используют массив как целое (т. е. начинают нумерацию элементов согласно значениям предопределенной переменной **ORIGIN**), то при необходимости использования в работе с массивами иного начального значения для индексов следует соответственно переопределить переменную **ORIGIN** с учетом сделанных выше замечаний и рекомендаций. Полный перечень встроенных функций пакета векторного и матричного аргументов представлен в прил. 7 (группы 9 и 11). Наряду с функциями этих групп пакет располагает еще рядом различного назначения встроенных функций, использующих в качестве аргументов векторные выражения. Например, функция '**corr**(V_1, V_2)' вычисляет корреляцию по Пирсону между двумя векторами одинаковой длины. Подробнее о таких встроенных функциях речь будет идти в гл. 6.

Ввиду прозрачности смысла упомянутых функций и способов их применения отметим лишь особенности их выполнения. Функции '**min**', '**max**' отыскивают соответственно минимальный и максимальный элементы массива, при этом минимакс ищется отдельно по действительным и комплексным частям этих элементов. Две функции '**Re**', '**Im**' выделяют соответственно действительные и мнимые части элементов массива. Функция '**augment**' позволяет объединять путем

конкатенации вектора и матрицы в их произвольном сочетании при условии, что число строк у обоих аргументов одинаково. Конкатенация состоит в получении прямоугольной матрицы, представляющей собой присоединение справа к первому аргументу (матрице, вектору) второго аргумента (матрицы, вектора). Например,

$$A_1 = \begin{bmatrix} 1 & 2 & 3 & 3 \\ 2 & 5 & 7 & 4 \\ 8 & 6 & 6 & 9 \end{bmatrix} \quad B = \begin{bmatrix} 6 & 9 \\ 2 & 0 \\ 0 & 3 \end{bmatrix} \quad \text{augment}(A, B) = \begin{bmatrix} 1 & 2 & 3 & 3 & 6 & 9 \\ 2 & 5 & 7 & 4 & 2 & 0 \\ 8 & 6 & 6 & 9 & 0 & 3 \end{bmatrix}$$

Таким образом, функция 'augment' дает еще один способ изменения как размеров массивов, так и их типа. При использовании функций упорядочивания 'sort, csort, rsort' элементы массива сортируются в возрастающем порядке. При этом, если массив комплексный, то сортировка осуществляется только по действительной, если же чисто мнимый — то по мнимой части его элементов.

При использовании векторного аргумента следует иметь в виду, что пакет в качестве его требует именно вектор-столбец. Попытка использования вектора-строки приводит к ошибке с диагностикой 'Must be vector', поэтому перед использованием вектора-строки его следует транспонировать. Приведем фрагмент документа, иллюстрирующий использование встроенных функций пакета векторного и матричного аргументов:

[H] 1. Определения векторов и матриц: $N := 3$

$$M1 := \begin{bmatrix} 1 & 0.5 & 0.2 \\ 1.1 & 1 & 1.4 \\ 2 & 3 & 1 \end{bmatrix} \quad M2 := \begin{bmatrix} 1 + 0.5i & 3 & 2 + 1.5i \\ 5i & i & 6 + 4i \\ 12 + 2i & 6 & 4 + 10i \end{bmatrix} \quad V1 := \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$V2 := \begin{bmatrix} i \\ 2i \\ 3i \end{bmatrix} \quad V3 := \begin{bmatrix} 5 \\ 6 \\ 7 \end{bmatrix}$$

$$V := (4 \ 2 + 8i \ 9i \ 1 + 2i)$$

2. Использование встроенных функций пакета

$$\text{augment}(V1^T, V2^T) = (1 \ 2 \ 3 \ i \ 2i \ 3i) \quad \text{Re}(V2^T) \cdot \text{Im}(M2) = (0 \ 0 \ 0)$$

$$(\max(M1) - \min(V1)) \cdot \text{tr}(M2) = 10 + 23i \quad V1 \cdot \text{sort}(V2) = 14i$$

$$|\text{csort}(M1, \text{last}(V2)) + \text{rsort}(M2, \text{last}(V1))| = -322.96 - 102.52i$$

$$\text{tr}[\text{augment}[\text{augment}(V1, V2), M1^{(2)}]] = 2 + 2i$$

$$\text{sort}(V^T)^T = (9i \ 1 + 2i \ 2 + 8i \ 4)$$

$$\frac{\text{rows}(M1) + \text{cols}(M2)}{\text{length}(V3) + \text{last}(V1)} \cdot \text{mean}(V2) = 2.4i$$

$$[\text{K}] \quad \text{augment}(V1, V2) = \begin{bmatrix} 1 & i \\ 2 & 2i \\ 3 & 3i \end{bmatrix} \quad \text{reverse}(V1) \times V2 = \begin{bmatrix} 4i \\ -8i \\ 4i \end{bmatrix}$$

Дополнительно к встроенным пакет дает возможность определять и пользовательские функции, использующие в качестве аргументов векторные и матричные выражения. Определяются они подобно обычным пользовательским функциям. При этом определения их могут носить как локальный, так и глобальный характер. Приведем

в качестве примера фрагмент документа с пользовательскими функциями:

[H] 1. Определение пользовательских функций

Soln (A, B): = $A^{-1} \cdot B$ Возвращает вектор-решение системы линейных уравнений вида $AX = B$

Simtr (A): = $\text{tr} (A \cdot A^T)$ Возвращает след симметрической матрицы, образованной из матрицы A

Comb (M1, M2): = $\text{Re} (M1) + \text{Im} (M2)$ Возвращает матрицу, элементы которой образованы путем сложения действительных и мнимых частей элементов аргументов

L (M, B, k, n): = $\text{augment} [M^{(k)}, B^{(n)}]$

R (M, B, k, n): = $\text{augment} [(M^T)^{(k)}, (B^T)^{(n)}]$

Z (M, B, k, n): = $\text{augment} (L (M, B, k, n), R (M, B, k, n))$

2. Определение аргументов для вычисления функций

$$A := \begin{bmatrix} 0.2 & 1 \\ 0.3 & 2 \end{bmatrix} \quad B := \begin{bmatrix} 2.42 \\ 3.47 \end{bmatrix} \quad D := \begin{bmatrix} 2i & 0.1 \\ 0.2 & 3i \end{bmatrix}$$

$$M := \begin{bmatrix} 1 & 3 & 9 \\ 2 & 4 & 5 \\ 5 & 7 & 3 \end{bmatrix} \quad S := \begin{bmatrix} 51.2 & 1 & 5 \\ 2.36 & 2 & 6 \\ 3.47 & 3 & 7 \end{bmatrix}$$

3. Вычисление функций пользователя

$$\text{Soln} (A, B) = \begin{bmatrix} 13.7 \\ -0.32 \end{bmatrix} \quad \text{Comb} (A, D) = \begin{bmatrix} 2.2 & 1 \\ 0.3 & 5 \end{bmatrix} \quad \text{Simtr} (A) = 5.13$$

$$L (M, S, 10) = \begin{bmatrix} 3 & 5.21 \\ 4 & 2.36 \\ 7 & 3.47 \end{bmatrix} \quad R (M, S, 10) = \begin{bmatrix} 2 & 51.2 \\ 4 & 1 \\ 5 & 5 \end{bmatrix}$$

$$Z (S, M, 1, 2) = \begin{bmatrix} 1 & 9 & 2.36 & 5 \\ 2 & 5 & 2 & 2 \\ 3 & 3 & 6 & 3 \end{bmatrix}$$

[K]

Группы определенных глобально пользовательских функций можно оформлять в виде файлов-документов (библиотек функций) и по мере необходимости командой 'Append' дополнять ими текущий документ. Простота и наглядность представления векторных и матричных выражений в пакете сочетается с их преимуществами в вычислительном алгоритме, так как на вычисление такого типа выражений требуется в ряде случаев значительно меньше времени относительно их ранжированных аналогов. Так, например:

$$k := 0 \dots 100 \quad A_k := k \quad B_k := k$$

$$\sum_k A_k \cdot B_k = 3.384 \cdot 10^5 \quad (8.2 \text{ c})$$

$$A \cdot B = 3.384 \cdot 10^5 \quad (3 \text{ c})$$

оценка времени вычисления произведения двух векторов в среде пакета на основе p -конструкции и векторного произведения показывают более, чем в 2,7 раза большую эффективность второго подхода к организации вычисления. Для более сложных конструкций данная разница будет значительно существеннее.

4.3. ПАРАЛЛЕЛЬНЫЕ ОПРЕДЕЛЕНИЯ И ВЫЧИСЛЕНИЯ. ОПРЕДЕЛЕНИЕ НЕТРАДИЦИОННЫХ ОПЕРАЦИЙ НАД МАССИВАМИ

Пакет дает возможность использовать вектора и матрицы для создания параллельных определений, задавая в них сразу несколько переменных. В отличие от обычного параллельное определение в своей левой части содержит вектор или матрицу имен переменных, являющихся простыми или ранжированными, тогда как в правой — векторное или матричное выражение, результатом которого является массив того же типа и размера, что и в левой части. Его элементами являются корректные выражения. Пакет приписывает каждой переменной левой части соответствующее ей выражение — элемент результата правой части. Параллельное определение эквивалентно нескольким обычным определениям, но в отличие от них производится одновременно, поэтому переменные левой части нельзя использовать в правой части определения. Параллельные определения особенно полезны при организации итеративных вычислений, рассмотренных в разделе 2.6. Приведем фрагмент документа, иллюстрирующий использование такого типа определений:

[H] $X := 5 \quad N := 1$ $M := \begin{bmatrix} 0.2 & 0.3 \\ 0.4 & 0.5 \end{bmatrix}$

$$\begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} := \begin{bmatrix} \sin(X) + \ln(X) \\ \text{tr}(M) \cdot \max(M) \\ \text{mean}(M^{(N)}) \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} 0.651 \\ 0.35 \\ 0.4 \end{bmatrix} \quad X := 1 \dots 3$$

$$\begin{bmatrix} X_{1X} \\ X_{2X} \\ X_{3X} \end{bmatrix} := \begin{bmatrix} \sin(X) + \ln(X) \\ \text{tr}(M) \cdot \max(M) \\ \text{mean}(M^{(N)}) \end{bmatrix} \quad X_1 = \begin{bmatrix} 0.841 \\ 1.602 \\ 1.24 \end{bmatrix} \quad X_2 = \begin{bmatrix} 0.35 \\ 0.35 \\ 0.35 \end{bmatrix} \quad X_3 = \begin{bmatrix} 0.4 \\ 0.4 \\ 0.4 \end{bmatrix}$$

[K] Конец фрагмента документа

Любое вычисление отдельных выражений пакет позволяет произвести параллельно на основе их векторного или матричного представления. Для этих целей служит оператор **векторизации**, вводимый по клавишам 'Alt + (—)[минус]' и требующий рассмотрения p -выражения как вектора или матрицы, а операций над ними — как векторные или матричные. В целом ряде случаев **векторизация** операций над p -выражениями позволяет сократить время вычислений в 5—7 раз по сравнению с эквивалентными им обычными вычислениями. Для векторизации выражения следует поместить курсор на оператор высшего порядка для данного выражения и нажать клавиши 'Alt + (—)'. Если такой оператор найти затруднительно (например, для сложного выражения), то можно поступить следующим образом: взять векторизуемое выражение в скобки и, поместив курсор на любую из этих скобок, нажать указанные клавиши векторизации. В результате данной процедуры пакет поместит над векторным выражением направленную вправо стрелку-вектор. Для векторизации функции курсор следует поместить на ее имя, а затем нажать клавиши векторизации. Многократная векторизация одного и того же выражения эквивалентна ее однократному применению и особого смысла (кроме, быть может, оформительского) не имеет.

Наряду с удобством представления и более совершенной организацией вычислений операция векторизации позволяет существенно расширять смысл операций и функций, к которым она применяется. Так векторизация позволяет (не нарушая естественного хода вычислений) использовать вектора и матрицы там, где в обычных условиях допускаются только скалярные выражения. Например, если \overrightarrow{M} — массив, то $\overrightarrow{\sin(M)}$, $\overrightarrow{\sqrt{M}}$ и $\overrightarrow{\ln(M)}$ будут в среде пакета полагаться корректными выражениями, состоящими в поэлементном преобразовании массива M согласно соответственно функциям $\sin(x)$, \sqrt{x} и $\ln(x)$. Приведем фрагмент документа, иллюстрирующий использование встроенных функций пакета в новом качестве и вычисление выражения при различных значениях констант и переменных:

$$\begin{aligned}
 \text{[H]} \quad \overrightarrow{M} &:= \begin{bmatrix} 1.5 & 3.9 \\ 5.8 & 8.3 \end{bmatrix} \quad \overrightarrow{A} := \begin{bmatrix} 0.1 \\ 0.2 \\ 0.3 \end{bmatrix} \quad \overrightarrow{B} := \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad \overrightarrow{C} := \begin{bmatrix} 0.4 \\ 0.5 \\ 0.6 \end{bmatrix} \quad \overrightarrow{X} := \begin{bmatrix} 19.86 \\ 19.88 \\ 19.90 \end{bmatrix} \\
 \overrightarrow{Z} &:= \overrightarrow{[A^3 \cdot \sin(X) + A \cdot B^2 \cdot \ln(X) + \sqrt{C^3 \cdot X} - \sqrt{\sqrt{A} \cdot B \cdot C}]} \\
 \overrightarrow{\sin(M)} &= \begin{bmatrix} 0.997 & -0.688 \\ -0.465 & 0.902 \end{bmatrix} \quad \overrightarrow{[\sqrt{M}]} = \begin{bmatrix} 1.225 & 1.975 \\ 2.408 & 2.881 \end{bmatrix} \\
 \overrightarrow{\ln(M)} &= \begin{bmatrix} 0.405 & 1.361 \\ 1.758 & 2.116 \end{bmatrix} \quad \overrightarrow{Z} = \begin{bmatrix} 0.98 \\ 3.306 \\ 9.314 \end{bmatrix} \quad \overrightarrow{(A \cdot B)} = \begin{bmatrix} 0.1 \\ 0.4 \\ 0.9 \end{bmatrix} \\
 \text{[K]}
 \end{aligned}$$

Приведенный фрагмент иллюстрирует также применение операции векторизации в векторному произведению $\overrightarrow{(A \cdot B)}$, которое в отличие от традиционного результата дает новый вектор, чьи элементы есть $A_k \cdot B_k$ ($k = \overline{0, n}$). Данное свойство операции векторизации дает возможность расширять применение векторов и матриц для скалярных вычислений, а также определять интересные нетрадиционные операции над векторами и матрицами. Следующий фрагмент документа иллюстрирует не только новый контекст традиционных матричных операций в условиях их векторизации, но и предупреждает об осмозрительном использовании операции векторизации:

$$\begin{aligned}
 \text{[H]} \quad i &:= 0 \dots 2 \quad j := 0 \dots 2 \quad A_{i,j} := i + j \quad B_{i,j} := i \cdot j + 1 \\
 \overrightarrow{A} &= \begin{bmatrix} 0 & 1 & 2 \\ 1 & 2 & 3 \\ 1 & 3 & 4 \end{bmatrix} \quad \overrightarrow{B} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 3 & 5 \end{bmatrix} \quad \overrightarrow{A \cdot B} = \begin{bmatrix} 3 & 8 & 13 \\ 6 & 14 & 22 \\ 9 & 20 & 31 \end{bmatrix} \quad \overrightarrow{(A \cdot B)} = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 4 & 9 \\ 2 & 9 & 20 \end{bmatrix} \\
 \overrightarrow{\begin{bmatrix} A^{(2)} \\ B^{(2)} \end{bmatrix}} &= \begin{bmatrix} 2 \\ 1 \\ 0.8 \end{bmatrix} \quad \overrightarrow{\begin{bmatrix} A \\ B \end{bmatrix}} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 1 \\ 2 & 1 & 0.8 \end{bmatrix} \quad \overrightarrow{\begin{bmatrix} A \\ B \end{bmatrix}} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 1 \\ 2 & 1 & 0.8 \end{bmatrix} \\
 \overrightarrow{[(A)^B]} &= \begin{bmatrix} 0 & 1 & 2 \\ 1 & 4 & 27 \\ 2 & 27 & 1024 \end{bmatrix} \quad \begin{aligned} X &:= 0.42 \\ Y &:= 0.47 \\ Z &:= 0.67 \quad T := 0.88 \end{aligned}
 \end{aligned}$$

$$M := \begin{bmatrix} X & Z \\ Y & T \end{bmatrix} \xrightarrow{[M^M]} \begin{bmatrix} 0.695 & 0.765 \\ 0.701 & 0.894 \end{bmatrix}$$

$$[K] \xrightarrow{[e^M]} \xrightarrow{[M^{\ln(M^T)}]} \cdot \begin{bmatrix} M \\ M^T \end{bmatrix} = \begin{bmatrix} 5.964 & 8.502 \\ 5.357 & 7.637 \end{bmatrix}$$

Таким образом, не взирая на целый ряд преимуществ операции векторизации, использовать ее следует весьма осмотрительно, так как можно получить неверные результаты. Действительно, векторизация, базируясь на поэлементной обработке, не только расширяет область применения традиционных операций и функций, но в ряде случаев изменяет их смысл (произведения матриц и векторов и т. д.) и даже делает формально допустимыми операции (деление векторов и матриц), не имеющие смысла в обычной линейной алгебре. Поэтому там, где смысл операций (и в первую очередь над массивами) не должен меняться, векторизацию использовать не следует. С другой стороны, векторизация отдельной переменной не изменяет ее смысл, что позволяет применять ее для внешнего оформления векторных вычислений. Векторизация выражений предполагает использование в них массивов одинакового размера, так как векторизованные операции над массивами производятся строго поэлементно. Так векторизованное выражение $\overrightarrow{[M^n]}$ в результате дает не n -ю степень матрицы M , а матрицу, чьи элементы являются n -й степенью соответствующих элементов исходной матрицы. Сказанное еще раз подтверждает необходимость корректного использования операции векторизации особенно в тех выражениях, которые содержат операции над массивами. Ряд других интересных примеров на использование операции векторизации будет рассмотрен в следующих главах книги.

Глава 5

РЕШЕНИЕ УРАВНЕНИЙ, СИСТЕМ УРАВНЕНИЙ И НЕРАВЕНСТВ

Наряду с традиционной задачей нахождения нулевых корней уравнений пакет позволяет решать как традиционные системы уравнений, так и системы неравенств, а также системы уравнений при заданных ограничениях. Данные вопросы и рассматриваются детально в настоящей главе.

5.1. ВЫЧИСЛЕНИЕ НУЛЕВЫХ КОРНЕЙ УРАВНЕНИЙ

Для вычисления нулевого корня уравнения пакет располагает встроенной функцией 'root', имеющей следующий формат

root (<Выражение>, <Переменная>)

Аргумент 'Выражение' функции определяет левую часть уравнения вида $F(X) = 0$, нулевой корень которого следует вычислить, тогда как 'Переменная' указывает имя ведущей переменной уравнения, т. е. переменной, относительно которой разрешается уравнение.

Перед использованием функции 'root' само уравнение должно быть приведено к виду $F(X) = 0$. В качестве первого аргумента функции 'root' следует использовать левую часть уравнения, т. е. $F(X)$. Второй ее аргумент должен быть определен. Пакет использует его значение (X_0) в качестве первого приближения при вычислении его левого корня. При вычислении действительного корня значение определяется действительным, а при вычислении комплексного — комплексным. Результатом вычисления согласно функции 'root' является действительное или комплексное число, полагающееся нулевым корнем уравнения $F(X) = 0$. Приближенное вычисление корня уравнения производится методом секущих с точностью, определяемой значением предопределенной переменной пакета TOL (по умолчанию $TOL = 0.001$).

Пакет использует значение X_0 в качестве начальной точки для организации последовательных приближений к искомому нулевому корню X_0^* уравнения, прекращая вычисление при соблюдении условия $|F(X_0^*)| < TOL$. При превышении определенного количества итераций без достижения данного условия пакет прекращает вычисление и идентифицирует функцию 'root' диагностическим сообщением 'Not converging'. Основными причинами появления данного сообщения являются следующие:

- отсутствие у функции $F(X)$ нулевых корней данного типа;
- некорректное определение первого приближения X_0 корня;
- между первым приближением и истинным нулевым корнем существует локальный минимакс или точка разрыва функции $F(X)$.

Скорость сходимости итераций к искомому значению нулевого корня прямо пропорциональна величине $A = |X_0 - X_0^*|$ и обратно пропорциональна величине переменной TOL. В связи с этим перед использованием функции 'root' в случае сложного поведения левой части уравнения $F(X) = 0$ рекомендуется стандартными методами математического анализа исследовать функцию $Y = F(X)$ на предмет определения числа нулевых корней, их типов и приближенного распределения их по оси X . В простейшем случае для этих целей рекомендуется использовать графическое представление функции $Y = F(X)$, получаемое тут же в документе средствами самого пакета согласно описанию в разделе 2.8. Для иллюстрации сказанного приведем фрагмент документа:

```
[H] X:=1+i L(X):=X^2+X+1 root(L(X), X)=-0.5+0.866i
F(X):=sin(X)*(X+3.47)+1.942 ORIGIN:=1 TOL:=0.00001
X:=-10,-9.9..10
```

Рис. 10

```
X1:=-10 X2:=-6 X3:=-2 X4:=2 X5:=6 X6:=10
ROOT(X):=root(F(X), X) Y:=X^T
[K] ROOT(Y)=(-9.74 -5.065 -1.526 6.078 9.574)
```

В приведенном фрагменте посредством функции 'root' вычисляется комплексный нулевой корень квадратного трехчлена $L(X)$, а затем

производится вычисление нулевых корней функции $F(X)$ на интервале $[-10, 10]$ с точностью $E = 10^{-5}$. Для этого переопределяются соответствующим образом переменные пакета ORIGIN и TOL, а переменная X ранжируется в интервале $[-10, 10]$ с шагом $h = 0.1$. После этого создается по клавише 'd' (эта) графическая область. Для большей наглядности посредством использования локального графического формата (клавиша {F | f} с курсором в области графика: Size = 20, 40) размер ее устанавливается 20×40 . Средние указатели осей X и Y получают соответственно значения X и $F(X)$, нижний указатель оси Y получает нулевое значение, а для остальных указателей области пакет вычисляет значения автоматически, после чего выводится график самой функции $F(X)$. По данному графику несложно, зная масштаб, оценить количество действительных корней функции на заданном отрезке и примерные начальные приближения для каждого из них. Следует отметить, что такой графический подход может быть использован для грубого решения уравнений, если графическая область имеет максимально допустимые размеры, а в качестве устройства вывода используется плоттер. Возвращаясь снова к фрагменту документа, на основе анализа графика функции $F(X)$ определяем вектор-столбец X первых приближений к искомым корням уравнения и на основе его транспозиции находим эквивалентный ему вектор-строку Y. После этого определяем $ROOT(X)$ как функцию искомых корней уравнения и, наконец, векторируя функцию $ROOT(Y)$ на векторе Y начальных приближений, получаем вектор-строку $ROOT(Y)$ искомых нулевых корней.

При использовании функции 'root' следует иметь в виду следующее. Для повышения точности вычислений, а также при наличии близлежащих корней уравнения следует соответственно переопределять в документе переменную TOL, например, $TOL := 0.00001$. Однако значительное уменьшение значения TOL может привести к существенным временным затратам, а в ряде случаев — и к сообщению 'Not converging'. Соображения по вычислению множественных нулевых корней уравнения приводились выше. Если в окрестности нулевого корня X_0^* функция $Y = F(X)$ имеет малую кривизну, то функция $root(F(X), X)$ пакета может сходиться к значению X_0^* , которое расположено относительно далеко от фактического корня X_0 . В этом случае даже при выполнении условия $|F(X_0^*)| < TOL$ величина X_0^* может быть далека от точки X_0 , в которой $F(X_0) = 0$. В таком случае рекомендуется либо уменьшать значение переменной TOL, либо пытаться вычислять значение корня посредством следующей функции:

$$root \left[\frac{F(X)}{\frac{d}{dX} F(X)}, X \right].$$

Если уже известен один нулевой корень $X = A$ уравнения $F(X) = 0$, то второй можно вычислять посредством функции

$$root \left[\frac{F(X)}{(X-A)}, X \right],$$

т. е. применять метод отделения корней. Данный подход при вычислении близлежащих корней может оказаться более эффективным, чем определение для переменной TOL меньшего значения. Приведем фрагмент документа, использующего оба способа вычисления нулевых корней уравнения:

$$[N] \quad F(X) := 7 \cdot X^5 + X^4 + 3 \quad Y(X) := (X + 5) \cdot (X \cdot \sin(X) + 19.42) \quad X := 0$$

$$\text{root} \left[\frac{F(X)}{\frac{d}{dX} F(X)}, X \right] = -0.875 \quad \text{root} \left[\frac{Y(X)}{X + 5}, X \right] = 29.008$$

Вычисление действительных корней уравнения $F(X) = 0$ на заданном интервале при условии, что разность их значений не меньше величины A:

$$\text{Min} := 0 \quad A := 1 \quad \text{Max} := 5$$

$$n := \text{Min}, \quad \text{Min} + A .. \text{Max} - 1$$

$$F(X) := X^X + 2 \cdot X - 5$$

$$D(n) := F(n) \cdot F(n + 1)$$

$$R(n) := \text{if } (D(n) > 0, i, \text{ if } (D(n) < 0, \text{root}(F(n), n), \text{ if } (F(n) \approx 0, n, n + 1)))$$

n	n+1	R(n)	F(X) := sin(X)
0	1	i	R(n)
1	2	1.53448887	0
2	3	i	i
3	4	i	i
4	5	i	3.1415895
			i

[K] Символ «i» идентифицирует отсутствие действительного корня уравнения $F(X) = 0$ на интервале $[n, n + 1]$.

На основе встроенной функции 'root' пакета может быть определена функция пользователя следующим образом:

$$\langle \text{Имя} \rangle (\langle \text{Список переменных} \rangle) \{ i := | \equiv \}$$

$$\text{root} (\langle \text{Выражение} \rangle, \langle \text{Переменная} \rangle)$$

где 'Имя' задает любое допустимое имя функции, а 'Список переменных' может включать через запятую имена переменных из аргумента 'Выражение', включая и само имя ведущей переменной 'Выражения'. В случае глобального определения функции пользователя глобально должно быть определено и 'Выражение', в противном случае оно идентифицируется пакетом сообщением 'Undefined'. При этом, если список переменных включает ведущую переменную, то этот аргумент специфицирует первое приближение для функции 'root', используемой в правой части определения. В противном случае определение функции пользователя использует любое первое приближение, имеющееся в документе выше его. Результатом таким образом определенной функции пользователя является вычисленное значение функции 'root' на заданных аргументах 'Выражение' и 'Переменная'. Приведем фрагмент документа, в котором вычисляются оба корня квадратного трехчлена с учетом табуляции его коэффициентов, а также собственные значения матрицы:

[H] $A := (1 \ 2 \ 3) \quad B := (-7 \ -12 \ -10) \quad C := (6 \ 7 \ 8)$
 $F(A, B, C, X) := A \cdot X^2 + B \cdot X + C \quad QE(A, B, C, X) := \text{root}(F(A, B, C, X), X)$

$$QE1(A, B, C, X) := \text{root}\left[\frac{F(A, B, C, X)}{X - QE(A, B, C, X)}, X\right]$$

$$\overrightarrow{QE(A, B, C, X)} = (1 \ 0.655 \ 1.333) \quad \overrightarrow{QE1(A, B, C, X)} = (6 \ 5.345 \ 2)$$

Вычисление собственных значений матрицы:

$$[K] \quad A := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \quad \begin{array}{l} \lambda := 1 \\ \lambda_0 := \text{root}(|A - \lambda \cdot \text{identity}(\text{rows}(A))|, \lambda) \\ \lambda_0 = 3.39678 \cdot 10^{-8} \end{array}$$

В приведенном примере определяются на основе функции 'root' две функции пользователя 'QE' и 'QE1' соответственно для вычисления первого и второго (методом отделения) корней уравнения $F(A, B, C, X) = 0$. Средство векторизации данных функций позволяет решать задачу на векторах коэффициентов A, B, C , что предоставляет наряду с повышением эффективности вычислений компактную и прозрачную запись алгоритма решения в векторной форме. Наряду с векторным подходом к решению подобных задач можно использовать и подход на основе p -вычислений, но он представляется нам менее удобным и требует, порой, значительно больших временных затрат. В конце примера вычисляется действительное собственное значение квадратной матрицы.

5.2. ОПРЕДЕЛЕНИЕ РЕШАЮЩИХ БЛОКОВ В СРЕДЕ ПАКЕТА

Для решения системы уравнений и/или неравенств в среде пакета MathCAD используется специальная конструкция, называемая **решающим блоком** или просто **блоком**. Там, где это не вызывает необходимости уточнения, будем использовать просто термин «блок», понимая под ним решающий блок. Блок состоит из *заголовка блока* (ключевое слово GIVEN), его *тела* (определения, система уравнений и/или неравенств) и *конца блока* (функция FIND или MINERR). При кодировании зарезервированных пакетом слов FIND, GIVEN и MINERR допускается произвольное сочетание прописных и строчных букв, т. е. данные слова не являются буквозависимыми.

Блок всегда открывается ключевым словом GIVEN, за которым следуют необходимые определения, уравнения и/или неравенства, составляющие его тело. Блок не имеет ограничения по горизонтали, а по вертикали все строки, ниже строки, содержащей слово GIVEN, и выше строки с функцией FIND или MINERR, относятся к телу блока. Концом блока является первое допустимое выражение, содержащее функцию FIND или MINERR, которые имеют следующий формат:

FIND ((Список ведущих переменных))
MINERR ((Список ведущих переменных))

где 'Список переменных' включает перечень переменных блока,

относительно которых должна быть разрешена входящая в его тело система уравнений и/или неравенств. Такие переменные решающего блока будем называть **ведущими**. При этом, если любая из этих функций имеет один аргумент, то она возвращает единственное значение, являющееся решением блока. В противном случае возвращается решающий вектор-столбец значений с количеством элементов, равным количеству элементов 'Списка ведущих переменных' функции завершения блока. Отличие функций FIND и MINERR состоит в том, что первая пытается получить приближенное решение блока, тогда как вторая вместо решения возвращает значения ведущих переменных, минимизирующих величину $|ERR|$ вектора отклонения от точного решения. При наличии у блока решения значения и смысл функций FIND и MINERR совпадают.

Как ведущие, так и другие переменные блока должны быть предварительно определены до начала блока. Пакет использует значения ведущих переменных в качестве первых приближений при численном решении блока. Переменные можно определять и внутри самого блока, но только глобально. Локально определенные переменные внутри блока не имеют смысла вне его. При создании и редактировании блока рекомендуется для удобства и сокращения временных задержек предварительно переводить пакет командой 'Manual' в М-режим, а после проведения данной работы по команде 'Auto' возвращать в основной А-режим, либо пользоваться клавишей 'F9' (Calc).

Тело блока (кроме необходимых определений) содержит по крайней мере одно условие, имеющее следующий формат:

(Выражение 1) (Операция отношения) (Выражение 2)

В отличие от обычных выражений условие может содержать выражения по обе стороны от операции отношения и должно быть скалярным и действительным по обе стороны от нее. В качестве условия может быть и единственное выражение, например, уравнение от одной переменной. При решении блока пакет ищет значение, делающее такое условие истинным. Допустимые основные операции отношения для условий блока приведены в табл. 12.

Приведенные в табл. 12 операции отношения достаточно прозрачны и особых пояснений не требуют, за исключением операции (\neq) неравенства. Пакет не допускает использования данной операции непосредственно для связи обеих частей условий в теле блока, а при попытке сделать это идентифицирует ситуацию 'Illegal context'. В принципе, такое ограничение не является существенным, так как условие $X \neq A$ можно заменить эквивалентным ему условием $|X - A| > 0$. Наряду с этим, операцию неравенства можно использовать, например,

Таблица 12

Символ операции		Смысл операции отношения
Кодирование	Изображение	

Alt + '='	\approx	Равенство
>	$>$	Больше, чем
<	$<$	Меньше, чем
Alt + ')'	$>$	Не меньше, чем
Alt + '($<$	Не больше, чем
Alt + '#'	\neq	Не равно

в любой части условия блока `eo` встроенной функцией `'if'` пакета. Следующий простой фрагмент документа иллюстрирует сказанное:

[H] `X:=0 Y:=0 A:=1`

`Given $X + Y \approx 1$ $X^2 + Y^2 \approx$ if ($A \neq 8, 4, A$)`

`R(A):=FIND(X, Y) R(A)T=(1.823 -0.823)`

[K] `R(8)T=(2.436 -1.436)`

Операция равенства определяет обычное в математическом смысле равенство, а все остальные операции определяют обычные неравенства. При решении блока пакет пытается найти такие значения ведущих переменных, которые при подстановке во все условия блока делали бы их одновременно истинными.

Пакет решает блок с помощью модифицированного метода Левенберга—Маркварта [21] в области действительных чисел. При попытке решить блок для комплексных значений переменных пакет рассматривает их действительные и мнимые части как отдельные переменные и из каждого условия создает пару действительных условий соответственно для действительной и мнимой частей. При этом, если все первые приближения ведущих переменных заданы действительными значениями, то пакет ищет действительное решение блока, в противном случае — комплексное.

Используемый пакетом метод не позволяет решать блок, содержащий условий меньше, чем ведущих переменных. В этом случае пакет помечает решающий блок диагностическим сообщением `'Too few constraints'`. Для устранения данной ситуации блок рекомендуется пополнять фиктивными условиями или уменьшать количество ведущих переменных. Пакет прекращает решение блока после достижения определенного критерия (нахождение решения блока или выход на особую или аварийную ситуацию) [21], проверяет значение переменной `ERR` пакета и возвращает результат решения блока в следующем виде:

искомые значения ведущих переменных блока, если $|ERR| \leq TOL$;

функция `FIND`, помеченная сообщением `'Did not find solution'`, если $|ERR| > TOL$ (решение блока не найдено);

искомые значения ведущих переменных, если блок заканчивается функцией `MINERR`.

В любом случае после завершения решения блока пакет устанавливает значение переменной `ERR` пакета (имя переменной буквозависимо), равное величине вектора отклонения (сумма квадратов отклонений по всем условиям блока). В этом плане функция `ERR` пакета может служить для определения погрешности полученного решения блока. При этом значение переменной `ERR` (для функции `MINERR` конца блока) сохраняется до следующего использования функции `MINERR` в документе. Точность же полученного решения блока определяется значением предопределенной переменной `TOL` пакета, которую при необходимости можно переопределить. Более того, на количество итераций при решении блока имеется ограничение, при превышении которого без возврата решения пакет прекращает вычисления и отмечает функцию `FIND` или `MINERR` конца блока диагностическим сообщением `'Not converging'` и полагает

ERR = 0. Как правило, это бывает в случае неразрешимости блока. Для иллюстрации основных моментов сказанного приведем фрагмент документа, содержащий простые примеры определения и решения блоков:

$$[H] \text{ Given } X \equiv 0 \quad X^2 - 3 \cdot X + 1 \approx 0 \quad \text{Find}(X) = 0.382$$

$$\text{GIVEN } Y \equiv 1 \quad X \equiv 8 \quad X \cdot Y < 0 \quad X^3 + Y^3 > 4$$

$$\text{FIND}(X, Y) = \begin{bmatrix} 8 \\ 0 \end{bmatrix}$$

$$X \equiv 6 \quad \text{given } Y \equiv 7 \quad Z \equiv 1 + 4i \quad |Y \cdot Z| > X$$

$$|\ln(Z) \cdot \cos(Y)| > X \cdot Y$$

$$\text{FIND}(Y, Z) = \begin{bmatrix} 0.452 \\ 1 + 13.6i \end{bmatrix}$$

$$\text{Given } Y \equiv 1 \quad X \equiv 8 \quad Z \equiv \cos(Y) \cdot e^X$$

$$X^2 + Y^2 > (X \cdot Y)^X \quad \frac{Z}{X} \approx 19.47$$

$$\sqrt{X + Y + Z} \cdot \int_Y^X X dX < \frac{d}{dY} (\ln(Y) + \sin(Y))$$

$$\text{Find}(X, Y, Z) = \begin{bmatrix} 1.575 \\ 3.394 \\ 30.657 \end{bmatrix}$$

$$\text{GIVEN } F(X) := X^2 \quad r(X) := X^3 \quad \sqrt{F(X) + r(X)} < 5 \quad \text{find}(X) =$$

$$\text{ERR} = 0$$

did not find solution

$$\text{Given } X \equiv 5 \quad Y \equiv 10 \quad X + Y \approx 10 \quad \text{FIND}(X, Y) = \blacksquare \blacksquare$$

$$X := 0 \quad Y := 0$$

too few constraints

$$\text{Given } 5 \cdot X - 8 \cdot Y + 7 \approx 0 \quad 2 \cdot X + 6 \cdot Y - 9 \approx 0$$

$$\text{Find}(X, Y)^T = (0.65217 \quad 1.28261) \quad X := 3 \quad Y := 2$$

$$\text{Given } 2 \cdot X^2 - X \cdot Y - 5 \cdot X + 1 \approx 0 \quad X + 3 \cdot \log(X) - Y^2 \approx 0$$

$$\text{Find}(X, Y)^T = (3.4874 \quad 2.2616) \quad X := 1 \quad Y := 1 \quad Z := -1$$

$$\text{Given } X^Y + Y^2 + Z^3 - 0.5 \cdot Z \approx 0$$

$$X \cdot \tanh\left[\frac{Y - X}{Z}\right] + Y \cdot Z + 1.25 \approx 0 \quad \sin[X + Y^2 + Z] + X \cdot Y \cdot Z + 0.8 \approx 0$$

$$\text{Find}(X, Y, Z)^T = (1.0040193 \quad 0.9436133 \quad -1.3715272)$$

$$A := 5 \quad B := 6 \quad C := 7 \quad \text{Given}$$

$$\int_0^A e^X dx \approx \int_1^B x^3 dx + \int_2^C x dx \quad A > 0 \quad B > 0 \quad C > 0$$

$$[K] \quad M := \text{Find}(A, B, C) \quad M^T = (5.24529 \quad 5.07946 \quad 7)$$

Приведенные в документе примеры решения блоков достаточно просты и с учетом сказанного особых пояснений не требуют. Ниже будут приведены более сложные примеры использования блоков.

5.3. ИСПОЛЬЗОВАНИЕ РЕШАЮЩИХ БЛОКОВ В ДОКУМЕНТЕ

Перед вводом решающего блока в документ рекомендуется перевести пакет командой 'Manual' в М-режим, чтобы по мере формирования тела блока не производились преждевременные вычисления входящих в него условий и не возникали ошибки, связанные с незавершенностью блока, а также излишние временные затраты, порой существенно снижающие эффективность работы с документом. После завершения формирования блока командой 'Auto' пакет вновь переводится в А-режим и вычисляет созданный решающий блок. Для проведения разовых вычислений в документе можно, оставаясь в М-режиме, использовать клавишу 'F9' (Вычисление). Ввод решающего блока осуществляется в следующем порядке (сверху вниз по строкам и слева направо по столбцам документа): определяются первые приближения для ведущих переменных блока, ключевое слово GIVEN, тело блока и завершающее блок выражение, содержащее функцию FIND или MINERR.

По завершающей блок функции FIND или MINERR пакет возвращает решение блока или значения переменных, минимизирующих вектор отклонения от точного решения блока. Если данные функции имеют один аргумент, то возвращается результат — число, в противном случае возвращается вектор-столбец результата. При этом следует иметь в виду, что передача документу результатов решения блока осуществляется только через функции FIND и MINERR его конца и получить эти значения, непосредственно используя имена ведущих переменных, невозможно. Прямое использование имен ведущих переменных после соответствующего им блока возвращает значения их первых приближений, определенные локально до решающего блока либо глобально в самом блоке. Следующий фрагмент документа иллюстрирует сказанное:

[H] A: = 1.42 X: = 0 Y: = 0

Given B = 1.47 $3 \cdot X + 7 \cdot Y \approx A$ $9 \cdot X - 4 \cdot Y \approx B$

[K] Z: = Find(X, Y) $\Sigma Z = 0.325$ $X + Y = 0$

Получить результат вычисления блока можно одним из шести основных способов:

1) завершить решающий блок функцией 'FIND (<список ведущих переменных>) = ' ; пакет возвращает число или вектор-столбец результата согласно 'списку ведущих переменных' ;

2) завершить решающий блок функцией 'MINERR (<список ведущих переменных>) = ' ; пакет возвращает число или вектор-столбец значений ведущих переменных, минимизирующих величину вектора отклонения ;

3) завершить решающий блок конструкцией вида 'X: = FIND (<список ведущих переменных>) ' ; пакет возвращает число или вектор-столбец X результата согласно 'списку ведущих переменных' ;

4) завершить решающий блок конструкцией вида 'Y: = MINERR (<список ведущих переменных>) ' ; пакет возвращает число или вектор-столбец Y значений ведущих переменных, минимизирующих вектор отклонения ;

- 5) завершить решающий блок пользовательской функцией вида
(Имя)((Список параметров)) = FIND ((Список ведущих переменных))

Результатом вычисления данной конструкции является определение функции пользователя, которая может использоваться в дальнейшем, передавая значения ведущих переменных, решающих блок;

6) аналогичным образом функцию пользователя можно определить на основе функции MINERR конца блока, которая может использоваться в дальнейшем, передавая значения ведущих переменных, минимизирующих вектор отклонения. При использовании способов 5) — 6) следует иметь в виду, что функция пользователя на основе решающего блока не допускает глобального определения, в противном случае она идентифицируется сообщением 'Illegal context'.

При организации завершения решающего блока следует иметь в виду, что функции FIND и MINERR его конца нельзя непосредственно использовать в выражениях и в качестве аргумента других функций, например, $\ln(\text{find}(X))$ или $\sin(X) + \text{minerr}(X, Y)$. Но в отличие от FIND использование функции MINERR в таких конструкциях не всегда вызывает аварийную ситуацию, хотя результат такого использования в общем случае некорректен. Так, использование функции MINERR(X) с одним аргументом допустимо в любом корректном выражении, но при этом функция возвращает всегда значение первого приближения для ведущей переменной блока, а не его решение. Тогда как использование функции MINERR с количеством аргументов больше двух в выражении вызывает аварийную ситуацию с диагностикой 'Too many arguments'. Ниже будут приведены примеры подобного характера. В этом отношении функции FIND и MINERR конца решающего блока несколько отличны от других типов встроенных функций, поддерживаемых пакетом. Переменная ERR пакета наряду с основным своим назначением может в ряде случаев использоваться для оценки точности решения блока в целом.

Одним из весьма полезных свойств пакета является возможность определения пользовательской функции в терминах решающего блока. Каждое использование в документе определенной таким образом функции пользователя приводит к решению соответствующего блока. Это может оказаться полезным для организации вложенных решающих блоков, работы с векторами и т. д. Список параметров пользовательской функции содержит ведущие переменные и/или другие переменные блока. При этом, если список параметров содержит ведущие переменные, то пакет использует их значения в качестве первых приближений в случае использования данной функции в документе. Для ведущих же переменных, не входящих в список параметров функции, используются первые приближения, заданные перед самим решающим блоком или определенные глобально внутри него. Такая организация конструкции блока (когда для его решения берутся только начальные значения ведущих переменных, определенные перед его первым использованием в документе) в значительной мере сужает возможности применения функций пользователя,

определенных в терминах решающего блока. Использование в списке параметров функции переменных блока, отличных от ведущих, позволяет решать блок при различных значениях этих переменных, т.е. *табулировать* решение блока по различным его параметрам. При этом следует помнить, что если ведущие переменные в блоке имеют разные размерности, то результат вычисления такого блока нельзя представить в векторном виде, либо определить для него пользовательскую функцию. В этом случае результат следует представлять как набор значений отдельных ведущих переменных блока.

Определив функцию пользователя в терминах решающего блока, можно использовать ее в векторных и ранжированных выражениях, ранжируя любой параметр функции. Однако применение ранжированных функций пользователя требует, порой, значительных временных затрат пакета, поэтому рекомендуется использовать векторный подход. При этом при векторизации функции пользователя переменная ERR пакета не определяется.

Следует иметь в виду, что функции пользователя в терминах функций FIND и MINERR конца решающего блока нельзя определять глобально, в противном случае использование таких функций приводит к ошибкам с диагностикой 'Illegal context' или 'Undefined'. Приведем фрагмент документа, содержащий примеры использования решающих блоков:

[H] 1. Решение системы уравнений: $X = 0 \quad Y = 0 \quad Z = 0$
 $\text{GIVEN } X + Y + Z \approx 1 \quad X^2 + Y^2 + Z^2 > 2 \quad X^3 + Y^3 + Z^3 < 5$

$R := \text{FIND}(X, Y, Z) \quad \text{ERR} = 1.664 \quad R^T =$

did not find solution	undefined
-----------------------	-----------

2. Решение системы неравенств: $Y = 4$

$\text{Given } X^2 + Y^2 < 25 \quad Y - X^2 - 3 \geq 0 \quad X \geq 0 \quad Y \geq 0$

$R := \text{FIND}(X, Y) \quad \text{ERR} = 0 \quad R^T = (0 \ 4)$

3. Нахождение отрицательного нулевого корня уравнения:

$\text{Given } X = 0 \quad 2 \cdot X^3 + 5 \cdot X^2 + 4 \cdot X + 10 \approx 0 \quad X \leq 0$

$\text{FIND}(X) = -2.5 \quad \text{ERR} = 1.243 \cdot 10^{-14}$

4. Использование вложенных решающих блоков:

$X := 2 \cdot \sqrt{2} \quad Y := X \quad R := 1 \quad K := 2$

$\text{Given } X^2 + Y^2 \approx R^2 \quad Y - K \cdot X \approx 0 \quad X \geq 0$

$\text{Block}(R, K) := \text{FIND}(X, Y)$

$\text{Block}(5, 5) = \begin{bmatrix} 0.981 \\ 4.903 \end{bmatrix} \quad R := 5 \quad K := R$

$\text{GIVEN } R^3 + K^3 \approx 10 \quad |\text{Block}(R, K)| \leq 25$

$\text{FIND}(R, K) = \begin{bmatrix} 8 \\ 2 \end{bmatrix} \quad \text{ERR} = 7.105 \cdot 10^{-15}$

5. Определение функции пользователя в терминах решающего блока с последующей ее табуляцией:

$X := 0 \quad A := 1 \quad B := A \quad N := 3$

$\text{Given } e^{A \cdot X} \approx B + N \cdot X$

$\text{Func}(A, B, N) := \text{FIND}(X)$

$A := (2 \ 3 \ 4) \quad B := (3 \ 4 \ 5) \quad N := (6 \ 7 \ 8)$

Func (A, B, N) = (-0.429 -0.543 -0.614) ERR =

undefined

6. Определение функции пользователя в терминах функции MINERR:

Given $e^{A \cdot X} - B \cdot X^N \approx 0$

Mfunc (A, B, N):= MINERR (X) A:=(1 2 3) B:=(4 5 6)

N:=(7 8 9) Mfunc (A, B, N) = (-0.526 -0.688 -0.564)

Вычислены точки максимального сближения по оси Y функций $Y1 = e^{AX}$ и $Y2 = BX^N$.

7. Использование функций FIND и MINERR в выражениях:

X:=0.48 Y:=0.45

GIVEN $X + Y > 2$ $X^2 + Y^2 < 19.9$

FIND (X) = 1.55

Given $X + Y > 2$ $X^2 + Y^2 < 19.9$

Find (X) + sin (X) =

Illegal context

Given $X + Y > 2$ $X^2 + Y^2 < 19.9$

Minerr (X) + sin (X) = 0.942

Given $X + Y > 2$ $X^2 + Y^2 < 19.9$

sin (Minerr (X)) = 0.462

Given $X + Y > 2$ $X^2 + Y^2 < 19.9$

Minerr (X, Y) = $\begin{bmatrix} 0.48 \\ 1.52 \end{bmatrix}$

Given $X + Y > 2$ $X^2 + Y^2 < 19.9$

Minerr (X, Y) + 2 =

too many arguments

8. Использование в условиях блока ранжированных, векторных и матричных выражений:

X:=5 Y:=9 A:=1..5 B:=(0.42 0.47 0.67 0.89)

Given

$$\cos(X) \cdot \int_X^Y X \cdot \ln(X) dX + \frac{d}{dX} \sin(X) \cdot \ln(X) \approx \prod_A \sum_A (\Sigma B) \cdot 10^{-3}$$

Z (X):= FIND (X)

$$\int_0^1 \ln(X) \cdot e^X dX + Z(X) = 3.398 \quad \text{ERR} = 3.136 \cdot 10^{-14}$$

9. Недопустимость использования глобального определения функции пользователя, определяемой в терминах функций FIND и MINERR конца решающего блока:

Given A=1 X=A

$F(X, A) \approx X^3 - A \cdot X^2 + 6 \cdot X - 15$ $F(X, A) \approx 0$

Z (A):= FIND (X) Z (3) = 2.782

Given $F(X, A) \approx 0$ Z (A) = Find (X) Z (3) =

Illegal context

Given $F(X, A) \approx 0$ $Z(A) \equiv \text{Minerr}(X)$ $Z(3) =$

Illegal context

10. Применение функции пользователя в терминах функции доступа FIND для получения всех решений системы нелинейных уравнений: $X := -0.4$ $Y := 0.5$ $Z := 0.5$

Given $X + Y + Z^2 - 0.2 \approx 0$ $|X - A| > 0$ $|Y - B| > 0$
 $|Z - C| > 0$ $X^2 + Y^2 + Z^2 - 0.7 \approx 0$ $X^2 + Y + Z - 1 \approx 0$

FR(A, B, C) := Find(X, Y, Z) F := FR(0, 0, 0)

$F^T = (-0.484 \ 0.676 \ 0.089)$ $\text{FR}[F_0, F_1, F_2]^T = (-0.42 \ 0.108 \ 0.716)$

11. Решение систем нелинейных уравнений в комплексной области:

$Z := 1 + i$ $W := 2 - 3i$

Given $Z \cdot \cos(W) - A \cdot (W + Z + 1) \approx 0$ $W \cdot \cos(Z) + B \cdot [Z^2 - W^2] \approx 0$

R(A, B) := Find(Z, W) $R(2.5)^T = (-0.67601 \ -0.60248)$

$R(1, 1)^T = (-0.981 - 2.629 \cdot 10^{-10}i \ -0.742 + 3.562 \cdot 10^{-11}i)$

12. Вычисление матрицы, определитель которой удовлетворяет заданным условиям:

A := 1 B := 1 C := 1 D := 1

Given $\left| \begin{bmatrix} A & B \\ C & D \end{bmatrix} \right| < 4$ $\left| \begin{bmatrix} A & B \\ C & D \end{bmatrix} \right| > 1$ $A + B \approx 3$
 $C \cdot D \approx 5$

R := Find(A, B, C, D) $\Phi(i, j) := 2 \cdot i + j$ $i := 0 \dots 1$

$j := 0 \dots 1$ $\text{RM}_{i,j} := G_{\Phi(i,j)}$ $|\text{RM}| = 2.58746$

$\text{RM} = \begin{bmatrix} 2.017 & 0.983 \\ 2.147 & 2.329 \end{bmatrix}$ $R_0 + R_1 = 3$ $R_2 \cdot R_3 = 5$

[K]

Ниже будет рассмотрен ряд других интересных примеров на использование решающих блоков. При наличии в документе решающего блока пакет обрабатывает находящиеся в нем конструкции независимо от конструкций в других областях документа. В связи с этим при использовании решающего блока следует учитывать особенности его обработки пакетом по сравнению с остальной частью документа.

Область действия локальных определений и условий блока не выходит за его пределы, т. е. они не влияют на другие переменные и выражения вне блока. При решении блока пакет вычисляет его определения и условия на каждом шаге итеративного алгоритма и при нахождении решения блока передает его функции FIND или MINERR конца блока, позволяя использовать это решение дальше в документе. Решение блока можно использовать только непосредственно через функцию его конца или определяя через нее новые функции и/или переменные.

При использовании внутри решающего блока графиков, вычисляемых выражений и глобальных определений они рассматриваются пакетом как находящиеся логически выше данного блока. И в этом плане использование данных конструкций внутри решающего блока, как правило, не имеет особого смысла, а в ряде случаев может быть и некорректным. Тогда как использование в теле блока текстовых областей и зон позволяет во многих случаях лучше иллюстрировать документ. В предыдущих примерах было это продемонстрировано.

Пакет не допускает использования вложенных решающих блоков, но такие конструкции могут быть введены в документ опосредованно на основе функций пользователя, определяемых в терминах некоторых решающих блоков. Приведенные выше примеры иллюстрируют такую возможность. Дополнительно к сказанному следует отметить, что условия в теле блока могут содержать ранжированные, векторные и матричные выражения при условии, что они вычисляемы в них, например:

[H] X: = 0 B: = (1 2 3) A: = 1, 1.1 .. 10

$$M: = \begin{bmatrix} 1 & 0.47 \\ 0.42 & 2 \end{bmatrix}$$

 GIVEN $(\Sigma B) \cdot \cos(X) - \left[\sum_A 1 \right] \cdot |M| \cdot X \approx 0$

[K] find (X) = 0.037

Для выключения (по команде 'Equation' пакета) решающего блока необходимо выключать в нем по крайней мере выражение, содержащее функцию FIND или MINERR конца блока. Выключение только заголовка блока 'GIVEN' не исключает блока из процесса вычислений в документе.

Как уже отмечалось, независимо от способа завершения блока (по функции FIND или MINERR) для его решения используется один и тот же итеративный алгоритм. Однако в отличие от функции FIND по функции MINERR пакет возвращает значения ведущих переменных блока, минимизирующих величину вектора отклонения. И в этом плане такой подход может быть использован для решения ряда задач оптимизации, один из примеров такого рода был приведен выше. Но функция MINERR не во всех случаях гарантирует минимизацию вектора отклонений, поэтому рекомендуется использовать дополнительные проверки полученного с ее помощью результата. В случае затруднений при решении блока (по FIND или MINERR) рекомендуется использовать графические представления условий (или части их) блока, чтобы удовлетворительнее локализовать область возможного решения блока и более точно определить первые приближения для ведущих переменных решающего блока.

Пакет в случае успешного вычисления возвращает только одно решение блока, даже если блок на самом деле имеет множество таких решений. Для получения всех решений блока следует использовать тот или иной способ отделения решений, наилучшим образом учитывающий специфику решаемого блока (варьирование значений для первых приближений, ввод дополнительных отделяющих неравенств, отделение корней уравнений, использование графической информации и т. д.). Следующий фрагмент документа иллюстрирует отделение решений блока путем ввода дополнительного параметрического неравенства:

[H] A: = 3 X: = 0.5 · A Y: = X B: = 1
 GIVEN $X^3 + Y^3 \approx 3 \cdot A \cdot X \cdot Y$ $X + Y \approx A$ $B \cdot X \leq B$
 Res (B): = FIND (X, Y)

Res (1) = $\begin{bmatrix} 0.634 \\ 2.366 \end{bmatrix}$ Res (—1) = $\begin{bmatrix} 2.366 \\ 0.634 \end{bmatrix}$
 [K]

В следующем разделе обсуждаются особые и аварийные ситуации, возникающие при решении блоков, и приводятся основные рекомендации по их устранению.

5.4. АВАРИЙНЫЕ И ОСОБЫЕ СИТУАЦИИ ПРИ ИСПОЛЬЗОВАНИИ РЕШАЮЩИХ БЛОКОВ В СРЕДЕ ПАКЕТА

При работе с решающими блоками в среде пакета могут возникнуть особые и аварийные ситуации, идентифицируемые соответствующими диагностическими сообщениями. При невозможности получения решения блока пакет идентифицирует функцию FIND или MINERR его конца сообщением 'Did not find solution' и устанавливает значение переменной ERR пакета, равное вектору отклонения на момент прекращения обработки блока. Так как условия блока остались неудовлетворенными, то величина переменной ERR может быть относительно большой. Причин отсутствия решения у блока может быть несколько.

Если блок на самом деле не имеет решения, то пакет пытается найти такие значения для его ведущих переменных, которые минимизируют вектор отклонений. Пакет не сможет получить решение блока и в том случае, когда для одного из первых приближений определено действительное значение, тогда как соответствующая ему ведущая переменная блока требует комплексного значения. При нахождении решения блока пакет может попасть в локальный минимум для вектора отклонений, т. е. невозможно его минимизировать и вернуть решение через функцию FIND. В этом случае рекомендуется изменять первые приближения ведущих переменных либо дополнять тело блока неравенством, предохраняющим от попадания в локальный минимум. Но при решении блока пакет может попасть в точку, не являющуюся локальным минимумом, но из которой он не может определить следующую итерацию. В этом случае рекомендации те же, что и для предохранения от локального минимакса.

Пакет может не получить решения блока и по причине требуемой точности вычисления, определяемой предопределенной переменной TOL пакета. Если значение переменной ERR пакета относительно мало, то пакет может получить решение, близкое к искомому, но не способное удовлетворить все условия блока с погрешностью меньшей, чем TOL. В этом случае рекомендуется увеличить значение переменной TOL пакета, т. е. понизить точность решения блока. При необходимости вычисления значения ведущих переменных блока, невзирая на невозможность получения решения блока, следует функцию FIND конца блока заменить на функцию MINERR. Функция MINERR возвращает значения ведущих переменных, даже если они не снижают нормы вектора отклонений в пределах от нуля до TOL. Однако в том случае, когда блок на самом деле не имеет решения, данный прием не снимает вопроса, так как пакет помечает обе функции FIND и MINERR диагностическим сообщением 'Did not find solution'.

Диагностическое сообщение 'Not converging' идентифицирует ситуацию, когда пакет при решении блока произвел слишком много последовательных итераций без возврата решения по функции FIND конца блока. Как правило, данная ситуация возникает при отсутствии решения у блока либо при определении значений для первых приближений ведущих переменных, весьма далеких от решения блока. В этом случае рекомендуется использовать более приемлемые значения для первых приближений ведущих переменных решающего блока, применяя, например, графическое представление условий, анализ типов возможных решений и т. д.

Диагностическое сообщение 'Too few constraints' идентифицирует решающий блок, содержащий условия, количество которых меньше, чем количество его ведущих переменных. В такой ситуации пакет не имеет достаточно информации для получения решения. Для устранения подобной ситуации блок следует дополнять условиями согласно количеству его ведущих переменных. Еще одним подходом является фиксация некоторых ведущих переменных блока путем присвоения им конкретных значений, что позволяет понизить их количество до числа условий. Эффективным может оказаться и способ введения в решающий блок фиктивных условий, подобных « $Y \approx Y$ » и являющихся, по сути дела, тождествами. Такой подход позволяет получить решение из широкого класса решений блока.

Наконец, в случае возникновения особой или аварийной ситуации при использовании функций пользователя (определенных в терминах решающего блока) в p -выражениях не выводятся полученные предыдущие до данной ситуации значения ведущих переменных, что существенно ограничивает применение указанного подхода для исследования, например, задач линейного и нелинейного программирования.

Глава 6

ВСТРОЕННЫЕ ФУНКЦИИ ПАКЕТА MathCAD

Пакет MathCAD располагает весьма широким набором встроенных функций, предназначенных как для организации вычислений, так и для работы с файлами данных. Все 75 встроенных функций пакета подразделяются на 16 групп, хотя в ряде случаев фирменная классификация встроенных функций является достаточно спорной. Полный перечень встроенных функций пакета с их краткой характеристикой приведен в прил. 7. В предыдущих и последующих главах книги рассматривался и будет рассматриваться подробнее ряд встроенных функций, смысл и назначение других функций либо достаточно просты, либо хорошо известны широкому кругу читателей и пользователей ЭВМ различного класса, поэтому вполне достаточно ограничиться их описанием (прил. 7). Здесь будут рассмотрены встроенные функции пакета MathCAD, требующие определенных пояснений и рекомендаций по их использованию. Это, в первую очередь, относится к встроенным функциям групп 7, 8, 10 и 13 (прил. 7).

6.1. ВСТРОЕННЫЕ ФУНКЦИИ ИНТЕРПОЛЯЦИИ

Группа 7 (прил. 7) включает функцию линейной интерполяции и четыре функции кубической сплайн-интерполяции. Функция `interp (VX, VY, X)` осуществляет линейную интерполяцию на основе векторов значений VX и VY $\{\text{length}(VX) = \text{length}(VY)\}$, где действительные элементы вектора VX должны быть отсортированы в порядке возрастания. Линейная интерполяция состоит в определении прямых линий $Y(X)$ между точками данных, на основе которых вычисляются интерполированные значения. Для нахождения интерполированного значения функции $Y(X_0)$ пакет определяет две точки X_j и X_{j+1} ($X_j \leq X_0 \leq X_{j+1}$) и в качестве линии интерполяции использует прямую, определяемую уравнением

$$(X_{j+1} - X_j) Y - [Y(X_{j+1}) - Y(X_j)] X + Y(X_{j+1}) X_j - Y(X_j) X_{j+1} = 0.$$

Искомое интерполированное значение $Y(X^0)$ получаем подстановкой в данное уравнение прямой значения $X = X^0$ и решением его относительно переменной $Y = Y(X^0)$. Вектора VX и VY здесь и ниже в разделе определяют соответственно значения данных $\langle X_0, X_1, \dots, X_j, \dots, X_n \rangle$ и $\langle Y(X_0), Y(X_1), \dots, Y(X_j), \dots, Y(X_n) \rangle$.

$X_1, \dots, X_j, \dots, X_n$ и $\langle Y(X_0), Y(X_1), \dots, Y(X_j), \dots, Y(X_n) \rangle$.

При этом действительные элементы вектора VX должны быть упорядочены в порядке возрастания. Если $X^0 < X_0$, то для линии интерполяции полагается $X_j = X_0$ и $X_{j+1} = X_1$; если же имеет место $X^0 > X_n$, то полагается $X_j = X_{n-1}$ и $X_{j+1} = X_n$. В качестве иллюстрации приведем фрагмент документа по вычислению интерполированных значений в точках, определяемых p -переменной X :

```
[N] ORIGIN:=1 K:=1..10 VX_K:=0.1·K+1 VY_1:=0.42
VY_2:=0.47 VY_3:=0.49 VY_4:=0.59 VY_5:=0.62
VY_6:=0.64 VY_7:=0.66 VY_8:=0.67 VY_9:=0.89
VY_10:=0.9 X_K:=K+0.25 Y:=interp(VX, VY, X)
```

VX_K	VY_K
1.1	0.42
1.2	0.47
1.3	0.49
1.4	0.59
1.5	0.62
1.6	0.64
1.7	0.66
1.8	0.67
1.9	0.89
2.0	0.90

Рис. 11

График зависимости между векторами данных VX и VY

$Y^T = (0.48 \ 0.925 \ 1.025 \ 1.125 \ 1.225 \ 1.325 \ 1.425 \ 1.525 \ 1.625 \ 1.725)$
 В приведенной задаче для интерполирования значений вектора $Y(X)$
 [K] функция 'interp' линейной интерполяции векторизуется.

Для получения интерполяционных формул с гладкими производными всевозрастающее значение получает сплайн-интерполяция. Кубическая сплайн-интерполяция состоит в сглаживании кривой на множестве точек таким образом, что первая и вторая производные сглаживаемой кривой являются непрерывными. Искомая кривая определяется рядом отрезков кубических функций, соединенных своими концами. Для обеспечения данного типа интерполяции пакет располагает четырьмя встроенными функциями: 'lspline', 'pspline', 'cspline' и 'interp'. Сплайн-интерполяция реализуется в следующие два этапа:

1) использование функции 'lspline', 'pspline' или 'cspline' на векторах данных VX и VY возвращает вектор VS , содержащий значения вторых производных сглаживаемой кривой в заданных точках; три указанные функции соответствуют трем различным типам кривых, используемых для интерполяции на отрезках $[X_j, X_{j+1}]$ ($j = \overline{0, n}$): прямая линия, парабола и кубическая парабола;

2) использование функции 'interp' на векторах VS (получен на предыдущем этапе), VX и VY возвращает искомое интерполированное значение $Y = Y(X)$.

Функция $\{l | p | c\} spline(VX, VY)$ на векторах данных VX и VY возвращает вектор VS значений вторых производных, используемых в дальнейшем функцией 'interp' интерполяции. Тогда как функция $interp(VS, VX, VY, X)$ возвращает значение интерполированной на основе векторов VS , VX и VY функции в точке X . Кубическая сплайн-интерполяция соответствует соединению заданных пар точек $[X_j, Y(X_j)]$ и $[X_{j+1}, Y(X_{j+1})]$ ($j = \overline{0, n}$) гладкой кривой, тип которой определяется согласно функции, выбранной на первом этапе интерполяции: прямая (lspline), парабола (pspline) или кубическая парабола (cspline). При этом части сглаживаемой кривой интерполяции на каждом отрезке $[X_j, X_{j+1}]$ определяются различными функциями, но в точках X_j соединения отрезков производная справа равна производной слева ($j = \overline{1, n-1}$). Для вычисления интерполированного значения в точке X пакет находит точки (X_j, X_{j+1}) такие, что $X_j \leq X \leq X_{j+1}$ ($j = \overline{0, n}$) и возвращает соответствующее величине X значение ординаты кривой интерполирования, отвечающей отрезку $[X_j, X_{j+1}]$. Если $X < X_0$, то пакет использует кривую интерполяции для отрезка $[X_0, X_1]$, если же $X > X_n$, то используется кривая интерполяции для отрезка $[X_{n-1}, X_n]$. При этом не рекомендуется использовать функцию 'interp' для вычисления интерполированных значений в точках X , достаточно далеких от отрезка интерполяции $[X_0, X_n]$, так как получаемые таким образом значения в общем случае некорректны для дальнейшего использования. Вектор VS содержит информацию, определяющую точную форму кривой интерполяции на заданном отрезке интерполирования. Используя векторное

представление, приведем фрагмент документа, в котором применяется сплайн-интерполяция для вычисления значений функции $Y(X)$ на векторе X значений ее аргумента:

```
[H] K:=0..5 VXK:=K+0.42 VY0:=19.42 VY1:=19.59
    VY2:=19.62 VY3:=19.66 VY4:=19.67 VY5:=19.86
VS:=cspline (VX, VY) XK:=0.8·K Y:=interp (VS, VX, VY, X)
    VS=(-0.357 -0.14 0.077 -0.109 0.18 0.469)
    YT=(19.255 19.515 19.599 19.619 19.654 19.661)
```

Рис. 12

[K]

Наряду с получением интерполированных значений функции на векторе X значений аргумента в приведенном фрагменте выводятся значения элементов промежуточного вектора VS (результат первого этапа кубической интерполяции), а также — графики зависимостей между векторами VX и VY исходных данных и кривой, построенной на основе интерполированных значений функции $Y(X)$ на векторе X значений ее аргумента. Приведенные графики демонстрируют вполне удовлетворительное соответствие исходной кривой и результата кубической сплайн-интерполяции на отрезке $[0, 4]$.

6.2. ФУНКЦИИ БЫСТРОГО ДИСКРЕТНОГО ПРЕОБРАЗОВАНИЯ ФУРЬЕ

Многообразие применения преобразования Фурье в теории вероятностей (теория характеристических функций), при решении краевых задач и сингулярных интегральных уравнений (метод Винера—Хопфа), а также в таких прикладных областях, как электроника, электротехника, техническая кибернетика и многих других хорошо отвечает группа 8 (прил. 7) встроенных функций для быстрого дискретного преобразования Фурье (БДПФ). Данная группа функций дает возможность пользователю достаточно эффективно организовать в своем документе метод вычисления дискретного преобразования Фурье, сокращающий количество операций с n^2 до $n \log_2 n$, где n — число интервалов представления. Пакет MathCAD включает четыре функции для использования БДПФ: две функции ('fft' и 'ifft') для действительных данных и две ('cfft' и 'icfft') для комплексных. Все эти функции пакета дискретны, определены на векторах и возвращают в качестве результата также вектора. Следует при этом помнить, что при определении вектора V (аргумента функций БДПФ) необходимо определять и его элемент V_0 (если используется значение переменной пакета $ORIGIN = 0$), ибо в противном случае пакет автоматически присвоит ему нулевое значение, что может привести к некорректным результатам.

Функция $fft(V)$ возвращает БДПФ вектора V действительных данных, представляющих результаты измерения на регулярных временных интервалах. Результатом функции является вектор комплексных коэффициентов, представляющих значения в частотном диапазо-

не. Если исходный вектор V содержит $n = 2^m$ элементов, то результирующий вектор C коэффициентов C_k вычисляется по формуле

$$C_k = \frac{1}{\sqrt{n}} \sum_{j=0}^n V_j e^{2\pi i (k/n)j} \quad (n = 2^m; k = \overline{0, n/2})$$

и содержит $2^{m-1} + 1$ элементов. Для БДПФ действительных данных вторая половина преобразования является сопряжением первой, что является причиной того, что пакет игнорирует вторую половину результирующего вектора C , возвращая только его первую половину плюс один элемент, т. е. $\text{length}(C) = 2^{m-1} + 1$. При необходимости получить вторую половину вектора C следует вместо $\text{fft}(V)$ использовать функцию $\text{cfft}(V)$.

Функция $\text{ifft}(V)$ возвращает обратное преобразование Фурье вектора V действительных данных, представляющих значения в частотном диапазоне. Результатом функции является вектор D , представляющий значения во временном диапазоне. Функция 'ifft' является строго обратной к функции 'fft' , примененной к действительным данным во временном диапазоне, т. е. для каждого допустимого действительного вектора V имеет место соотношение $\text{ifft}(\text{fft}(V)) = V$. Аргументом функции $\text{ifft}(V)$ является вектор V , подобный результирующему вектору функции 'fft' . Для вычисления результата функции $\text{ifft}(V)$ пакет сначала формирует новый вектор $W = (V_0, V_1, \dots, V_n, V_1, V_2, \dots, V_n)$, а затем возвращает вектор D , чьи элементы D_k вычисляются по формуле

$$D_k = \frac{1}{\sqrt{n}} \sum_{j=0}^n W_j e^{-2\pi i (k/n)j} \quad (n = 2^m; k = \overline{0, 2n-1})$$

и количество элементов которого равно 2^{m+1} .

Для применения БДПФ к комплексным данным пакет располагает функциями $\text{cfft}(V)$ и $\text{icfft}(V)$, являющимися также взаимно обратными, т. е. для любого допустимого комплексного вектора V имеет место соотношение $\text{icfft}(\text{cfft}(V)) = V$. При этом функция $\text{cfft}(V)$ аналогична функции $\text{fft}(V)$ за исключением того, что для первой вектор V может содержать комплексные элементы и результирующий вектор имеет ту же длину, что и вектор — аргумент V . Функция $\text{icfft}(V)$ аналогична функции $\text{ifft}(V)$ за исключением того, что длина вектора-аргумента первой функции меньше на единицу длины вектора-аргумента второй функции, а длины векторов аргумента и результата первой функции совпадают. Приведем фрагмент документа, иллюстрирующий использование БДПФ для действительных данных:

[H] $k := 0 \dots 63$ $V_k := 3 \cdot \left[1 - \cos \left[\frac{\pi}{5} \cdot k \right] \right] + \sinh(0.01 \cdot k)$

$C := \text{fft}(V)$ $n := \text{last}(C)$ $n = 32$ $j := 0 \dots n$

$D := \text{ifft}(C)$ $nl := \text{last}(D)$ $nl = 63$ $t := 0 \dots nl$

Рис. 13

Результаты прямого и обратного БДПФ представлены графически

[K]

В ряде случаев может потребоваться использование дискретного преобразования Фурье в версии, отличной от реализованной в пакете. Как правило, в этих случаях легко определить новые функции пользователя в терминах встроенных функций БДПФ пакета. Например, на основе определения прямого и обратного БДПФ вида:

$$F(V) = \frac{1}{n} \sum_{W=1}^n f(W) e^{-2\pi i (V/n) W};$$

$$f(W) = \sum_{V=1}^n F(V) e^{2\pi i (W/n) V}$$

легко ввести две функции пользователя

$$\text{gfft}(V) := \frac{1}{\sqrt{n}} \cdot \text{icfft}(V) \quad \text{и} \quad \text{igfft}(V) := \sqrt{n} \cdot \text{cfft}(V),$$

использование которых соответственно вместо встроенных функций $\text{fft}(V)$ и $\text{ifft}(V)$ позволяет погрузить в среду документа и сделать доступными для пользователя новое определение БДПФ. В книге [27] представлен обзор последних успехов в разработке алгоритмов для дискретного преобразования Фурье и цифровой обработки сигналов. Наряду с этим в книге даны интересные обсуждения прикладных аспектов представленных алгоритмов для широкого класса последовательных и параллельных суперЭВМ.

6.3. ВСТРОЕННЫЕ СТАТИСТИЧЕСКИЕ ФУНКЦИИ

Математическая статистика занимается как статистическим описанием результатов опытов или наблюдений в различных областях, так и построением и проверкой подходящих математических моделей, содержащих понятие вероятности. Ее методы расширяют возможности научного предсказания и рационального принятия решений во многих задачах, где существенные параметры не могут быть известны или контролируемы с достаточной степенью точности. Настоящий раздел имеет дело с классической выборочной статистикой. Для обработки различного рода статистических данных пакет располагает встроенными функциями выборочной статистики, статистических распределений, корреляции, регрессии и подготовки данных для построения гистограмм. Если статистическая функция определена на векторах данных, то требуется корректное использование индексов в соответствии со значением переменной пакета ORIGIN.

Для выборочной статистики пакет предлагает три встроенные функции $\text{mean}(V)$, $\text{var}(V)$ и $\text{stdev}(V)$, заданные на векторе V данных (элементы которого V_j ($j = 0, n-1$) определяют некоторую выборку) и возвращающие скалярные значения соответственно выборочных средних, дисперсии и стандартного отклонения. Указанные характеристики выборочного распределения вычисляются соответственно по следующим формулам:

$$\text{mean}(V) = \frac{1}{n} \sum_0^{n-1} V_j; \quad \text{var}(V) = \frac{1}{n} \sum_0^{n-1} [V_j - \text{mean}(V)]^2;$$

$$\text{stdev}(V) = \sqrt{\text{var}(V)}; \quad V = (V_0, V_1, \dots, V_{n-1}).$$

Для выборочных распределений пакет располагает следующими тремя функциями $\text{spnorm}(X)$, $\text{erf}(X)$ и $\Gamma(Z)$, определяемыми соответственно следующими интегралами:

$$\text{spnorm}(X) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^X e^{-t^2/2} dt \text{ (стандартное нормальное распределение);}$$

$$\text{erf}(X) = \frac{2}{\sqrt{\pi}} \int_0^X e^{-t^2} dt \text{ (функция ошибок);}$$

$$\Gamma(Z) = \int_0^{\infty} t^{Z-1} e^{-t} dt; Z > 0 \text{ (гамма-функция Эйлера).}$$

В первых двух интегральных формулах значение переменной X должно быть действительным. Учитывая возможность представления ряда важных математических функций в терминах функции Эйлера $\Gamma(Z)$, ее можно использовать также для определения в документе соответствующих функций пользователя. Например, следующее определение вводит известную Бета-функцию $B(p, q)$:

$$B(p, q) = \frac{\Gamma(p) \cdot \Gamma(q)}{\Gamma(p+q)}.$$

Для вычисления коэффициента корреляции и линейной регрессии между двумя связанными выборками $VX = (X_0, X_1, \dots, X_{n-1})$ и $VY = (Y_0, Y_1, \dots, Y_{n-1})$ пакет располагает тремя функциями 'corr', 'slope' и 'intercept'. Коэффициент корреляции вычисляется по формуле

$$\text{corr}(VX, VY) = \frac{\sum_{j=0}^{n-1} [X_j - \text{mean}(VX)] [(Y_j - \text{mean}(VY))]}{\text{stdev}(VX) \cdot \text{stdev}(VY) \cdot n}.$$

При предположении приблизительно линейной корреляции величина $|\text{corr}(VX, VY)| \leq 1$ есть мера степени связи между выборками VX и VY . Чем ближе величина $|\text{corr}(VX, VY)|$ к 1, тем эта связь теснее, и наоборот. Функции $\text{slope}(VX, VY)$ и $\text{intercept}(VX, VY)$ возвращают скаляр, определяющий соответственно наклон и отсекаемый на оси Y отрезок прямой линии регрессии, полученной по методу наименьших квадратов для связанных выборок VX и VY . Таким образом, прямая линия регрессии VY на VX описывается уравнением $Y = \text{slope}(VX, VY)X + \text{intercept}(VX, VY)$. Для реализации встроенных функций 'corr', 'slope' и 'intercept' пакет имеет специальную подпрограмму, реализующую метод наименьших квадратов как для действительных, так и для комплексных данных обрабатываемых выборок.

Функция $\text{hist}(V1, V2)$ возвращает вектор частот распределения данных, определяемых вектором $V2$, по интервалам, определяемым вектором $V1$. Векторы $V1$ и $V2$ должны быть действительными, а длина результирующего вектора частот равна величине $\text{length}(V1) - 1$.

Пакет интерпретирует упорядоченные в порядке возрастания элементы вектора $V1$ как концы прилегающих друг к другу интервалов на оси X . Результирующий вектор R функции 'hist' таков, что его элементы R_k определяют число элементов $V2_p$ вектора $V2$, удовлетворяющих неравенству $V1_k \leq 2_p < V1_{k+1}$ ($k = 0, n - 2$), при этом значения $V1_{n-1} < V2_p < V1_0$ игнорируются.

Встроенная функция $\text{rnd}(X)$ вычисляет псевдослучайное число из интервала $(0, X)$. При необходимости вычисления нового псевдослучайного числа достаточно курсор установить на выражение, содержащее функцию $\text{rnd}(X)$, и нажать клавишу 'F9'. Совместно с функцией $\text{rnd}(X)$ можно использовать и команду пакета

RANDOMIZE [⟨начальное число⟩] | — | C R

которая устанавливает начальное число для генератора псевдослучайных чисел (ГПСЧ) пакета. Если команда 'Randomize' введена без параметра, то пакет выводит текущее начальное число для ГПСЧ (по умолчанию $\text{Seed} = 1$) и дает возможность его изменить. После ввода команды 'Randomize' каждое использование функции $\text{rnd}(X)$ будет генерировать новую последовательность псевдослучайных чисел (ГПСЧ). Следующий фрагмент документа иллюстрирует результат вычисления десяти псевдослучайных чисел из интервала $(0, 1)$ посредством функции 'rnd' в зависимости от начального значения для ГПСЧ:

```
[H] j: = 0..9 Rj:= rnd (1)
      RT= (0 0.19 0.58 0.35 0.82 0.17 0.71 0.3 0.09 0.15)
      Rj:= rnd (1)
      RT= (0.99 0.12 0.01 0.53 0.6 0.17 0.45 0.06 0.78 0.52)
      *** Следующие результаты вычислений производятся после ввода команды
      пакета 'Randomize 15', установки курсора на первое определение Rj и на-
      жатия клавиши 'F9' (Calc)***
      RT= (0 0.83 0.99 0.68 0.88 0.11 0.17 0.79 0.55 0.93)
[JK] RT= (0.87 0.53 0.25 0.64 0.87 0.45 0.4 0.63 0.66 0.48)
```

Таким образом, при одинаковых начальных значениях для ГПСЧ пакет всегда генерирует одну и ту же ГПСЧ, а при разных начальных значениях разные последовательности. Следовательно, для генерации различных ГПСЧ необходимо использовать команду 'Randomize' с различными значениями ее параметра. Команду 'Randomize' можно использовать и для организации вычислений с одной и той же ГПСЧ, устанавливая одно и то же начальное значение для ГПСЧ. Следующий фрагмент документа иллюстрирует использование рассмотренных статистических функций пакета:

```
[H] ORIGIN:= 1 k:= 1..100 VXk: = rnd (48) VYk: = rnd (43)
      X:= 0,0.1..1 mean (VX) = 24.284 mean (VY) = 20.566 h:= 10
      var (VX) = 184.424 var (VY) = 139.189 stdev (VX) = 13.58 d:= 5
      stdev (VY) = 11.798 corr (VX, VY) = -0.021 B:= intercept (VX, VY)
      A:= slope (VX, VY) Y (X): = A · X + B j:= 1..d Lj:= j + h · j
```

$p := 1 \dots d - 1$ $H := \text{hist}(L, VX)$ $t := 1 \dots (d - 1)$ $H^T = (16 \ 25 \ 28 \ 6)$
 $G_{2 \cdot p - 1} := H_p$ $G_{2 \cdot p} := H_p$ $D_{2 \cdot p - 1} := L_p$ $D_{2 \cdot p} := L_p + 10$

Рис. 14

[K]

В приведенном фрагменте наряду с использованием рассмотренных статистических функций относительно выборок VX и VY приводятся график прямой регрессии $Y(X) = AX + B$ и гистограмма выборки VX относительно вектора L интервалов частот. Для читателя может представить определенный интерес простой прием, использованный для представления гистограммы в общепринятом виде, так как функция 'hist' пакета возвращает только вектор частот, не позволяющий непосредственно получать гистограммы в традиционном наглядном виде.

6.4. ФУНКЦИИ УПРАВЛЕНИЯ ВЫЧИСЛЕНИЯМИ В ДОКУМЕНТЕ

Из группы 13 (прил. 7) рассмотрим только две функции 'until' и 'if', позволяющие вводить элементы управления в вычислительный процесс документа. Остальные функции данной группы достаточно хорошо известны и особых пояснений не требуют. Встроенная функция пакета

until (expr1, expr2)

позволяет прекратить итеративный вычислительный процесс в зависимости от значения выражения 'expr1'. Принцип выполнения функции 'until' состоит в следующем: итеративное вычисление p -выражения, содержащего функцию 'until', прекращается как только значение p -выражения 'expr1' становится отрицательным; функция возвращает значение выражения 'expr2' до тех пор, пока значение ее первого аргумента положительно. Последним возвращаемым значением функции 'until' является значение выражения 'expr2', вычисленное в момент первого отрицательного значения p -выражения 'expr1'.

Функция 'until' не действует для скалярных выражений и имеет смысл только для p -выражений. Не рекомендуется использовать (или весьма осмотрительно) функцию 'until' с выражениями, содержащими более одной p -переменной, так как при указанном условии сразу прекращаются все итерации как только первый аргумент функции принимает отрицательное значение. Приведем фрагмент документа, использующего встроенные функции 'until' и Хэвисайда для вычисления нулевого корня функции с заданной точностью E и на заданном отрезке методом последовательного анализа ее значений в точках отрезка, вычисляемых с определенным шагом:

[H] $X := 1,1 \dots 2$ $E := 0,01$ $F(X) := e^X - X^5$

$$\sum_X (\text{until}(|F(X)| - E, X) - \Phi(|F(X)| - E) \cdot X) = 1,295$$

[K]

$F(1,295) = 0,009$

Как следует из определения функции 'until' и приведенного примера, использование функции может оказаться весьма эффективным в итеративных вычислениях с заданным условием сходимости. В первую очередь это относится к итеративным вычислениям по общей схеме $X_{k+1} = F(X_k)$ при $X_k < E$ или $F(X_k) < E$, где F — некоторая функция и X_k или $F(X_k)$ — искомое значение с точностью E . В гл. 10 будут приведены и другие интересные примеры на использование функции 'until'.

Пакет не располагает средствами для организации циклических и ветвящихся вычислительных процессов. Однако встроенная функция пакета

if (cond, expr1, expr2)

позволяет возвращать значения выражения 'expr1' или 'expr2' в зависимости от истинности условия 'cond'. В качестве условия 'cond' может быть любое допустимое выражение, хотя, как правило, в качестве этого аргумента функции используются выражения, включающие одну из операций отношения $\{ \approx | > | < | \geq | \leq | \neq \}$, смысл которых достаточно ясен без дополнительных пояснений. Если аргумент 'cond' принимает истинное значение ($\neq 0$), то функция 'if' возвращает значение выражения 'expr1', в противном случае ($\text{cond} = 0$) возвращается значение выражения 'expr2'. Таким образом, в зависимости от истинности или ложности аргумента 'cond' функция 'if' возвращает соответственно значение выражения 'expr1' или 'expr2'. При этом невычисляемый в данный момент аргумент функции не проверяется на наличие ошибок. В качестве иллюстрации сказанного приведем фрагмент документа, использующего функцию 'if':

[H] $\sigma(N, M) := \text{if}(N - M, 0, 1) \quad X := 0, 0.1 \dots 2$
 $Y(X) := \text{if}(X \leq 1, e^X, \ln[X^3]) \quad \sigma(1,2) + \sigma(2,2) = 1$

Рис. 15

Вычисление суммы числового ряда с заданной точностью E , используя операцию ранжированного суммирования и встроенные функции 'until' и 'if' пакета:

$\infty := 10^8 \quad n := 1 \dots \infty \quad E := \text{TOL} \cdot 0.1 \quad \sigma(n) := \frac{1}{n \cdot (n+1)}$
 $Z := \sum_n \text{until}(\text{if}(|\sigma(n)| < E, -1, 1), \sigma(n)) \quad Z := 0.99685$

Представление функции $\sin(z)$ рядом Маклорена:

[K] $\sin(z) := \sum_n (-1)^{n-1} \cdot \frac{z^{2 \cdot n - 1}}{(2 \cdot n - 1)!}$

В данном примере проиллюстрировано использование функции 'if' для определения двух кусочно-заданных функций, включая встроенную функцию Кронекера, а также построение графика одной из функций, определенных в терминах функции 'if'. Затем функции 'if' и 'until' используются для вычисления суммы числового ряда с заданной точностью. В гл. 10 будет приведен ряд других интересных примеров на использование обеих функций.

Наряду с этим данный пример (как и несколько предыдущих) иллюстрирует и определенную избыточность множества встроенных функций пакета. Действительно, следующие определения функций пользователя еще раз подтверждают сказанное:

$$\begin{aligned}\log(Z) &:= \ln(Z) / \ln(10) \quad \text{last}(V) := \text{length}(V) + \text{ORIGIN} - 1 \\ \text{stdev}(V) &:= \sqrt{\text{var}(V)} \quad \text{rows}(M) := \text{length}(M^{(1)}) \\ \text{cols}(M) &:= \text{length}(M^T{}^{(1)})\end{aligned}$$

Для ряда других встроенных функций также имеется возможность определения их в терминах встроенных функций пакета (\tanh , \cosh , \tan , \sinh). Пользователю для круга своих вычислительных задач рекомендуется создавать наборы глобально определенных функций, помещать их в специальный MCD-файл (или группу MCD-файлов) и по мере необходимости загружать в текущий документ по команде 'Append' пакета. Подобная библиотечная организация в процессе эксплуатации пакета показала достаточно высокую эффективность.

Глава 7

МЕТОДЫ ДОСТУПА И РАБОТА С ФАЙЛАМИ ДАННЫХ

Наряду с прямым использованием числовой информации в документе пакет располагает средствами для создания и чтения файлов данных (ФД). Допустимыми для пакета являются текстовые файлы в коде ASCII, содержащие числовую информацию. Пакет, создавая такого типа ФД, разрешает использование их другими программными средствами (Бейсик-программами, текстовыми редакторами, табличными процессорами, системами программирования, СУБД и т. д.). Кроме того, пакет может использовать ФД такого типа, подготовленные другими программными средствами. Для создания и доступа к ФД пакет располагает двумя группами встроенных функций: 1) доступа к неструктурированным ФД (НФД) и 2) доступа к структурированным ФД (СФД).

Файл данных, допускаемый пакетом MathCAD, должен быть текстовым файлом формата ASCII, в котором содержатся числовые данные (записи), разделенные символами пробела, запятой, точки с запятой, табуляции или символами «перевода строки и возврата каретки». В качестве числовых данных файла могут быть целые и действительные числа со знаком, а также числа в экспоненциальной форме (Е-формат), например:

```
1947  19.89      -32.44      42.12E-3   -31.14E+5
      -145  13.02E+89  84.8586  19.42E-23
```

где формат «AE ± n» представляет число вида $A \times 10^{\pm n}$. Методы доступа пакета не позволяют непосредственно работать с комплексными данными, но ниже будут даны рекомендации на эту тему. При создании пакетом ФД помещаемые в него данные разделяются пробелами и знаками возврата каретки, ограничивающими размер строки

при последующем выводе его на экран. Для создания ФД (особенно большого размера) с целью последующего применения его в среде пакета рекомендуется пользоваться любым доступным программным средством, позволяющим создавать и корректировать числовые ФД в формате ASCII подобно обычным текстовым файлам. Так, при использовании пакета на отечественном ПК ИСКРА 1030 весьма простым и эффективным средством для этих целей оказывается текстовый редактор, поставляемый с системой АДОС 2.0. Например, пакет может использовать любой из числовых файлов формата ASCII, подготовленных следующими программными средствами:

- системами программирования (Basic, Turbo-C, Turbo-Pascal);
- текстовыми процессорами (WordStar, Sprint);
- табличными процессорами (SuperCalc, Quattro);
- СУБД (dBase, Rbase, Foxbase, Reflex, Paradox);
- непосредственно с консоли по системной команде COPY;
- программами статистического анализа программных и технических средств ПК.

Рассмотрим каждую из групп встроенных функций пакета, реализующих доступ к файлам данных двух перечисленных типов.

7.1. ДОСТУП К НЕСТРУКТУРИРОВАННЫМ ФАЙЛАМ ДАННЫХ

Неструктурированным в среде пакета называется ФД, содержащий в общем случае набор отдельных числовых данных. Для доступа к неструктурированным ФД (НФД) пакет располагает тремя встроенными функциями: READ, WRITE и APPEND, перед рассмотрением назначения которых обсудим ряд общих для них свойств и особенностей. Все три функции доступа к данным имеют единый формат

WRITE ({<Имя файла> | <Имя переменной>})

READ ({<Имя файла> | <Имя переменной>})

APPEND ({<Имя файла> | <Имя переменной>})

где в качестве аргумента может быть либо имя ФД (но не его спецификатор), либо имя **ассоциированной** с ФД переменной. Все три имени встроенных функций должны кодироваться прописными буквами. Аргумент этих функций не вычисляется, а идентифицирует имя или ассоциированную переменную ФД, куда (откуда) записывается (читаются) данные. В отличие от имени самих функций их аргумент кодируется в полном соответствии с системными соглашениями по именам файлов и переменных. В качестве параметра функций не допускается прямое использование спецификатора ФД, что в свою очередь не позволяет непосредственно аргументом функции идентифицировать произвольный файл в системе. Для этих целей предварительно следует использовать специальную команду пакета

FILENAME <Имя переменной> [<СФ>] | — | F F

позволяющую ассоциировать с указанным параметром 'СФ' ФД переменную, имя которой определяется первым параметром команды.

При отсутствии у команды второго параметра пакет запрашивает спецификатор ФД ('Filename = '), а при вводе команды посредством командного меню (клавиша 'F10') — имя переменной (File variable:) и после этого предлагает определить имя ассоциируемого с ней файла ('Filename = '). Пакет использует результат выполнения команды 'Filename' как для существующего, так и для вновь создаваемого ФД. С помощью команды 'Filename' устанавливается логическая связь между переменной и спецификатором ФД любого типа (НФД или СФД) таким образом, что каждая функция доступа к данным, использующая эту переменную в качестве аргумента, будет адресоваться к ассоциированному с ней ФД. Следует иметь в виду, что с одним и тем же ФД может быть ассоциировано несколько переменных (но не наоборот), что позволяет организовывать в документе более гибкий доступ к данным.

Используя команду 'Filename' для установления логической связи между переменными и ассоциированными с ними файлами данных любого типа (как неструктурированными, так и структурированными), следует иметь в виду, что текущий документ может одновременно работать не более чем с 15 различными ФД. Попытки посредством использования команды 'Filename' определить большее число ФД приводят к ошибке 'Too many files', когда осуществляется попытка доступа к ФД, определенному в документе сверх имеющихся 15. Однако для одного и того же ФД можно определить достаточно много ассоциированных с ним как простых, так и ранжированных переменных.

При использовании ассоциированных с ФД переменных их можно ранжировать, тем самым получая возможность организовывать последовательный и некоторые другие способы доступа к данным. Наряду с этим ассоциированные переменные в отличие от функций доступа позволяют задавать полный путь к ФД, связывая переменную со спецификатором файла, что дает возможность работать с ФД, расположенными на любом допустимом устройстве ПК (НГМД, НМД, Винчестер и т. д.). После удаления текущего документа по команде 'Clear' информация команды 'Filename' не сохраняется (в отличие, например, от содержимого ВПП любого типа), поэтому в последующих документах требуется новое определение ассоциированных с файлами переменных. Но при сохранении текущего документа в дисковом файле по команде 'Save' информация команды 'Filename' сохраняется, позволяя без повторного определения использовать ассоциированные переменные при загрузке документа в память по команде 'Load' пакета. После использования команды 'Filename' в середине документа в ряде случаев может потребоваться перечисление некоторых выражений, содержащих функции доступа к данным (по клавише 'F9'), или всего документа по команде 'Process'. Во избежание возможных упущений рекомендуется использовать второй подход актуализации текущего документа.

Для доступа к НФД пакет предлагает три основные конструкции, базирующиеся на встроенных функциях доступа к данным:

WRITE (<Имя файла> | <Имя переменной>) {:=|≡} <Выражение>
APPEND (<Имя файла> | <Имя переменной>) {:=|≡} <Выражение>
<Выражение> {:=|≡} **READ** (<Имя файла> | <Имя переменной>)

Данные конструкции являются базовыми, наряду с ними пакет допускает использование функций доступа **WRITE**, **APPEND** и **READ** в значительной мере аналогично другим функциям пакета. Подробнее об этом будет идти речь ниже. В случае глобального определения указанных конструкций все связанные с ними переменные (включая и предопределенные переменные пакета) и выражения также должны определяться глобально.

Для создания или обновления НФД используются функции доступа **WRITE** и **APPEND** пакета. Функция **WRITE** служит для формирования нового или полного обновления существующего ФД, заданного ее аргументом, в качестве которого может быть либо непосредственно имя ФД (и тогда он должен находиться в активном каталоге), либо имя ассоциированной с ФД переменной (определяемой посредством команды 'Filename' пакета и допускающей опосредованное использование для адресации ФД его спецификатора). При использовании функции **WRITE** следует помнить, что применение ее к существующему ФД приводит к полному его обновлению без выдачи какого-либо запроса на определение санкции со стороны пользователя, поэтому от пользователя требуется внимательность, обусловленная необходимостью сохранения существующей информации в его файлах данных.

При помещении в НФД нескольких записей следует использовать с функцией **WRITE** *p*-выражение или *p*-переменную, позволяющие осуществлять многократную запись в открытый файл. Помещаемые в выходной файл данные разделяются пробелами, а также символами «перевод строки», чтобы длина строки выводимых данных не превышала 80 байтов. Данные в выходной файл по функции **WRITE** пишутся с максимальной точностью (независимо от глобального формата чисел документа — 15 знаков после десятичной точки) и без приписанных им единиц измерений, если таковые имеются. При создании нового НФД посредством функции **WRITE** для определения его имени можно указывать полное имя, как включая расширение имени любого допустимого вида, так и опуская расширение имени. В последнем случае функция **WRITE** по умолчанию присваивает созданному НФД расширение имени '.DAT'. В случае создания НФД вне среды пакета посредством другого допустимого программного средства (текстовый редактор, программа на языке программирования, СУБД и т. д.) следует учитывать данную особенность использования имени файла данных в среде пакета. В противном случае возможны ошибки при его последующем использовании функциями доступа **APPEND** и **READ**.

Для возможности помещения в выходной файл данных из различных частей документа следует использовать функцию **APPEND**, которая отличается от функции доступа **WRITE** только тем, что позволяет дописывать данные в конец существующего НФД. При этом для нового ФД данная функция полностью эквивалентна по резуль-

тату применения функции WRITE, открывая и создавая новый файл согласно информации своего аргумента. Функция APPEND позволяет дописывать в существующий НФД как одиночные значения, так и их набор, для чего следует использовать ее в *p*-выражении или с *p*-переменной в качестве аргумента функции.

Следует иметь в виду, что в качестве правой части для функций WRITE и APPEND могут использоваться векторные, матричные и индексированные выражения, но определяемые ими значения записываются в выходной файл в виде последовательного неструктурированного набора записей, т. е. НФД. Примеры такого характера будут приведены в конце следующего раздела.

Встроенная функция READ служит для чтения из НФД, указанного своим аргументом, очередной записи данных. При задании через аргумент функции (непосредственно либо с помощью ассоциированной переменной) искомого НФД следует иметь в виду, что пакет при отсутствии расширения у имени файла полагает его по умолчанию '.DAT'. Попытка читать из НФД, не имеющего расширения, приведет к ошибке с диагностикой 'File not found', хотя на самом деле файл существует. Поэтому при создании НФД вне среды пакета обязательно следует указывать его расширение и для удобства дальнейшего использования файла в среде пакета полагать его '.DAT', так как данное расширение имени НФД пакет предполагает по умолчанию при использовании функций доступа WRITE, APPEND и READ.

При неудачной попытке найти или открыть НФД функция READ идентифицируется сообщением 'File not found', а при неверном формате данных файла возникает сообщение 'File error'. Следует помнить, что каждое новое выражение, содержащее функцию READ (как и другие функции доступа), переоткрывает соответствующий файл данных, давая возможность производить обработку его с самого начала. При необходимости организации доступа к ряду последовательных значений или к отдельному конкретному значению НФД следует использовать функцию READ в *p*-выражениях или с *p*-переменными, например, ранжируя ассоциированную с НФД переменную. При этом использование ассоциированной *p*-переменной непосредственно не ранжирует функцию доступа, ее следует использовать с подходящим *p*-выражением. Например, конструкция вида «X: = 1..100 WRITE (X): = 48» позволит создать новый файл, определяемый ассоциированной с ним *p*-переменной 'X', только из одной записи, равной 5, а не из 100 одинаковых записей. В связи с принципом выполнения функций доступа невозможно различными выражениями, содержащими READ, читать последовательные части одного и того же НФД. Для получения возможности просмотреть весь НФД функцию READ следует ранжировать либо на основе ее аргумента-переменной, либо используя ее в соответствующем *p*-выражении. Если функция READ при чтении записей файла встречает его конец, то пакет идентифицирует данную ситуацию сообщением 'End of file' и чтение данных прекращается. В качестве иллюстрации сказанного приведем фрагмент документа, в котором используются рассмотренные средства доступа к НФД пакета и их свойства:

[Н] Для определения ассоциированных переменных с НФД вводится группа команд пакета:

```
Filename X B:File.AVZ
Filename X1 A:File.AVZ
Filename Y B:File.DAT
Filename Y1 B:File.DAT
Filename Z C:File.Res
```

```
k:=5 X:=1..k Y:=1..k m:=2..5
WRITE(X):=rnd(47)+0.X APPEND(Y):=rnd(42)+0.Y
```

$S1 := \frac{1}{k} \cdot \sum_X \text{READ}(X)$ $S2 := \frac{1}{k} \cdot \sum_Y \text{READ}(Y)$ $S1 = 27.521$ $S2 = 15.414$

Вычисление эмпирических центральных моментов m-го порядка выборок

$mX(m) := \frac{1}{k-1} \sum_X (\text{READ}(X) - S1)^m$ $mY(m) := \frac{1}{k-1} \sum_Y (\text{READ}(Y) - S2)^m$

mX (m)		mY (m)
157.206	APPEND (X1) := mX (m)	222.284
—10422.654		—1831.038
$2.868 \cdot 10^5$		28223.218
$-7.894 \cdot 10^6$	APPEND (Y1) := mY (m)	$04.35 \cdot 10^6$

Вывод содержимого файлов данных File.AVZ и File.DAT после их создания посредством функций WRITE и APPEND и дополнения по функции APPEND значениями эмпирических центральных моментов

X := 1..k + 4	Y := 1..k + 4
READ (X) + 0 . X	READ (Y) + 0 . Y
42.842	4.56
34.2	35,751
31.385	6.505
14.806	3.333
14.374	26.921
157.206	222.284
—10422.654	—1831.038
$2.868 \cdot 10^5$	28223.218
$-7.894 \cdot 10^6$	$-4.35 \cdot 10^6$

Рис. 16

Иллюстрация невозможности ранжирования функций доступа только на основе ранжирования ассоциированной с НФД переменной и некоторых других их свойств

$$\begin{aligned} \text{READ}(X) &= 42.842 & \text{READ}(Y) &= 4.56 & \text{READ}(X) &= -7.618 \cdot 10^6 \\ \sum_X \delta(3, X) \cdot \text{READ}(X) &= 42.842 & \sum_X \text{READ}(X) \cdot \text{READ}(Y) &= 3.442 \cdot 10^{12} \\ X &:= 0 \dots 10 & \text{WRITE}(X) &:= X^3 \cdot \sin(X) & M_X &:= \text{READ}(X) \\ N_X &:= \cos(X) \cdot \sqrt{\text{READ}(X)} \end{aligned}$$

Рис. 17

Примеры организации на основе функции READ специальных методов доступа к НФД, реализованных на основе функций пользователя

$$X = 1 \dots 10 \quad k = 5 \quad n = 9 \quad \text{WRITE}(X) := X^2 + 1$$

F(5)	M ₇
2	26
5	37
10	50
17	65
26	82

$$N(8) = 65$$

$$N(4) = 17$$

$$N(10) = 101$$

$$N(400) = 0$$

$$Y := 1 \dots 100$$

$$Y1 := 1 \dots 200$$

$$\text{WRITE}(X) := \text{READ}(Y) + 0 \cdot Y$$

$$\text{APPEND}(X) := \text{READ}(Y1) + 0 \cdot Y1$$

Копирование файлов, определяемых ассоциированными с ними переменными

$$\text{READ}(X1) + 0 \cdot Y1$$

Распечатка файла данных после его копирования

19.42
19.47
19.67
...
19.89

$$X = 1 \dots 1000$$

$$m_X = \text{READ}(X)$$

$$z = k \dots n$$

$$F(k) = \text{until}[k - X - 1, m_X]$$

Чтение первых k записей НФД

$$N(k) = \sum_X \delta(k, X) \cdot m_X$$

Чтение k-й записи из НФД

$$M_z = \sum_X \delta(z, X) \cdot m_X$$

Чтение записей НФД, начиная с k-й по n-ю включительно

{K}

Приведенный фрагмент документа иллюстрирует применение функций доступа к НФД и некоторые особенности их выполнения. С учетом сказанного и приводимых в нем комментариев фрагмент документа не требует более подробного описания.

Наряду с действительными НФД пакет дает возможность работать и с комплексными файлами. В этом случае пакет по команде WRITE или APPEND создает на основе комплексного выражения файл данных, каждая пара элементов которого отвечает соответственно действительной и мнимой частям комплексного значения, помещаемого в файл. Таким образом, после создания НФД на основе комплексных данных он будет содержать ровно в два раза больше элементов, чем исходные данные: все элементы $(2 \times j + 1)$ отвечают действительным частям комплексных данных, а элементы $(2 \times j)$ — мнимым частям. На основе данной организации файла нетрудно создавать и использовать данные в их первоначальном комплексном виде. Следующий простой фрагмент документа иллюстрирует сказанное:

[H] $X := 0 \dots 2$ $Y := 0 \dots 5$ $WRITE(X) := 3 \cdot (X + 1) + (2 \cdot X + 1) \cdot i$

READ(X) + 0 · Y

READ(X) + i · READ(X) + 0 · X

3
1
6
3
9
5

$3 + i$
$6 + 3i$
$9 + 5i$

[K]

Приведенный пример достаточно прозрачен и особых пояснений не требует. Примеры подобного типа будут рассмотрены в следующем разделе.

7.2. ДОСТУП К СТРУКТУРИРОВАННЫМ ФАЙЛАМ ДАННЫХ

Под структурированным файлом данных (СФД) пакет понимает файл, содержащий числовые значения, упорядоченные в виде некоторой прямоугольной матрицы (таблицы) или вектора. Как правило, числовые файлы такого вида готовятся программными средствами типа «электронных таблиц» и табличных процессоров (например, SuperCalc, Quattro и т. д.) перед выводом их на печать, что и определило добавление к именам встроенных функций доступа трех букв 'PRN': WRITEPRN, APPENDPRN и READPRN. Сами же эти файлы относятся к классу так называемых PRN-файлов и имеют по умолчанию расширение имени '.PRN'. Для обеспечения доступа

к СФД пакет располагает подобно случаю НФД тремя встроенными функциями, имеющими следующий формат:

WRITEPRN (<Имя файла> | <Имя переменной>)
APPENDPRN (<Имя файла> | <Имя переменной>)
READPRN (<Имя файла> | <Имя переменной>)

Данные функции доступа по своему формату и основному назначению полностью соответствуют функциям **WRITE**, **APPEND** и **READ** доступа к НФД, поэтому рассмотрение их будет вестись с акцентом на их отличия от указанных функций доступа в случае НФД.

Для использования функций доступа к СФД пакет допускает следующие конструкции:

WRITEPRN (<Аргумент>) { : = | ≡ } <Выражение>
APPENDPRN (<Аргумент>) { : = | ≡ } <Выражение>
<Выражение> { : = | ≡ } **READPRN** (<Аргумент>)

В качестве аргумента функции доступа, как и в случае НФД, может использоваться либо непосредственное имя файла данных, либо имя ассоциированной с ним переменной, определенной посредством команды 'Filename' пакета. В данном случае также невозможно определить в текущем документе более 15 СФД, но один и тот же СФД может иметь большое количество ассоциированных с ним переменных. При глобальном определении функций доступа все связанные с ними переменные и выражения, включая и предопределенные переменные пакета, должны также определяться глобально. При использовании имен СФД по умолчанию полагается для них расширение '.PRN', но если функция **READPRN** обращается к ФД без расширения имени, то в случае отсутствия расширения или отличия его от '.PRN' идентифицируется особая ситуация 'File not found'. Поэтому для СФД всегда необходимо указывать расширение имени (необязательно '.PRN') при создании его вне среды пакета, например, с консоли по системной команде **COPY**.

В качестве 'Выражения' при определении функций доступа используется, как правило, матричное или векторное выражение, но функции **WRITEPRN** и **APPENDPRN** могут использовать также ранжированные и индексированные выражения. Однако в отличие от аналогичного случая для НФД результат таких операций имеет специфический вид, который следует учитывать при последующих его использованиях в текущем или других документах. Так использование индексированных выражений с функциями доступа **WRITEPRN** и **APPENDPRN** приводит к тому, что структура создаваемого ими СФД принимает вид вектора-столбца. С одной стороны, данная возможность может использоваться, например, для преобразования матриц в вектора. Тогда как, с другой стороны, такого характера операция **APPENDPRN** может создать неверный СФД, дописав, например, в конце СФД, содержащего матрицу, вектор-столбец. Последующие применения к такому СФД функции доступа **READPRN** приведут к ситуации 'File error'. Ниже будут приведены интересные

примеры подобного характера. Переходим теперь непосредственно к рассмотрению функций доступа к структурированным файлам данных.

Встроенная функция READPRN предназначена для чтения за одну операцию доступа всего СФД, указанного ее аргументом, определения количества строк (M) и столбцов (N) массива и создания в документе соответствующей (M × N)-матрицы из значений файла. При этом функция READPRN читает из файла только числовую информацию, игнорируя любую другую, включая текстовые метки и символы кавычек. Все строки СФД должны содержать одинаковое количество значений, в противном случае при выполнении функции READPRN пакет диагностирует ошибку 'File error'. Строки, не содержащие числовых значений, функцией READPRN игнорируются. Эта особенность оказывается (как будет показано ниже) весьма полезной при создании СФД вне среды пакета. Так, для определения числовой (M × N)-матрицы на основе некоторого СФД с именем 'matrix . prn' достаточно в документе ввести определение вида 'M: = READPRN (matrix)'. При этом для определяемой матрицы M не следует указывать нижних индексов. Функция доступа READPRN, открыв требуемый файл, выделяет отдельные его значения посредством разделяющих их пробелов, запятых, точек с запятыми или другой текстовой информацией. Поэтому в случае использования любой текстовой информации там, где функция READPRN предполагает числовые значения, не позволяет ей читать такой файл и вызывает ошибочную ситуацию 'File error'. После чтения значений в матрицу M функцией READPRN к ней применимы все допустимые пакетом матричные и векторные операции. Приведем простой пример чтения числовой (2 × 3)-таблицы, подготовленной посредством системной команды 'COPY CON Grodno.PRN' и помещенной в дисковый файл 'Grodno.PRN' на активном устройстве:

[H]	XXXXXXX	XXXXXXX	XXXXXXX	E	— Шаблон создаваемой строки
	0.42	0.47	0.59	E	числовой (2×3)-таблицы;
	0.62	0.64	0.67	E	идентифицирует нажатие
					клавиши 'Enter'

^Z (клавиша 'F6')

M := READPRN (Grodno) i := 0 .. rows (M) — 1 j := 0 .. cols (M) — 1

$$SR := \frac{1}{\text{rows}(M) \cdot \text{cols}(M)} \cdot \sqrt{\sum_{i=1}^{\text{rows}(M)} \sum_{j=1}^{\text{cols}(M)} M_{i,j}^2} \quad SR = 0.23$$

[K]

Встроенные функции доступа READ и READPRN в ряде случаев можно использовать в выражениях, аналогично другим функциям пакета, особенно применяя операцию векторизации. Однако в общем случае такие выражения некорректны, что иллюстрируют приводимые в конце раздела интересные примеры. Поэтому при желании использовать данные функции непосредственно в выражениях рекомендуется предварительно апробировать результаты такой организации вычислений в документе.

Для создания или обновления СФД используются встроенные функции доступа WRITEPRN и APPENDPRN, с некоторыми отличиями от подобных функций WRITE и APPEND для НФД. Функция доступа WRITEPRN служит для создания нового или полного обновления существующего файла данных, заданного ее аргументом. При этом санкцию на запрос для обновления существующего файла функция не запрашивает подобно случаю функции WRITE. В определении функции WRITEPRN можно использовать как векторно-матричные, так и ранжированные или индексированные выражения. Сказанное иллюстрирует следующий простой пример:

$$\begin{array}{l}
 \text{[H]} \quad j := 1..3 \quad i := 0..1 \quad A := \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} \quad B := \begin{bmatrix} 5 \end{bmatrix} \\
 \text{WRITEPRN}(X) := A \cdot j \quad \text{WRITEPRN}(V) := A + B_j \\
 \text{READPRN}(X) = \begin{bmatrix} 1 & 3 \\ 2 & 4 \\ 2 & 6 \\ 4 & 8 \\ 3 & 9 \\ 6 & 12 \end{bmatrix} \quad \text{READPRN}(V) = \begin{bmatrix} 6 & 8 \\ 7 & 9 \\ 6 & 8 \\ 7 & 9 \end{bmatrix} \\
 \text{[K]}
 \end{array}$$

Из приведенного примера легко получить алгоритм ранжирования и индексации, реализуемый функцией WRITEPRN. Так, например, ранжируя выражение в первом определении функции WRITEPRN, создающей дисковый файл 'X.rgn', получаем в результате не три отдельные (2×2) -матрицы A , умноженные на числа 1, 2 и 3 соответственно, а единую (6×2) -матрицу, являющуюся результатом своего рода «конкатенации» этих трех матриц, которая и составила выходной СФД 'X.rgn'.

Встроенная функция доступа APPENDPRN аналогична по своему действию функции WRITEPRN, но отличается только тем, что не обновляет полностью существующий СФД, а дописывает новое значение в его конец.

Совместно с функциями доступа WRITEPRN и APPENDPRN можно использовать предопределенные переменные пакета PRNCOLWIDTH и PRNPRECISION, определяющие формат создаваемого СФД. Значение переменной PRNCOLWIDTH (по умолчанию равно 8) определяет ширину столбцов файла в байтах, а значение переменной пакета PRNPRECISION (по умолчанию равно 4) — общее количество выводимых цифр числа (например, число 19.4248 запишется в файл в виде 19.42). При этом, если в файл помещаются значения, содержащие больше цифр, чем определено данной переменной, то пакет производит их округление (округляя как дробные, так и целые числа, помещаемые в файл). Для задания отличного от стандартного формата PRN-файла в документе следует определить указанные переменные пакета с новыми значениями. Созданный таким образом PRN-файл доступен для обработки многими программными средствами, включая «электронные таблицы», табличные и текстовые процессоры. Приведем простой пример использования описанных средств доступа для создания СФД:


```

[H]      i:=0..2   j:=0..2   k:=0..1   Ni,k:= $\frac{i}{k+1}$ 
      Mi,j:= $\sqrt{i+j}$    PRNCOLWIDTH:=6   PRNPRECISION:=3
      M= $\begin{bmatrix} 0 & 1 & 1.414 \\ 1 & 1.414 & 1.732 \\ 1.414 & 1.732 & 2 \end{bmatrix}$    N= $\begin{bmatrix} 0 & 0 \\ 1 & 0.5 \\ 2 & 1 \end{bmatrix}$    Y:=1..5
WRITEPRN (matrix) := augment (M, N)   APPENDPRN (matrix) := N(1) + 0 * Y
      L:=READPRN (matrix)   rows (L) = 15   cols (L) = 1
      LT=(0 0.5 1 0 0.5 1 0 0.5 1 0 0.5 1 0 0.5 1)
[K]   Конец фрагмента документа

```

Подобно случаю НФД пакет позволяет осуществлять работу и с комплексными СФД. Суть такой возможности состоит в следующем. При попытке записать СФД на основе комплексных значений пакет помещает в него вместо каждого значения пару значений, представляющих собой соответственно действительную и мнимую части записываемого комплексного числа, создавая файл действительных чисел, количество столбцов которого в два раза больше количества столбцов в чисто действительном СФД с тем же числом элементов и той же табличной структуры. При чтении таких файлов следует этот момент учитывать. Следующий простой фрагмент документа иллюстрирует сказанное:

```

[H]   k:=2       j:=0..k       i:=0..k       zi,j:= (3 * i + 1) + j * i
      WRITEPRN (M) := Z       N:=READPRN (M)
      N= $\begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 2 \\ 4 & 0 & 4 & 1 & 4 & 2 \\ 7 & 0 & 7 & 1 & 7 & 2 \end{bmatrix}$    T(j):=N(2*j) + i * N(2*j+1)
                                     T= $\begin{bmatrix} 1 & i+i & i+2i \\ 4 & 4+i & 4+2i \\ 7 & 7+i & 7+2i \end{bmatrix}$ 
[K]

```

Приведенный пример с учетом сказанного достаточно прозрачен и особых пояснений не требует. В заключении главы кратко рассмотрим основные различия организации работы с ФД обоих типов и приведем некоторые рекомендации по их использованию.

В основе различий обоих типов файлов данных и возможностей доступа к ним лежит организация данных файлов функциями доступа WRITE и APPEND, с одной стороны, и WRITEPRN и APPENDPRN, с другой. НФД содержит набор записей, разделенных пробелом, каждая из которых имеет переменную длину, определяемую содержащимся в ней действительным числом. Например, число, представленное конечным набором цифр (—45.678, 19.42 и т. д.), определяет длину записи ровно по количеству своих цифр, включая знак и десятичную точку плюс один пробел. Тогда как, например, иррациональные и трансцендентные числа требуют под запись значительно больше байтов, ибо только десятичная их часть будет содержать 15 цифр (максимальная точность, обеспечиваемая пакетом). Понимая теперь под длиной записи НФД количество всех

цифр, входящих в содержащееся в ней значение (включая знак числа и десятичную точку), легко получить формулу для оценки объема такого файла:

$$R \text{ (НФД)} = \sum_{j=1}^n (L_j + 1),$$

где n — число записей файла и L_j — длина j -й записи ($j = \overline{1, n}$).

Структура СФД имеет более четкую упорядоченную структуру, легко позволяющую на ее основе создавать табличные массивы в документе. Структура СФД имеет следующий вид:

```

ПСВК
ПСВК
<Шаблон> <Шаблон> <Шаблон> ... <Шаблон> ПСВК
. . . . .
<Шаблон> <Шаблон> <Шаблон> ... <Шаблон> ПСВК

```

где под 'шаблоном' понимается запись файла длиной, равной значению предопределенной переменной пакета PRNPRECISION. Под аббревиатурой 'ПСВК' понимается двухбайтный управляющий код ПК, обозначающий 'перевод строки и возврат каретки'. Каждый шаблон содержит одно значение СФД, а количество шаблонов в каждой строке должно быть одинаковым. Свободные от цифровой информации байты шаблона заполняются пробелами. Имея в виду такую организацию СФД, нетрудно получить точную оценку занимаемого им места на диске:

$$R(M, N) = (\text{PRNCOLWIDTH} \times N + 2) \times M,$$

где M и N — соответственно количество строк и столбцов СФД. Данную информацию можно использовать при выборе более эффективной организации работы с данными в среде пакета, а также при подготовке большого объема файлов данных обоих типов. Так на основе представленных формул можно получить оценочные характеристики при выборе той или иной организации данных (если суть задачи позволяет сделать такой выбор). Для этого достаточно проанализировать поведение следующего коэффициента:

$G = \text{PRNCOLWIDTH}$

$$K(\text{СФД/НФД}) = \frac{G \cdot M \cdot N + 2M}{\sum_1 L_j + N \cdot M} = \frac{G + \frac{2}{N}}{\frac{1}{M \cdot N} \sum_1 L_j + 1} \approx \frac{G \cdot M \cdot N}{\sum_1 L_j},$$

определяющего отношение объемов, занимаемых двумя файлами обоих типов и имеющих одинаковое количество записей. Для эмпирической оценки поведения данного коэффициента были собраны статистические данные и на их основе получены графики, определяющие поведение коэффициента для двух крайних условий. Следующий фрагмент иллюстрирует полученные результаты:

[Н]

$$t := 0 \dots 9 \quad T_t := 5 \cdot t + 10$$

	СФД	НФД		СФД	НФД
R1 :=	1729	1713	R2 :=	820	306
	3894	3878		1830	691
	6915	6905		3240	1230
	10797	10799		5050	1923
	15607	15659		7260	2771
	21231	21317		9870	3849
	27751	27902		12880	5183
	35166	35409		16290	6775
	43458	43819		20100	8621
	52539	52998		24310	10725

Характеристики объемов соответственно СФД и НФД при иррациональных значениях максимальной для пакета точности (R1) и при целочисленных значениях (R2)

$$K1 := \begin{bmatrix} \overrightarrow{R1^{(0)}} \\ \overrightarrow{R1^{(1)}} \end{bmatrix} \quad K2 := \begin{bmatrix} \overrightarrow{R2^{(0)}} \\ \overrightarrow{R2^{(1)}} \end{bmatrix}$$

Рис. 18

[К]

Уже из данных оценок можно сделать ряд практических выводов: чем выше точность используемых данных, тем менее существенно различие между ФД обоих типов относительно занимаемого ими объема на диске, а при максимальной точности пакета предпочтение уже даже на стороне СФД;

при использовании в качестве данных небольших (порядка 6—8 цифр) или целых чисел предпочтительнее оказываются НФД, если они допустимы условиями вычислений;

при необходимости работы с таблицами с количеством столбцов, большим 80, рекомендуется использовать функции доступа СФД.

На использование того или иного типа ФД оказывают влияние и другие факторы. Так при создании больших по объему ФД вне среды пакета предпочтительнее оказываются НФД вследствие более простой организации. Тогда как СФД формировать несколько сложнее. В этом случае предлагается довольно простой и эффективный прием, сводящийся к следующему (на примере формирования СФД с помощью системной команды COPY):

- 1) ввести команду COPY с необходимыми параметрами;
- 2) после готовности к вводу дважды нажать клавишу 'Enter';
- 3) ввести строку шаблонов создаваемого массива вида:
XXX ... X * XXX ... X * ... XXX ... X * E

где длина шаблона 'XXX ... X*' определяется значением переменной PRNCOLWIDTH пакета, а символ 'E' определяет позицию строки, в которой нажимается клавиша 'Enter'; число шаблонов в эталонной строке соответствует количеству столбцов создаваемого массива данных;

4) вводить под каждым шаблоном нужное числовое значение, помещая его относительно границ шаблона произвольно и заполняя пустые его позиции пробелами, запятыми или точками с запятой;

5) после ввода всех строк массива по функциональной клавише 'F6' завершить создание массива и по клавише 'Enter' записать его в заданный дисковый файл.

Описанная процедура с очевидными изменениями подходит и для создания СФД с помощью какого-либо текстового редактора. Данная процедура может быть легко погружена и в некоторую программу ввода и контроля данных, написанную на языке программирования. Тогда как ряд языков программирования, например Бейсик, позволяют непосредственно создавать ФД обоих типов, которые могут использоваться в среде пакета.

Способ организации файлов данных обоих типов определяет и взаимосвязь соответствующих им встроенных функций доступа. Так функция доступа READ применима и к СФД, возвращая массив неструктурированного типа. Тогда как в общем случае применение функции READPRN к НФД (не вызывая особых ситуаций) возвращает нулевое значение. Следующий фрагмент документа иллюстрирует использование функций доступа СФД, их некоторые взаимосвязи с функциями доступа НФД, а также примеры специального применения указанных функций доступа:

[H] Y := 1..3 X := 1..3 Z := 1..3 V1 := (42 47 23)

$$M = \begin{bmatrix} 1 & -7 & 2 \\ 2.1 & 8 & 2 \\ 8 & 9 & 6 \end{bmatrix} \quad V2 := \begin{bmatrix} 48 \\ 43 \end{bmatrix}$$

WRITEPRN (X) := V1 + X² APPENDPRN (Y) = M + 0 · Y

APPENDPRN (Z) := V2 + 0 · Z

A := READPRN (Y)

B := READPRN (Z)

$$\text{READPRN (X)} = \begin{bmatrix} 43 & 48 & 24 \\ 46 & 51 & 27 \\ 51 & 56 & 32 \end{bmatrix}$$

$$A^T = \begin{bmatrix} 1 & 2.1 & 8 & 1 & 2.1 & 8 & 1 & 2.1 & 8 & 1 & 2.1 & 8 & 1 & 2.1 & 8 & 1 & 2.1 & 8 \\ -7 & 8 & 9 & -7 & 8 & 9 & -7 & 8 & 9 & -7 & 8 & 9 & -7 & 8 & 9 & -7 & 8 & 9 \\ 2 & 2 & 6 & 2 & 2 & 6 & 2 & 2 & 6 & 2 & 2 & 6 & 2 & 2 & 6 & 2 & 2 & 6 \end{bmatrix}$$

Y1 := 1..6 APPEND (Y1) := B X1 := 1..9 WRITE (X1) := M

$$B = \begin{bmatrix} 48 \\ 43 \\ 48 \\ 43 \\ 48 \\ 43 \end{bmatrix}$$

READ (X1) + 0 · X1

READ (Y1) + 0 · Y1

1
-7
2
2.1
8
2
8
9
6

48
43
48
43
48
43

READPRN (X1) = 0

PRNCOLWIDTH := 6

PRNPRECISION := 3

```

READPRN (Y1) = 0      ORIGIN := 1      p := 2      i := 1..p      j := 1..p
READPRN [Y1_dat] = [ 48 49 50 51 52 53 54 55 56 57 49 50 51 52 ]
                    [ 56 57 58 59 51 52 53 54 55 56 57 58 59 60 ]
                    [ 55 56 57 58 59 60 61 62 54 55 56 57 58 59 ]
Gi,j = √i+j      APPENDPRN (X) := G      WRITE(G) := G
                    READPRN (X) = [ 1.4 1.7 ]      READ (G) = 1.414
PRNCOLWIDTH := 10      PRNPRECISION := 9      V1 := 19.4742
V2 := 1967.89      V3 := 48      V4 := -19.899      V5 := 20.089
WRITE (File) := V      WRITEPRN (G) := V      d := 1..4      X := 1..5
READ (File) + 0 . X

```

19.474
1967.89
6.928
-19.899
20.089

$$\text{READPRN (G)} = \begin{bmatrix} 19.474 \\ 1967.89 \\ 6.928 \\ -19.899 \\ 20.089 \end{bmatrix}$$

$$Z := \begin{bmatrix} 42.47 & 0 \\ -67 & -48.6 \end{bmatrix} \quad \text{WRITE (Z)} := Z$$

$$\text{READ(Z)} \cdot \frac{d}{d}$$

42.47
0
-67
-48.6

$$\text{WRITE (Z)} := Z_{i,j}$$

$$\text{WRITEPRN(G)} := Z_{i,j}$$

$$\text{READ (Z)} + 0 \cdot d$$

42.47
0
-67
-48.6

$$\text{READPRN (G)} = \begin{bmatrix} 42.47 \\ 0 \\ -67 \\ -48.6 \end{bmatrix}$$

$$\text{READ (Z)}^2 \cdot \frac{d}{d}$$

1803.701
0
4489
2361.96

$$\overline{[\sqrt{\text{READPRN (G)}}]} = \begin{bmatrix} 6.517 \\ 0 \\ 8.185i \\ 6.971i \end{bmatrix}$$

$$\overline{\sin (\text{READPRN (G)})} = \begin{bmatrix} -0.998 \\ 0 \\ 0.856 \end{bmatrix}$$

$$\overline{\ln(\text{READPRN}(G) + 1)} + \text{READPRN}(G) = \begin{bmatrix} 3.772 \\ 0 \\ 4.19 + 3.142i \\ 3.863 + 3.142i \end{bmatrix}$$

$$\overline{\ln(\text{READPRN}(G) + 1)} = \begin{bmatrix} 3.772 \\ 0 \\ 4.19 + 3.142i \\ 3.863 + 3.142i \end{bmatrix}$$

$$M := \begin{bmatrix} 1 & 4 \\ 2 & 5 \end{bmatrix}$$

$$\text{WRITEPRN}(F) := M$$

$$\text{WRITEPRN}(S) := \overline{\sin(\text{READPRN}(F))}$$

$$\text{READPRN}(S) = \begin{bmatrix} 0.841 & -0.757 \\ 0.909 & -0.959 \end{bmatrix}$$

$$\overline{\sin(\text{READPRN}(F))} = \begin{bmatrix} 0.841 & -0.757 \\ 0.909 & -0.959 \end{bmatrix}$$

[K]

$$\overline{\begin{bmatrix} M \\ \text{READPRN}(S) \end{bmatrix}} = \begin{bmatrix} 1.188 & -5.285 \\ 2.199 & -5.214 \end{bmatrix}$$

Приведенный пример достаточно прозрачен и с учетом сказанного особых пояснений не требует. Отдельные фрагменты примера могут натолкнуть читателя на другие интересные подходы к использованию встроенных функций доступа по организации работы с данными в среде пакета. Следует, однако, иметь в виду наличие определенных различий обычных встроенных функций пакета и встроенных функций доступа к данным любого из рассмотренных двух типов. Поэтому каждое нестандартное использование их в вычислительных конструкциях требует предварительной проверки на контрольных примерах.

Следует отметить, что в общем случае использование функций доступа к данным в решающих блоках некорректно, о чем свидетельствует следующий простой фрагмент документа:

$$[H] \quad K := 1 \quad X := 19.47$$

$$M := \begin{bmatrix} 52 \\ 71 \end{bmatrix}$$

$$\text{Given} \quad \text{WRITE}(D1) := X \quad K^* + K \approx 6 \quad \text{WRITEPRN}(R2) := M$$

$$[K] \quad \text{Find}(K) = 1.63437 \quad \text{READ}(D1) = 0 \quad \text{READPRN}(R2) = 0$$

Опыт эксплуатации пакета показал недостаточно эффективную систему доступа к данным в среде пакета, не ориентированную на массовую обработку данных. При решении вычислительных задач, оперирующих большими массивами данных сложной организации и структуры, рекомендуется на определенных этапах технологического процесса подключать другие программные средства, после отработки которых, передавая подготовленные файлы рассмотренного типа (НФД и СФД) для последующей обработки в среду пакета, который для этих целей является более предпочтительным.

ТЕКСТОВАЯ ИНФОРМАЦИЯ В СРЕДЕ ПАКЕТА MathCAD РЕДАКТИРОВАНИЕ, ФОРМАТИРОВАНИЕ И ПЕЧАТЬ ДОКУМЕНТОВ

Работу с текстовой информацией пакет MathCAD поддерживает на двух уровнях: **текстовая область** и **текстовая зона**. Уровень текстовой области предполагает использование текста небольшого объема, помещаемого в документ, как правило, в качестве различного рода комментариев среди областей выражений и графиков. Текстовая область создается по клавише двойных кавычек (") и основы ее использования рассматривались в разделе 2.2. Здесь отметим лишь некоторую дополнительную информацию, полезную для практического использования в документе текстовых областей.

Каждая текстовая область имеет **границу переноса**, по достижении которой текст автоматически переносится на следующую строку области. При создании текстовой области ее граница переноса по умолчанию зависит от текущей длины строки, равной 78 байтам. Для изменения границы переноса можно использовать команду 'Line-length', описываемую ниже. Устанавливая границу переноса в текстовой области, пакет руководствуется следующим правилом.

Пусть L — глобальная длина строки текста пакета (по умолчанию $L = 78$, столбцы $1 - L$). Если текстовая область начата между позициями 1 и $L-4$, то граница переноса полагается равной $L + 1$. Если же текстовая область начата за позицией $L-4$, то пакет устанавливает границу переноса так, что протяженность текстовой области составляет L байтов. В дальнейшем под **протяженностью** будем понимать длину строки текстовой области или зоны в байтах. Таким образом, каждая текстовая область может иметь свою протяженность. Перенос на границе текст, пакет, по возможности, разрывает его на пробелах, являющихся разделителями слов. Набор слов текста, завершающийся символом $\{. ! | ?\}$, образует **предложение**. На границе области слово переносится на следующую строку, в случае наличия слова во всю строку оно идентифицируется на границе переноса дефисом (-) и продолжается в следующей строке. При редактировании текста области пакет производит автоматическое выравнивание строк в соответствии с ее текущей границей переноса. Для выравнивания всех строк текста следует использовать команду

JUSTIFY | Ctrl + N | T J

Перед использованием данной команды курсор необходимо устанавливать в требуемую текстовую область документа.

Для установки новой границы переноса текстовой области используется команда

WIDTH [⟨длина строки⟩] | — | T W

определяющая протяженность выбранной области в байтах. Перед ее использованием курсор необходимо установить в нужную тексто-

вую область документа. После ввода данной команды происходит при необходимости автоматический перенос в строках текста области. Если команда 'Width' введена без параметра, то в ССП выводится текущая протяженность ('Width = nn') указанной области и предоставляется возможность изменить ее. При необходимости оперативно сузить протяженность области достаточно установить курсор в нужную позицию текста и нажать клавиши 'Ctrl + Enter'. Пакет устанавливает новую протяженность области и производит автоматический перенос слов текста относительно новой границы переноса. Пакет поддерживает возможность использования в текстовой области **разделительных** (пустых) строк, которые образуются нажатием клавиши 'Enter' и на период нахождения курсора в области идентифицируются слева треугольниками (◀). Часть текста между двумя пустыми строками будем называть **абзацем**, который должен завершаться двумя пустыми строками или пустой строкой, следующей за пробелом. Читателю рекомендуется на примере некоторой текстовой области проверить работу описанного механизма компоновки и выравнивания текста в области.

8.1. ИСПОЛЬЗОВАНИЕ ТЕКСТОВЫХ ЗОН В ДОКУМЕНТЕ

Текстовая зона в отличие от области характеризуется тем, что занимаемые ею строки документа отводятся полностью под текст. В этом плане текстовая зона допускает работу в ней в режиме текстового процессора и используется, как правило, для организации в документе больших текстовых массивов, что характерно для различного рода научно-технических отчетов, рукописей книг и статей, насыщенных математическими формулами и графиками. Основные отличия текстовой зоны от области состоят в следующем:

- зона занимает сплошной участок полных строк документа и в ней **непустимы** области выражений и/или графиков; при попытке поместить в нее, например, выражение по команде 'Paste' (клавиша 'F4') пакет в ССП выводит диагностическое сообщение 'Would overlap existing region' и команда игнорируется;

- зона при своем расширении по мере увеличения текста раздвигает документ, выталкивая следующие за ней области и другие зоны к его концу; и наоборот, при уменьшении текстовой зоны освобождающееся место в документе заполняется следующими за ней областями и зонами;

- по клавишам 'Ctrl + V' текстовая зона в отличие от областей выделяется вертикальной линией в первом своем столбце.

При этом пакет осуществляет реальное перемещение областей в документе только по клавишам 'Ctrl + N' или после вывода курсора из текстовой зоны, а до этого вводимый текст на экране накладывается на следующие за зоной области и другие зоны.

С точки зрения сказанного структура документа (рукопись книги, статьи, отчета и т. д.) могла бы иметь следующий вид. Назовем **экстентом** множество подряд идущих строк документа. Тогда документ можно представить себе как чередующиеся в произвольном по-

рядке экстенды двух типов: 1) текстовые зоны и 2) экстенды, содержащие в произвольном порядке и составах вычислительные, графические и текстовые области (рис. 19).

Для создания текстовой зоны курсор устанавливается на свободную строку документа (необязательно в его конце), а затем вводится команда

TEXTBAND | Ctrl + T | T T

В результате выполнения команды первая пустая строка зоны берется в двойные кавычки (""), курсор устанавливается в ее первый столбец, а пользователь получает возможность вводить текст подобно случаю текстовой области с описанными отличиями. Если, ничего не записав в созданную зону, вывести из нее курсор, то с экрана она исчезнет,

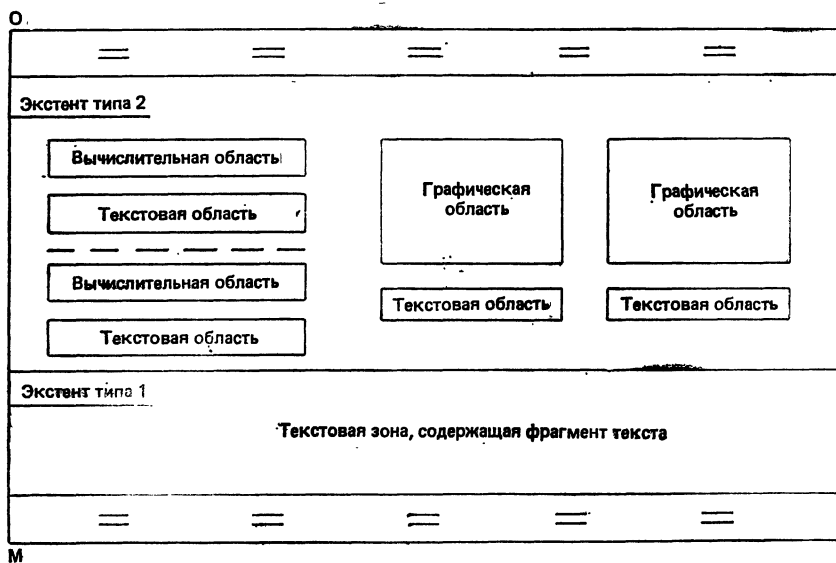


Рис. 19. Примерная структура документа для оформления материалов научно-технического характера к печати

оставаясь зарезервированной в документе и визуализируясь по мере возвращения в нее курсора. В алфавит текстовой области/зоны могут входить любые ASCII-символы, а также 17 греческих букв и буквы национальных алфавитов, допускаемые клавиатурой международного стандарта. Для символов, отсутствующих в ASCII-стандарте, в кодовой таблице ПК отводятся позиции в диапазоне 128—175 (расширенный набор символов IBM). Отклонение от данного принципа для отечественного ПК ИСКРА 1030 создало ряд затруднений при использовании на нем оригинального зарубежного ПО, а также явилось одной из причин отсутствия полной совместимости с известной серией IBM PC/XT [10].

Если используется клавиатура международного стандарта, то вводить национальные буквы можно простым нажатием соответству-

ющих клавиш, что обеспечивается драйвером клавиатуры, встроенным в поставляемую с ПК версию MS DOS или OS/2. При использовании клавиатуры и драйвера IBM национальные буквы вводятся путем фиксации клавиши 'Alt' и ввода трехцифрового кода (128—175) соответствующей буквы; цифры кода набираются на цифровой части клавиатуры (прил. 4). Набрав код символа и отпустив клавишу 'Alt', помещаем нужный символ в требуемое место текста. Граница переноса для зоны совпадает с концом строки документа и может быть изменена командой

LINELENGTH [⟨длина строки⟩] | — | W L

Параметр команды определяет новую длину строки в байтах. В случае его отсутствия пакет выводит в ССП текущее значение длины ('Linlength = пп'; по умолчанию пп = 78) и дает возможность изменить его. Следует помнить, что команда 'Linlength' действует только на текстовые зоны и не влияет на расположение информации в областях любого типа. После ввода команды 'Linlength' происходит автоматическая перекомпоновка строк в зонах с учетом новой границы переноса. Аналогично этому команда 'Width' действует только на текстовые области, не затрагивая зон. При этом, если команда 'Width' действует только на область с установленным в ней курсором, то 'Linlength' — на все зоны документа независимо от местоположения в нем курсора. При попытке применить команду 'Width' к зоне команда игнорируется с выдачей в ССП сообщения 'Cursor is in text band, not region'. Тогда как применение команды 'Linlength' к текстовой области также игнорируется относительно области, но действует на все находящиеся в текущем документе текстовые зоны. Подобно случаю текстовой области в зоне допускается использование разделительных линий и абзацев. После текстовой зоны пакет предусматривает свободные строки документа, отделяющие ее от следующих за ней областей и других зон.

В среде пакета в текстовой зоне можно готовить и редактировать исходные тексты программ (например, БЕЙСИК-программ) для той или иной системы программирования с последующим помещением их в текстовой ASCII-файл. Но так как такой файл создается в MCD-формате, то в нем первые строки, идентифицируемые слева наборами «.MCD» и «.CMD», представляют собой стандартный заголовок MCD-файла, который перед использованием данного файла в качестве исходного программного модуля следует удалять, например, текстовым процессором.

В ряде случаев целесообразно готовить большие текстовые области документа в среде некоторого текстового редактора либо извлекать их из уже существующих текстовых ASCII-файлов. Для этого можно воспользоваться следующей простой технологией. По команде 'ConfigSave <СФ>' создается на диске пустой MCD-файл, содержащий только стандартный заголовок, а затем текстовым редактором дополняется двумя строками вида

.TXT N O L 32003

bL, 32002, Ln, P

где N — номер строки документа, определяющий начало текстовой зоны (ТЗ); L — количество строк ТЗ; L_p — длина строки ТЗ, устанавливаемая по умолчанию ($L_p = 78$) либо по команде пакета 'Linlength'; P — общий объем ТЗ ($P = \sum S_j + 2L - 1$, S_j —

длина j -строки; $j = \overline{1, L}$). Значения для параметров N , L и P легко вычисляются на основе создаваемого или используемого текстового массива и в любой момент могут соответствующим образом корректироваться. Наконец, в созданный таким образом MCD-файл помещается требуемый текстовой ASCII-массив, результирующий MCD-файл можно помещать в заданное место текущего документа.

8.2. КОМАНДЫ ПЕРЛЮСТРАЦИИ И РЕДАКТИРОВАНИЯ ТЕКСТА

Рассматриваемые ниже команды перлюстрации служат для организации быстрого просмотра элементов текста и одинаково работают как в текстовых областях, так и в зонах. Однако ввиду значительных, порой, размеров текстовых зон именно в них они представляются нам наиболее актуальными, так как в текстовых областях перемещение курсора, как правило, легко осуществляется посредством КУК. Для сокращения количества команд используются две специальные команды направления поиска,

FORWARD | Ctrl + F | T F
BACKWARD | Ctrl + B | T B

которые определяют соответственно направление {вперед|вниз} и {назад|вверх} для команд перлюстрации, причем по умолчанию полагается направление {вперед|вниз} в документе. Команды перлюстрации представлены в табл. 13, из которой легко вытекает

Таблица 13

Команда	Клавиши	Назначение: перемещение курсора
SKIPWORD	Ctrl + W	На следующее слово текста
SKIPLINE	Ctrl + L	В конец текущей строки текста
SKIPSENTENCE	Ctrl + S	На следующее предложение текста
SKIPPARAGRAPH	Ctrl + P	На один абзац текста

их назначение и принцип использования. Команды перлюстрации можно вводить по их именам или посредством соответствующих им клавиш; в командном меню они отсутствуют. Простые примеры иллюстрируют сказанное:

Использованные клавиши	Результат перемещения курсора
Ctrl + B; Ctrl + P	В начало следующего абзаца
Ctrl + S; Ctrl + W (3 раза)	На четвертое слово следующего предложения
Ctrl + B; Ctrl + L (5 раз)	На пять строк вверх

Перлюстрацию текста как в области, так и в зоне можно осуществлять и по клавишам 'Tab' и 'Shift + Tab', позволяющим переме-

щать курсор соответственно вперед и назад на одно слово текста, минуя пустые строки. Однако на них команды 'Forward' и 'Backward' не действуют.

Как и в случае текстовой области, в зоне пакет предлагает ряд клавиш и команд для редактирования текста. Для удаления отдельного символа или небольшого участка символов можно использовать клавиши {Del | Bksp}. Легко осуществляется и вставка символов в любое место текста. Для оперирования с более крупными частями текста вводится понятие **помеченного блока**. Помеченный блок можно удалять, перемещать или копировать в ВПП. Для создания помеченного блока служит команда

TEXTMARK | Ctrl + X | T M

которая используется следующим образом. Курсор устанавливается на начало блока, нажимаются клавиши 'Ctrl + X', в результате чего символ в позиции курсора получает обратную подцветку (в монохромном режиме становится черным на ярко-белом фоне). Затем, не выходя за пределы текстовой зоны или области, курсор устанавливается в конец помечаемого блока, снова нажимаются клавиши 'Ctrl + X'. В результате данной процедуры формируется помеченный блок текста, визуальнo выделяемый подцветкой. На самом деле при копировании или удалении блока текста необязательно специально указывать оба его конца. Если указан один конец блока, то после использования команд 'Incopy' или 'Incut' пакет перемещает часть текста, расположенную между указанным концом и текущим положением курсора. При необходимости отменить помеченный с двух сторон блок достаточно еще раз нажать клавиши 'Ctrl + X' или вывести курсор из текстовой области или зоны. Таким образом, в тексте в каждый момент можно оперировать только с одним помеченным блоком.

С помеченным блоком можно проводить ряд операций по редактированию текста посредством команд 'Incopy', 'Incut' и 'Inpaste', рассмотренных в разделе 2.7. Здесь лишь напомним их назначение. Команда 'Incopy' копирует помеченный блок в ВПП без его удаления из текста, тогда как по команде 'Incut' копируемый блок удаляется из текста. По команде 'Inpaste' сохраняемый в ВПП помеченный блок помещается в текстовую зону или область по местоположению курсора. Курсор может устанавливаться в любую текстовую зону или область документа, а копируемый по его местонахождению блок производит по мере необходимости автоматическую перекомпоновку строк текста согласно текущей границе переноса. При этом по команде 'Inpaste' помеченный блок можно поместить только в существующую текстовую зону или область. Попытка поместить его в свободную область документа не дает результата и команда 'Inpaste' игнорируется, поэтому при необходимости поместить блок на свободное место документа необходимо предварительно создать в этом месте пустую текстовую зону или область.

При использовании команд 'Textmark', 'Incopy', 'Incut' и 'Inpaste' следует иметь в виду их принципиальное отличие от по-

добных им команд 'Mark', 'Copy', 'Cut' и 'Paste'. Если объектом действия команд второй группы являются целые области различного типа или текстовые зоны, то для команд первой группы в качестве объектов выступают либо помеченные блоки в текстовых областях и зонах, либо отдельные элементы выражений. Общий же принцип применения к объектам команд обеих групп совпадает, не считая того, что для команд первой и второй групп пакет отводит различные типы ВПП.

Для центровки строк текста в зоне или области используется команда

CENTER | — | T C

результат применения которой сводится к следующему. Если курсор установлен на пустую строку текста, то по команде 'Center' курсор вместе с идентифицирующим ее левым символом (◀) устанавливается в центр строки. При наличии текста в строке он центрируется, а курсор устанавливается в конец строки. Следует помнить, что центровка осуществляется относительно строки области или зоны, которые в общем случае для текущего документа (как следует из сказанного) не совпадают. Последующее редактирование текста (вставки, удаления) требует новых центровок соответствующих строк, так как центровка не сохраняется при таком редактировании текста. Чтобы не приводить громоздкие примеры, поясняющие сказанное о редактировании в текстовых областях и зонах, читателю рекомендуется на ПК практически апробировать различные варианты редактирования текстов, получив при этом соответствующий навык для данного типа работы с пакетом.

Рассмотрим теперь подробнее вопросы редактирования, форматирования и печати документов, которые были недостаточно отражены при первом рассмотрении в предыдущих главах. Как и выше, строки документа нумеруются сверху вниз, начиная с нулевой, а столбцы в строках нумеруются слева направо, также начиная с нулевого. Для управления перемещением курсора внутри текущего документа (в областях, зонах, вне их) используются КУК, клавиши 'Tab', 'Enter', а также комбинации ряда других клавиш, сводка и назначение которых приведены в прил. 6. Наряду с перечисленными средствами по управлению курсором пакет располагает двумя командами

GOTO [⟨Строка⟩ [⟨Столбец⟩]] | — | E G

MOVE [⟨Строка⟩ [⟨Столбец⟩]] | — | E M

предназначенными для установки курсора в документе на требуемые строку и столбец в ней. При отсутствии у команд обоих параметров пакет выдает запрос на их определение, ответ на который клавишей 'Enter' отменяет команду и не меняет текущего местоположения курсора. Если же указан только первый параметр 'строка', то курсор устанавливается в начало определяемой параметром строки (столбец 0), номер которой вычисляется в зависимости от использованной команды управления курсором (GOTO или MOVE).

По команде 'GOTO' курсор устанавливается на указанную первым параметром строку в столбец, определяемый ее вторым параметром (при его отсутствии столбец полагается нулевым). Например, по команде 'Goto 120 48' курсор устанавливается в 48-й столбец 120-й строки документа. При этом результат выполнения команды 'Goto A B' визуально представляется как установка окна экрана своим левым верхним углом в точку (A, B) текущего документа. Значения параметров команды 'Goto A B' могут быть положительными и отрицательными, определяя результат ее выполнения согласно табл. 14. При использовании команды 'Goto' без параметров пакет предлагает определить их по запросу 'Go to:'. Ответ клавишей 'Enter' сохраняет текущее местоположение курсора в документе.

Команда 'Move' до некоторой степени аналогична предыдущей команде, но в отличие от 'Goto' ее параметры определяют не абсолютные номера строки и столбца, а их смещения относительно текущего местоположения курсора. При этом знак минус определяет смещение курсора вверх (для строк) или влево (для столбцов), а знак плюс — соответственно смещение вниз и вправо. Например, если курсор находится в 50-м столбце 50-й строки документа, то по команде 'Move — 10 —5' курсор устанавливается в 45-й столбец 40-й строки, а верхний левый угол окна экрана совмещается с новым местоположением курсора в документе. Если в результате вычисления нового положения курсора в документе значение какой-либо его координаты становится отрицательным, то она полагается нулевой. Например, результатом применения команды 'Move — 48 —43' в случае нахождения курсора в 128-м столбце 45-й строки документа будет установка курсора в 85-й столбец нулевой строки текущего документа. При вводе команды 'Move' без параметров пакет запросом 'Move to: 0 0' позволяет либо ввести необходимые значения для параметров, либо по клавише 'Enter' определить смещения нулевыми, тем самым не меняя текущего местоположения курсора в документе. При наличии пакета читателю рекомендуется практически апробировать результаты выполнения команд 'Goto' и 'Move' по управлению курсором.

Таблица 14

Знак параметра		Номер	
A	B	стро-ки	пози-ции
+	+	A	B
—	+	O	B
+	—	A	O
—	—	O	O
—	Нет	O	O
+	Нет	A	O

8.3. РЕДАКТИРОВАНИЕ ТЕКУЩЕГО ДОКУМЕНТА

Выше (разделы 2.7, 8.2) рассматривались вопросы редактирования выражений и текстовой информации на уровне соответствующих им областей и зон. И в этом плане данные вопросы относятся к обсуждаемой в настоящем разделе тематике. Однако пакет располагает средствами редактирования документа и более общего уровня, о которых здесь и будет идти речь. Прежде всего по командам пакета

INSERTLINE | Ctrl + F9 | E I
 DELETLINE | Ctrl + F10 | E D

можно соответственно вставлять или удалять пустые строки документа, определяемые местоположением курсора. При этом удаляется только пустая строка, на которой установлен курсор, и вставляется пустая строка после строки, в которой установлен курсор. Строку, содержащую информацию, нельзя удалить по команде 'Deleteline'. Данные команды позволяют достаточно произвольно расширять и уплотнять текущий документ.

Для копирования и удаления областей, зон, а также помеченных блоков документа пакет располагает группой следующих команд:

COPY		F2		E C
CUT		F3		E C
PASTE		F4		E P
MARK		Ctrl + Y		E M

Первые три команды обсуждались выше и предназначены соответственно для копирования (без удаления), копирования (с удалением) в ВПП, а также копирования из ВПП в документ области любого типа, текстовой зоны или помеченного блока документа. Так как ВПП может хранить только одну копию, то по команде 'Paste' можно вернуть в текущий документ только последнюю, помещенную по команде 'Copy' или 'Cut', конструкцию. Однако копию из ВПП можно помещать в несколько мест текущего документа.

Наряду с отдельными областями и зонами команды 'Copy', 'Cut' и 'Paste' позволяют оперировать и с помеченными блоками документа, которые определяются командой 'Mark' пакета. Для определения в документе помеченного блока курсор устанавливается на первую область (или зону), включаемую в блок, и нажимаются клавиши 'Ctrl + Y', та же процедура продлевается и с последней областью помечаемого блока документа. При формировании в текущем документе помеченный блок можно выделять, начиная с последней области (зоны) и кончая первой, т. е. порядок отметки ограничивающих блок областей (зон) не играет роли. В результате отметки начала и конца помеченного блока все входящие в него области обводятся на экране прямоугольниками, а текстовые зоны слева отмечаются прямыми линиями (подобно тому как это делается для всего документа по клавишам 'Ctrl + V'). В текущем документе в каждый момент может быть не более одного помеченного блока. Помеченный блок документа отменяется по клавишам 'Ctrl + Y' или посредством КУК, а также при смене окон экрана (по клавише 'F8') или после копирования его в ВПП.

Созданный помеченный блок можно перемещать в ВПП и обратно в документ аналогично перемещению областей и зон. Помеченный блок можно передавать через ВПП как документу в другом окне экрана, так и вновь загруженному документу, если предыдущий документ сохранил его в ВПП и не производилась перезагрузка пакета. Следует помнить, что при копировании помеченного блока из ВПП в текущий документ по команде 'Paste' не диагностируется пакетом случай наложения копируемого блока на уже существующие области

документа, что предполагает особое внимание при проведении подобных операций. Для отмены нежелательной команды 'Paste' достаточно нажать клавишу 'F3', чтобы вернуть помеченный блок обратно в ВПП. При копировании помеченного блока в ВПП местоположение курсора в документе несущественно, т. е. при наличии в документе помеченного блока по командам 'Copy' и 'Cut' в ВПП копируется только блок данного типа. Поэтому перед копированием в ВПП других конструкций документа необходимо по клавишам 'Ctrl + Y' отменить помеченный блок.

В результате перемещения и редактирования областей и зон текущего документа, а также в случае возникновения ошибок и появления соответствующих им сообщений возможно наложение одних областей документа на другие. Такие наложения областей не влияют на корректность вычислений в документе, но делают его неудобным (а в ряде случаев и непригодным) для визуального использования на экране. Для устранения всех наложений в документе следует использовать команду пакета

SEPARATE | — | E S

При использовании данной команды возможны различного рода побочные эффекты, так как в результате ее выполнения перемещается большое число областей и зон в большом по объему документе. Поэтому для возможности контроля над процессом разделения областей можно использовать команду пакета 'Insertline', которая позволяет выборочно разделять наложенные друг на друга области документа. Так, в случае разделения областей и связанных с ними диагностических сообщений рекомендуется использовать именно команду 'Insertline', так как при использовании в этих целях команды 'Separate' после устранения вызвавших эти сообщения причин между областями документа останутся большие свободные области, для чего понадобится уплотнять документ.

Появление диагностических сообщений при достаточно плотном расположении областей в документе, даже при последующем устранении вызвавших их причин, может привести к искажению информации на экране. Для восстановления содержимого экрана используется команда пакета

REDRAW | Ctrl + R | S R

Данную команду рекомендуется использовать при искажении содержимого экрана также по другим причинам.

Одним из приемов редактирования текущего документа является поиск и/или замена заданного текста на указанный контекст, что весьма актуально при редактировании большого и содержащего различного типа информацию документа. Для этих целей пакет предлагает следующие две команды:

SEARCH [—] [Текст] | Ctrl + F5 | E F

REPLACE [Текст] | Ctrl + F6 | E R

позволяющие соответственно осуществлять поиск, а также поиск и/или замену указанной параметром 'Текст' цепочки символов, находящейся в любом месте текущего документа. При этом в области выражений параметр 'Текст' рассматривается как имя переменной, массива или функции, а в текстовой области или зоне как обычная цепочка символов. При этом в параметре 'Текст' обеих команд задаваемые цепочки символов должны быть буквозависимыми, т. е. следует строго различать прописные и строчные буквы при реализации определяемых командами операций поиска. Если у команд отсутствует параметр 'Текст' или команды вводятся не в командном режиме, то пакет запрашивает значения для их параметров.

По команде 'Search' осуществляется поиск цепочки символов (определяемой ее параметром или ответом на запрос 'Search for:' пакета), начиная с текущего положения курсора к концу или началу документа в зависимости от знака перед указанной цепочкой символов. Если перед цепочкой символов указан знак минус (—), то поиск осуществляется от курсора к началу документа, в противном случае от курсора к концу текущего документа. Найдя первое вхождение искомой цепочки символов, пакет прекращает поиск, устанавливает курсор на найденную цепочку и запрашивает необходимость продолжения поиска 'Press [+] or [—] to continue search'. Ответив нажатием клавиши (+), (—) или КУК, пользователь санкционирует соответственно продолжение поиска в прямом, обратном направлении или его прекращение. Если искомый текст не найден, то в ССП выводится сообщение 'Not found'. Например, по команде 'Search — Grodno' в обратном направлении (от курсора к началу документа) отыскивается слово 'Grodno', затем пакет запрашивает дальнейшие действия: продолжение поиска в нужном направлении или прекращение его.

По команде 'Replace' осуществляется поиск цепочки символов, определенной параметром 'Текст' (или ответом на запрос 'Search for:'), и санкционированная замена найденной цепочки на текст, определенный в ответе на запрос 'Replace with:' пакета. Подобно случаю команды 'Search' поиск искомой цепочки по команде 'Replace' осуществляется во всех областях и зонах текущего документа, но в отличие от команды 'Search' поиск искомой цепочки ведется только в прямом направлении, т. е. от текущего положения курсора к концу документа. Найдя первое вхождение искомой цепочки символов, пакет устанавливает на нее курсор и запрашивает дальнейшие действия согласно подсказке 'Yes, No, Quit or All (y | n | q | a)?'. В зависимости от ответа пакет производит следующие действия:

Yes (y) — заменяет найденную цепочку и переходит к поиску следующего ее вхождения в документ; при отсутствии искомой цепочки выводится сообщение 'Not found' и поиск завершается;

No (n) — пропускает найденную цепочку и переходит к поиску следующего ее вхождения в документ;

Quit (q) — найденная цепочка не заменяется и дальнейший поиск прекращается;

All (a) — осуществляется несанкционированная замена найденной

и всех последующих вхождений искомой цепочки до конца текущего документа.

Например, по команде 'Replace Grodno' в СПП выводится запрос 'Replace with:' и по ответу 'Tallinn' осуществляется замена всех вхождений в документ (начиная с текущей позиции курсора) слова 'Grodno' на слово 'Tallinn', если на запрос пакета ответить клавишей 'a' (All).

В заключение раздела рассмотрим вопросы редактирования двух документов в среде пакета одновременно, используя возможности пакета по разделению экрана на два окна (**основное и дополнительное**). Для работы в многооконном режиме пакет располагает следующими тремя командами:

SPLIT	F7	W S
UNSPLOT	Ctrl + F7	W U
SWITCH	F8	W J

которые предназначены соответственно для создания дополнительного окна экрана, отмены дополнительного окна и переключения с одного окна экрана на другое. После своей загрузки пакет находится в однооконном режиме, при этом окно совпадает со всем экраном. После установки курсора на требуемую строку экрана (как правило на строку 11 — середину экрана) нажимается клавиша 'F7' или вводится команда пакета 'Split'. Пакет создает дополнительное окно, в котором в качестве СПП выбирается выбранная строка экрана. Вновь созданное по команде 'Split' окно экрана становится активным. В дальнейшем с окном экрана будем ассоциировать сканируемый им документ. При этом дополнительное окно экрана всегда создается в неавтоматическом режиме (М-режим) и для перевода его в автоматический режим (А-режим) следует ввести команду 'Auto'. При этом, если удалено дополнительное окно экрана в А-режиме, то, создавая все последующие дополнительные окна экрана, пакет будет формировать их уже в А-режиме и также активными. Основное окно экрана после создания дополнительного становится неактивным и содержит документ, с которым можно продолжать работу путем переключения окон по команде 'Switch'.

Дополнительное окно экрана в логическом плане ничем не отличается от основного окна экрана, в нем можно работать с другим документом аналогично основному окну (ввести или создать новый документ, редактировать его, выполнять вычисления, работать с данными и т. д.). Работа в обоих окнах экрана происходит независимо друг от друга, что дает возможность, поочередно работая в двух окнах с одним и тем же документом, создавать две различные его версии и проводить визуальное сравнение результатов вычислений в обеих версиях. Пакет не допускает использования более двух окон экрана и при попытке определить третье окно экрана пакет просто изменяет размеры основного и дополнительного окон, перемещая в текущее местоположение курсора ССП дополнительного окна.

По команде 'Unsplit' происходит удаление дополнительного окна экрана с запросом пользователю о необходимости сохранения в дис-

ковом файле результатов работы с документом в удаляемом окне. В двухоконном режиме пакет позволяет производить целый ряд совместных работ с разными документами, ассоциированными с окнами экрана. Так, загрузив в каждом из окон экрана различные документы, посредством команд пакета 'Copy', 'Incopy', 'Paste' и 'Inpaste' можно копировать элементы (целые области, зоны, блоки или отдельные их части) из одного документа в другой, и наоборот. Это обеспечивается возможностью доступа к ВПП из любого активного документа и сохранением содержимого любого типа ВПП при переключениях с одного окна на другое. Данная возможность может быть весьма полезна при необходимости использования в различных документах или группах документов фрагментов из некоторого архивного файла, содержащего конструкции массового характера (системы единиц измерений, функции пользователя, выражения, определения, текст и т. д.). Наряду с этим следует иметь в виду, что обработка документов в различных окнах экрана не является полностью независимой. Так, выполнение команд пакета: 'Format', 'Plotformat', 'Surfaceformat', 'Sketchformat', 'Set', 'Redraw', 'Selectprinter', 'Dimensions', 'Memory' и 'Randomize' в любом из окон экрана одинаково влияет на обработку документов в обоих окнах. Например, по команде 'Memory' выводится объем памяти, занимаемый документами из обоих окон экрана. В основу выбора набора команд пакета, действующих одновременно на оба окна экрана, положен тот принцип, что глобальные характеристики пакета должны быть универсальны для одновременно обрабатываемых в среде пакета документов. Остальные виды обслуживания пакетом могут быть дифференцированы для каждого обрабатываемого документа индивидуально. Данный подход позволяет весьма разумно сочетать основные концепции пакета и возможности индивидуального подхода при одновременной обработке документов в среде пакета.

8.4. ФОРМАТИРОВАНИЕ И ПЕЧАТЬ ТЕКУЩЕГО ДОКУМЕНТА

Основные вопросы, связанные с выводом текущего документа на печать, рассматривались в разделах 1.4 и 2.9, где были представлены две команды пакета

SELECTPRINTER	[<N>]		—		S S
PRINT			—		S P

предназначенные соответственно для выбора типа принтера или плоттера и собственно самой операции вывода на принтер, плоттер или в дисковый PRN-файл, т. е. файл, полностью готовый к выводу на устройство печати посредством, например, системной команды 'Copy' или 'Print'. При этом для ряда типов принтеров, например Robotron CM 6329.02—M, команда 'Copy' даже оказывается предпочтительнее команды 'Print' системы MS DOS. Однако представленная в этих разделах информация достаточна только для типовых выводов документа в том виде, как он сформирован в среде пакета без какого-либо его форматирования и не включает такие важные вопросы, как: форма-

тирование выводимого документа, использование специальных ASCII-последовательностей для динамического изменения формата отдельных частей выводимого документа и использование плоттера в качестве устройства вывода. Данный раздел и восполняет этот недостаток.

Для форматированного вывода текущего документа или документа, помещаемого в PRN-файл, пакет располагает четырьмя командами для подготовки выходного документа

MARGIN [<Столбец>]	—	W	M
PAGELNGTH [<Длина>]	—	W	P
PAGEBREAK	—	W	I
BREAKPAGES	—	W	B

которые предназначены соответственно для установки левой границы документа, указания длины выводимой страницы, установки линии раздела страниц документа и режима переноса на следующую страницу областей графиков различных типов и выражений, если они пересекаются с линией раздела страниц.

Команда 'Margin' в зависимости от значения параметра 'Столбец' устанавливает левую границу вывода документа на принтер. По умолчанию для параметра полагается нулевое значение и вывод всех строк документа, начиная с нулевой колонки принтера. При отсутствии параметра пакет в ССП указывает его текущее значение ('Left margin = 0') и предоставляет возможность либо сохранить его, либо заменить новым. Например, по команде 'Margin 7' пакет осуществляет вывод текущего документа со сдвигом на семь колонок вправо от начала строки принтера, т. е. на печать документ выводится с левыми полями шириной в 7 колонок строки принтера.

По умолчанию пакет выводит на печать документ сплошным образом без разбиения на страницы. Команда 'Pagelength' задает своим параметром 'Длина' длину выводимой страницы документа в строках. На экране граница деления страниц показывается пунктирной линией, количество которых равно или на единицу меньше количества страниц документа. Такая разметка осуществляется на протяжении всего документа, включая и его пустую часть. Это позволяет формировать документ для вывода, учитывая расположение в нем информации. На печать линия раздела страниц не выводится. Нулевое значение параметра команды 'Pagelength' принимается по умолчанию и эквивалентно его отсутствию, т. е. осуществляется непрерывный вывод документа сплошным образом. Если команда 'Pagelength' введена без параметра, то в ССП указывается текущая длина страницы ('Page length=0') и предоставляется возможность сохранить ее или изменить. Например, по команде 'Pagelength 55' длина страницы вывода документа устанавливается в 55 строк.

Значения по умолчанию параметров для команд 'Margin' и 'Pagelength' устанавливаются в файле MCAD.MCC конфигурации пакета. При необходимости изменить их можно воспользоваться любым текстовым редактором, допускающим обработку текстовых ASCII-файлов. Поэтому, если пакет эксплуатируется с конкретными требова-

ниями по оформлению форматов выводимых документов, то во избежание частой форматизации документов информацию для ее проведения можно поместить в файл конфигурации пакета, что делает ее доступной по умолчанию.

В отличие от команды 'Pagelength', устанавливающей линию раздела страниц относительно начала документа с постоянным смещением (равным значению ее параметра) на протяжении всего документа, команда пакета 'Pagebreak' позволяет устанавливать в документе так называемую **жесткую линию раздела** страниц по месту, указываемому текущим положением курсора. В этом смысле жесткая линия раздела является частью документа и при корректировке подобна обычной области или зоне документа. Для задания жесткой линии раздела курсор устанавливается в нужную строку документа, а затем вводится команда 'Pagebreak'. В результате выполнения команды на экране появляется сплошная жирная линия раздела страниц со стрелкой в ее начале, указывающей на начало строки раздела документа. При выводе документа на печать на этой линии завершается очередная страница и начинается новая. Прогон бумаги на принтере между страницами для линии раздела составляет 28 строк, для жесткой линии — 32 строки устройства.

При наличии в документе жесткой линии раздела существующая или создаваемая разметка страниц по команде 'Pagelength' осуществляется следующим образом. До жесткой линии раздела проводятся пунктирные линии раздела согласно заданной длине страницы, начиная с нулевой строки или с предыдущей жесткой линии раздела, а за ней отсчет страниц и проведение пунктирных линий раздела производятся заново. При выгрузке документа в дисковый PRN-файл или MCD-файл форматизация его сохраняется. Для удаления жесткой линии раздела курсор устанавливается в ее начало и вводится команда пакета 'Cut' ('F3').

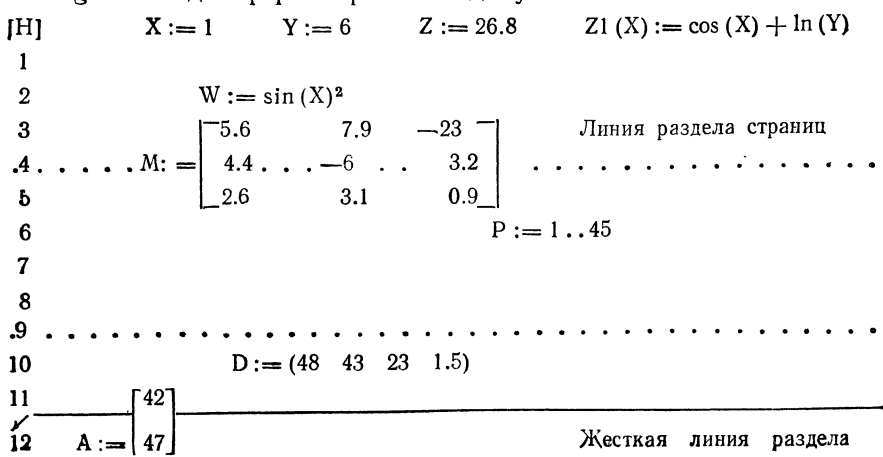
После удаления жесткой линии раздела пакет производит переразметку страниц документа (если использовалась команда 'Pagelength'), начиная с пунктирной линии (находящейся до удаленной жесткой линии) до следующей жесткой линии раздела или конца документа. Жесткая линия раздела используется, как правило, для тех частей документа, которые в любом случае необходимо расположить на одной или нескольких идущих подряд страницах при выводе документа на принтер или в дисковый PRN-файл. В качестве иного приложения можно привести использование жесткой линии раздела для организации в PRN-файле или при выводе на печать страниц документа нестандартного размера (относительно установленной по команде 'Pagelength' длины страницы).

Если линия раздела страниц (установленная по команде пакета 'Pagelength') приходится на область выражения или графическую область любого типа (двухмерного или трехмерного графика, рисунка), то при выводе на печать такие конструкции будут разнесены по двум страницам. Чтобы этого не произошло, следует использовать команду 'Breakpages' пакета, по которой любая область выражения или графика (и только область этого типа) при условии попадания ее

на линию раздела страниц будет выводиться в начало новой, страницы документа, а конец предыдущей страницы будет заполняться пустыми строками. При этом, если размеры страницы не позволяют целиком разместить на одной странице область графика или выражения, то по команде 'Bbreakpages' перемещения такой области не производится, тогда как области, размером меньше страницы, по мере необходимости переносятся в начало следующей страницы.

Использование команды 'Bbreakpages' влияет только на перемещение областей выражений и графиков любых типов в начало следующей выводимой страницы, если они пересекаются линией раздела страниц, но не влияет на жесткую линию раздела и на текстовые области и зоны. После разметки документа командами 'Pagelength' и 'Pagebreak' ввод команды 'Bbreakpages' приводит к перемещению всех областей выражений и графиков, на которые приходятся линии раздела страниц, в начало следующих страниц так, что при выводе на печать они не разрываются на линии переноса страниц. Результат данной операции визуализируется на экране, позволяя отслеживать процесс форматизации документа. Однако даже после ввода команды 'Bbreakpages' положение вновь вводимых в документ областей выражений и графиков не корректируется относительно линий раздела страниц, поэтому команду 'Bbreakpages' рекомендуется вводить перед самым выводом на печать или в дисковый PRN-файл полностью готового документа.

В случае пересечения областей любого типа жесткой линией раздела перемещения их в формируемом документе перед выводом на печать или в PRN-файл следует производить самому пользователю, например, с помощью пакета 'Cut' и 'Paste'. В этом случае пакет отслеживает ситуации наложения областей и при необходимости производит их разделение с последующим переносом областей таким образом, чтобы они не попадали на линию раздела страниц, если их размеры не превышают размер страницы документа. Следующий фрагмент иллюстрирует использование команд 'Pagelength 5' и 'Pagebreak' для форматирования документа.



13
$$M := \begin{bmatrix} 42 & 47 \\ 67 & 89 \end{bmatrix}$$
 $G^2 + V^2 + S^2 + A^2 = 116$
14
15
16

17
18 $V = 19.42 \quad L = 19.64$
19

[K] Конец фрагмента документа

Согласно приведенному фрагменту документ содержит пять неполных страниц, форматирован как тремя линиями раздела страниц (по команде 'Pagelength 5'), так и одной жесткой линией (по команде 'Pagebreak'), проходящей по 12-й строке документа. На стыке первой и второй страниц находится определение (3×3)-матрицы. Для устранения разрыва при выводе ее на печать к документу применяется команда пакета 'Breakpages', результат выполнения которой иллюстрирует следующий фрагмент:

[H] $X := 1 \quad Y := 6 \quad Z := 26.8 \quad Z1(X) := \cos(X) + \ln(Y)$
1
2 $W := \sin(X)^2$
3
4
5
6 $M := \begin{bmatrix} 5.6 & 7.9 & -23 \\ 4.4 & -6 & 3.2 \\ 2.6 & 3.1 & 0.9 \end{bmatrix}$ $P := 1 \dots 45$
7
8

9
10 $D := (48 \quad 43 \quad 23 \quad 1.5)$

11 $A := \begin{bmatrix} 42 \\ 47 \end{bmatrix}$ Жесткая линия раздела
12
13 $M := \begin{bmatrix} 42 & 47 \\ 67 & 89 \end{bmatrix}$ $G^2 + V^2 + S^2 + A^2 = 116$
14
15
16

17
18 $V = 19.42 \quad L = 19.64$
19

[K] Конец фрагмента документа

Из данного примера видно, что команда 'Breakpages' область определения матрицы М перемещает в начало следующей страницы, тогда как область определения вектора А остается без изменения относительно жесткой линии раздела. Читателю, располагающему пакетом MathCAD, рекомендуется практически освоить работу по форматированию документов в его среде.

Из вышесказанного и прил. 1 следует, что пакет MathCAD поддерживает весьма широкий набор мозаичных и лазерных принтеров, на которые можно выводить текущий документ. Однако каждый тип принтера имеет и свои особенности (специальные символы, набор шрифтов, способы оформления текста и т. д.), которые было бы в ряде случаев желательно использовать. В этом направлении существуют два основных подхода: глобальный и локальный. При **глобальном** подходе достаточно написать собственный драйвер для принтера и включить его в состав пакета. Подробности проведения такой процедуры можно найти в документации [21]. **Локальный** подход позволяет использовать особенности принтера не ко всему выводимому документу, а к определенной его части. Для этих целей пакет предоставляет такие средства, использование которых достаточно просто, но требует более глубокого знания работы конкретного принтера. Осуществляется это с помощью так называемых ASCII-последовательностей, представляющих собой последовательности порядковых номеров символов в кодовой таблице ПК, начинающихся символом '\ ' (обратный слэш). В операционной системе MS DOS данные трехзначные номера совместно с клавишей 'Alt' позволяют вводить любые символы, допускаемые ПК, включая отсутствующие на клавиатуре.

Для включения в документ ASCII-последовательности необходимо в области или зоне текста ввести цепочку символов вида '\N₁\N₂\N₃... \N_n', где N_j (j = 1, n) представляет собой порядковый номер некоторого символа в кодовой таблице ПК. Данная цепочка передается принтеру при выводе документа на печать и управляет выводом или просто иницирует вывод определяемых ею символов, например, букв кириллицы или другого национального алфавита. Для использования в документе ASCII-последовательностей необходимо знать не только кодовую таблицу ПК, но и управляющие символы принтера, включенного в состав ПК, а также возможности последовательностей по управлению работой принтера. Приведем простой фрагмент документа, в котором используются обе возможности, предоставляемые ASCII-последовательностями:

[N] Использование ASCII-последовательности для ввода букв кириллицы на основе кодовой таблицы ПК (ИСКРА 1030)

```
\194\208\219\219\216\221\221 2\190\190\19038\254
\191\208\224\226\216\215\208\221\216 13—75
```

Использование ASCII-последовательности для управления шрифтом выводимой текстовой информации на (ИСКРА 1030)

[K] HS is a \015 Homogeneous structure

В результате вывода данного документа на печать по команде 'Print' пакета на принтере типа 'Epson' ПК (ИСКРА 1030) появится текст следующего содержания:

Использование ASCII-последовательности для ввода букв кириллицы на основе кодовой таблицы ПК (ИСКРА 1030)

Таллинн 200038
Партизаны 13—75

Использование ASCII-последовательности для управления шрифтом выводимой текстовой информации на (ИСКРА 1030)

HS is a Homogeneous structure

В первой части примера ASCII-последовательность использована в текстовой области документа для ввода русского текста, а во второй — для изменения шрифта печати, позволив два последние слова фразы вывести на печать «петитом». Для управления выводом на печать посредством ASCII-последовательностей рекомендуется подробнее ознакомиться с описанием конкретного принтера, на который будет осуществляться вывод. Для принтеров, совместимых с 'Epson', следующие порядковые номера представляют интерес для управления выводом:

- 010 — пропуск строки на принтере;
- 012 — пропуск страницы (32 строки принтера);
- 014 — вывод текста транспарантным шрифтом;
- 015 — вывод текста шрифтом «петит».

За подробностями по управляющим символам принтеров следует обращаться к соответствующей технической документации. Использование ASCII-последовательностей дает еще один способ для ввода в документ и вывода на печать символов национальных алфавитов, например кириллицы, хотя он и является достаточно громоздким.

При использовании в документе ASCII-последовательностей следует помнить, что в общем случае такой документ становится принтерозависимым и теряет свою универсальность. В документе, содержащем текстовую и графическую информацию, рекомендуется во избежание возможных недоразумений при выводе на печать не использовать ASCII-последовательности, главным образом, для изменения следующих характеристик печати: горизонтального шага печати, плотности выводимых символов и шрифта. Так как ASCII-последовательности могут быть достаточно сложными по запрашиваемым от принтера действиям, то для корректного их использования в документе следует внимательно ознакомиться с руководством по конкретному принтеру, имеющемуся в составе ПК (например, [32] для принтера NX-15, совместимого с серией 'Epson FX'). Для удобства и большей уверенности рекомендуется проводить отладку сложных и больших ASCII-последовательностей вне документа, для которого они предназначены.

Так как пакет MathCAD использует плоттеры несколько отличным от принтеров образом, то здесь уместно привести эти отличия. Прежде всего для использования плоттера в качестве устройства вывода необходимо по команде 'Selectprinter' выбрать его тип из имеющихся в файле MCAD.MCP драйверов пакета. Выводимый на плоттер график (как и для принтера) является точной копией экрана. Текстовая и графическая информация, а также выражения выводятся на плоттер в том же взаимном расположении, что и в самом документе.

При выводе на плоттер всего или части документа пакет (в отличие от принтера) уплотняет его на одну страницу плоттера, игнорируя при этом линии раздела страниц и жесткие линии раздела документа. При таком способе вывода документа весьма актуальным является корректное разбиение его на части, выводимые поочередно на плоттер. Как правило, в качестве выводимой части удобнее всего брать область документа, помещающуюся на экран. В любом случае

при выборе частей вывода размером больше экрана следует выбирать для них один размер, так как пакет будет пропорционально уплотнять их до страницы плоттера и в случае различий в размерах исходных частей могут возникнуть затруднения при компоновке единого выходного документа на основе таких частей. При выводе графиков различного типа на плоттер пакет использует для нанесения кривых до четырех цветов. Если же вывод в цвете невозможен, то он осуществляется единым черным цветом.

При выборе в качестве устройства вывода принтера по умолчанию пакет для вывода определяет весь документ, тогда как в случае плоттера — область вывода документа следующим образом: а) если курсор установлен в левом верхнем углу экрана, то выводится содержимое всего экрана; б) в противном случае — прямоугольная область документа, образованная верхним левым углом экрана и текущим положением курсора.

При использовании команды пакета 'Print' в ССП указывается именно эта область обозреваемого документа для вывода. Для подтверждения следует нажать клавишу 'Enter', а при необходимости изменения области вывода достаточно указать в ССП новые характеристики расположения ее в выводимом документе.

При выводе области документа, совместимой по размеру с экраном, рекомендуется помещать ее в окно экрана и устанавливать курсор в верхний левый угол экрана, а затем использовать команду пакета 'Print' для вывода по описанному способу (а). Если же область вывода документа меньше размера экрана, то ее рекомендуется помещать в верхний левый угол экрана, а курсор устанавливать на ее нижний правый угол, чтобы было возможно легко осуществлять вывод по команде 'Print' пакета согласно описанному способу (б). В обоих случаях выводимая область документа будет расширена на всю страницу плоттера. При выводе на плоттер следует иметь в виду, что текстовая информация выводится достаточно медленно и при уплотнении выводимой области до страницы плоттера возможно искажение такого типа информации. Наряду с этим следует помнить, что при сохранении вывода на плоттер в дисковом PRN-файле его впоследствии можно выводить на плоттер только по системной команде 'Print', а не по 'Сору' (как в случае принтера), так как только эта команда вывода операционной системы MS DOS имеет средства для квинтированного установления связи с плоттером. В любом случае при использовании плоттера для вывода документов рекомендуется предварительно ознакомиться с техническим руководством по нему для выяснения основных особенностей функционирования в составе данного ПК.

Глава 9

РАБОТА С ГРАФИКАМИ И РИСУНКАМИ В СРЕДЕ ПАКЕТА

Пакет MathCAD допускает в своей среде работу с плоскими (двухмерными) и поверхностными (трехмерными) графиками, а также с иллюстративными объектами, подготовленными в среде других

CAD-систем или систем программирования, допускающих создание и редактирование такого типа объектов. В дальнейшем такие иллюстративные объекты (схемы, рисунки, чертежи, диаграммы и т. д.) будем называть просто **рисунками**, понимая под ними любой иллюстративный материал, который не может быть подготовлен в среде пакета. В разделе 2.8 рассмотрены основы работы в среде пакета с плоскими графиками, а в настоящей главе обсуждаются более специальные вопросы по использованию таких графиков наряду с вопросами создания и использования в среде пакета поверхностных графиков и иллюстративных материалов, подготовленных другими программными средствами.

9.1. РАСШИРЕННОЕ ИСПОЛЬЗОВАНИЕ ПЛОСКИХ ГРАФИКОВ

Раздел 2.8 книги представляет основы использования плоских графиков в среде пакета MathCAD, однако вне рамок обсуждения остался ряд важных вопросов, которым и посвящен настоящий раздел. Прежде всего уместно напомнить и расширить основные моменты по созданию плоского графика функции от одной переменной. Для создания области плоского графика используется клавиша 'O' (эт), по которой по месту расположения курсора в документе создается макет области с шестью указателями (раздел 2.8; рис. 3). Для полного определения области необходимо задать вместо средних указателей по осям p -выражения от некоторой p -переменной или индексированные выражения. Вместо остальных четырех указателей, определяющих диапазоны изменения значений по осям координат, вводятся либо числа, либо скалярные выражения. При этом подобно функциям, которые предполагается представить в графическом виде, они могут быть как действительными, так и комплексными. Однако пакет при формировании графика и использовании диапазонов будет принимать во внимание только действительные части участвующих в вычислениях значений.

Если какой-либо из этих четырех указателей остается неопределенным, то значение для него автоматически вычисляется пакетом перед выводом графика в области, т. е. производится **автомасштабирование** области графика. При этом автомасштабирование может быть как полным, так и частичным, когда пользователь ряд значений для диапазонов определяет по своему усмотрению. При использовании автомасштабирования следует иметь в виду следующие два важных момента:

- 1) автомасштабирование области графика может потребовать более чем двукратного увеличения временных затрат на вывод графика по сравнению с явным заданием диапазонов;
- 2) предельные значения диапазонов осей автомасштабированного графика выводятся округленными и безразмерными; но при выводе графика размерных величин предельные значения диапазонов корректны в базовой системе измерений, принятой в текущем документе.

В связи с этими двумя моментами сделаем более детальные пояснения и рекомендации. Хронометрирование создания полностью авто-

масштабированного графика кривой показывает следующий фрагмент документа:

$$\begin{aligned} \{N\} \quad X &:= -5, -4.99 \dots 5 & Y(X) &:= e^X + X^4 + 4 & P &:= 0 \\ Y \left[\text{root} \left[\frac{d}{dP} Y(P), P \right] \right] &= 4.668 & Y(-5) &= 629.007 & Y(5) &= 777.413 \end{aligned}$$

Рис. 20

[К] Варианты графика одной и той же функции при различных значениях диапазонов на осях координат области

Для создания графика требуется 339 с, тогда как прямое задание диапазонов по осям области графика позволяет свести временные затраты по созданию графика к 176 с, т. е. налицо более чем двукратный выигрыш во времени. Однако второй момент является более существенным. Во всех трех областях фрагмента поведение графика относительно координат области одинаковое, хотя значения диапазонов совершенно разные.

В первой области осуществлялось автомасштабирование, тогда как во второй и третьей диапазон по оси Y задавался непосредственно, приближаясь к вычисленным значениям минимума и максимума функции на диапазоне оси X . Таким образом, во избежание возможных недоразумений, которые могут возникнуть во время следующего анализа полученного графика, рекомендуется, по возможности, задавать диапазоны по осям области непосредственно. Для этого может потребоваться некоторый предварительный анализ представляемой функции, подобный описанному.

Для полной или частичной отмены режима автомасштабирования достаточно необходимые предельные значения диапазонов по осям области графика заменить по усмотрению пользователя, тогда как обратная процедура сводится к удалению по клавише 'Del' или 'Bksp' необходимых предельных значений диапазонов (восстановление указателей значений). Переход от автомасштабирования области графика к прямому определению диапазонов по его осям и наоборот легко осуществляется путем обычной замены (частичной или полной) предельных значений диапазонов, т. е. допускается любой уровень масштабирования плоского графика, начиная от автомасштабирования, частичного автомасштабирования и кончая непосредственным определением всех предельных значений для диапазонов осей области графика. Подобная корректировка графической области полностью соответствует правилам редактирования выражений (раздел 2.7). Это же относится и к редактированию выражения, находящегося по месту любого указателя области плоского графика.

Параллельно с описанным пакет допускает и другой путь создания плоского графика, для чего достаточно установить курсор в нужное место документа и ввести конструкцию вида ' $Y \{(T)\} [T] @$ ', где $Y(T)$ представляет собой p -выражение от p -переменной T или некоторое индексированное выражение по индексу T , предназначенное для оси Y области графика. После ввода данной конструкции на

экране возникает область графика с указанным выражением вместо среднего указателя оси Y и с активным средним указателем оси X, вместо которого следует указать соответствующее X-выражение (ранжированное или индексированное). Для доопределения области графика можно воспользоваться любым из описанных методов, включая автомасштабирование. Разница в двух описанных методах создания плоских графиков несущественна, но первый имеет некоторые преимущества, когда в документе требуется зарезервировать конкретное место под график, отформатировать его область, но сам график не создавать. В ряде случаев подобного результата можно достичь путем выключения графической области из процесса вычислений посредством команды 'Equation' пакета. Таким образом, пакет предоставляет несколько путей создания плоского графика в текущем документе, позволяя выбрать наиболее приемлемый для каждого конкретного случая. Не дублируя сказанного и в целях сокращения изложения, а также большей наглядности проиллюстрируем наиболее часто используемые способы создания отдельных плоских графиков следующим фрагментом документа:

[H]	$Z := -5$	$\text{ORIGIN} := Z$	$t := Z \dots Z$
	$A_t := 0.48 \cdot t^3 - 0.43 \cdot t^2 + 0.23 \cdot t - 1.7$		$B_t := 19.42 \cdot \sqrt[3]{t} + 19.67 \cdot t + 4$
	График значений вектора A		График векторизованного выражения и вектора A

Рис. 21

$t := -4, -3.95 \dots 4$	$j := Z \dots Z$	$X(t) := 48 \cdot \sin(t)$	$B_j := 3 \cdot j$
	$Y(t) := 23 \cdot \cos(t)$	$A_j := 2 \cdot j + i$	
График параметрической функции		График двух векторизованных выражений	

Рис. 22

Наряду с приведенными выше примерами графиков функций от одного переменного данные примеры охватывают основные способы создания плоских графиков в среде пакета.

Помимо описанных основных способов создания плоских графиков пакет допускает достаточно произвольное графическое представление двух ранжированных или индексированных выражений по осям Y и X при условии, что каждая пара таких выражений рассматривается относительно одних и тех же p -переменной или индекса.

Наряду с отдельными пакет допускает создание в одной области нескольких плоских графиков при условии, что представляемые ими значения имеют одинаковые размерности соответственно по осям Y и X. С учетом сказанного для получения нескольких плоских графиков в одной области после ее создания следует вместо средних указателей по осям Y и X ввести соответственно **последовательно** (списки) ранжированных или индексированных выражений

$$Y_1, Y_2, \dots, Y_M \text{ и } X_1, X_2, \dots, X_N$$

каждая пара (Y_j, X_j) которых определяет отдельный плоский график со своими (возможно различными) диапазонами по осям Y и X . Если графики имеют различные диапазоны изменения по осям области, то при автомасштабировании устанавливаются по каждой из осей Y и X диапазоны, включающие диапазоны всех выводимых графиков. Тогда как непосредственно диапазоны по осям области пользователь имеет возможность определять по своему усмотрению. Создавать несколько плоских графиков в одной графической области можно и по описанной для случая отдельного графика конструкции « $Y_1, Y_2, \dots, Y_M \text{ @}$ » со всеми очевидными изменениями. В указанных списках выражений при определении плоских графиков не обязательно требуется выполнение условия $M = N$. В зависимости от значений M и N возможны следующие три ситуации:

1) $M = N$ (основной случай); выводятся M плоских графиков в соответствии с числом пар выражений (Y_j, X_j) ($j = \overline{1, M}$);

2) $M < N$; выводятся M плоских графиков согласно числу пар выражений (Y_j, X_j) ($j = \overline{1, M}$); остальные выражения X_k ($k = \overline{M+1, N}$) игнорируются;

3) $M > N$; выводятся N плоских графиков согласно числу пар выражений (Y_j, X_j) ($j = \overline{1, N}$) и $(M - N)$ пар плоских графиков в соответствии с парами выражений (Y_k, X_N) ($k = \overline{N+1, M}$).

Отсюда, в частности, следует, что если для оси X последние выражения списка идентичны, то вместо них достаточно указать только одно выражение. Наряду с этим имеется большая свобода при описании графиков.

Описанная возможность создания нескольких плоских графиков в одной области может быть использована для визуализации линий масштабирования по осям координат, создавая таким образом в графической области **масштабную сетку**. Это позволяет существенно упрощать визуальный анализ находящихся в области графиков. Для создания координатных линий $X = A$ и $Y = B$ достаточно в списке выражений соответственно по осям Y и X ввести дополнительно пары (V, A) и (B, W) , где V, W — произвольные p -переменные соответственно с диапазонами изменения осей Y и X . Следующий фрагмент документа иллюстрирует основные способы создания нескольких плоских графиков и координатных линий:

[H] $J := -3, -2.9 \dots 3$ $X(j) := 10 \cdot \sin(j)$ $Z(j) := -2^j$
 $Y(j) := j^2$ $k := -10 \dots 10$ $d := -10$ $ORIGIN := d$

Три графика функций

Визуализация декартовой (YOX) -системы координат

Рис. 23

$I := -3, -2.8 \dots 3$
 Задание функций и системы координат полными списками

$Y(i) := i^2$ $t := d \dots -d$ $B_t := t$
 $A_t := 3 \cdot t + 5i$

Вывод графиков векторизованных выражений и координат

Рис. 24

$$X := -2, -1.9 \dots 2$$

$$Y(X) := X^4$$

Рис. 25

¶К] Вывод двух графиков функций и пяти координатных X-линий

Наряду с описанными пакет допускает и другие способы вывода нескольких графиков в одной области, часть из которых в качестве примеров будет рассмотрена ниже. Однако для улучшения визуализации графической области рекомендуется выводить графики, диапазоны изменения которых по осям (особенно оси Y) близки друг к другу, так как в противном случае соотношение масштабов не позволит удовлетворительно анализировать взаимное расположение кривых. При использовании в графической области масштабной сетки рекомендуется в списках выражений по осям области определять первыми, по возможности, именно пары выражений для координатных линий, а затем пары выражений для самих графиков. Это позволяет в ряде случаев получать большую свободу при описаниях выводимых графиков.

При выводе графика функции, имеющей особенности (например, сингулярность) в точках диапазона оси X, в качестве функции (на оси Y) можно указывать условие, устраняющее особенности, препятствующие или затрудняющие вывод графика. Например, если требуется вывести график функции вида

$$Y(X) := \cos(X); Z(X) := X; F(X) := \frac{Y(X)}{Z(X)},$$

имеющей в точке $X = 0$ сингулярность, то в качестве функции вывода можно указать функцию $U(X) := \text{if } (Z(X) \approx 0, \infty, F(X))$. Функция $U(X)$ на всей оси X совпадает с функцией $F(X)$, а в точке $X = 0$ полагается равной бесконечности. При этом, если значение переменной (∞) пакета неудовлетворительно, то ее в документе можно разумным образом переопределить. Этот же подход можно положить в основу другого способа устранения сингулярности функций при выводе их графиков. Варьирование значения переменной (∞) пакета в некоторых случаях позволяет также удовлетворительно решать вопрос масштабирования выводимых графиков в документе.

Подобно случаю выражений область плоского графика можно форматировать как на глобальном, так и на локальном уровнях. Посредством форматирования можно изменять размеры самой области, типы представления и соединения выводимых точек графика, устанавливать логарифмическую шкалу и вводить координатные сетки линейного или логарифмического типа. Для форматирования на локальном уровне курсор устанавливается в нужную область графика и нажимается клавиша {f | F}, по которой в правой части экрана выводится меню локального формата плоского графика (ЛГФ), содержащее следующие характеристики с их значениями по умолчанию и действия:

1. Size = 5, 15
2. Trace Types = L
3. Log Cycles = 0, 0
4. Subdivisions = 1, 1
5. Global Default
6. Revert
7. Done

Действия 5—7 формата полностью аналогичны по назначению и использованию аналогичным из ЛФР (раздел 2.2) и особых пояснений не требуют. Характеристика 1 определяет размер ($N \times M$) области графика в строках (N) и столбцах (M) документа. Значения ее параметров N , M должны быть положительными целыми числами не большими 127, т. е. максимальный размер области плоского графика ориентирован на оптимальное использование 128-колонного принтера, являющегося наиболее массовым.

Таблица 15

Символ	Смысловая нагрузка
{L l}	Соединение точек графика отрезками прямой
{D d}	Вывод только самих точек плоского графика
{S s}	Соединение точек графика вертикальными и горизонтальными линиями (ступенчатый контур)
{B b}	Вывод ступенчатого контура подобно случаю «S» и штриховка области под контуром до оси X
{x p o v}	В каждой точке графика помещается символ {X + квадратик ромбик (оба с центрами в точке графика)}; точки графика не соединяются
{X P O V}	То же, но точки графика соединяются отрезками прямых линий
{E e}	Заштриховывается прямыми вертикальными линиями (в каждой точке X-диапазона) область, ограниченная графиками первой и второй функций (порядок функций определяется их местом в Y-списке выражений графической области)

Характеристика 2 определяет вид выводимого графика. В качестве ее значения используется цепочка символов, назначение которых определяет табл. 15. Характеристика 2 действует на все графики области и позволяет существенно разнообразить визуализацию графической информации, а также вывести специального типа графики, например, гистограммы и диаграммы. Для одинарного графика использование цепочки из букв (табл. 15) в качестве значения характеристики 2 воспринимается, но имеет смысл только первая буква цепочки. В случае же нескольких графиков в области каждому из них ставится в соответствие своя буква цепочки (j-я в значении характеристики j-му в Y-списке выражений) и соответственно этому происходит оформление графиков в области.

Характеристика 3 (Log Cycles = Y, X) своими параметрами определяет типы соответственно осей Y и X графической области. Значениями этих параметров могут быть целые числа в интервале 0—63, определяющие типы осей координат следующим образом:

0 — обычная декартова (линейная) ось координат;

1 — логарифмическая ось координат.

При значении $N > 1$ параметра логарифмическая ось делится на N циклов, которые равномерно размечают ее прямыми линиями. Область графика может иметь: обе оси линейными (декартова система координат), одну из осей логарифмической (полулогарифмическая система координат), либо обе оси логарифмическими (логарифмическая система координат). Для логарифмической оси значения ее диапазона должны быть больше нуля, иначе идентифицируется ошибочная ситуация с диагностикой 'Must be positive'. Переход к логарифмическим осям и наоборот требует перевычисления в соответствующей графической области, о чем пакет выдает соответствующее сообщение.

Характеристика 4 (Subdivisions = Y, X) служит для определения линейной или логарифмической сетки соответственно по осям Y и X. Значениями параметров характеристики 4 являются целые числа из интервала 0—63, интерпретируемые пакетом следующим образом:

Линейная ось — значение N представляет число равномерно распределенных координатных линий ($N = 1$ — линии совпадают с самой осью; $N > 1$ — разбиение оси прямыми линиями на N одинаковых отрезков);

Логарифмическая ось — значение N (имеют смысл только значения $N = 1, 2, 9$) определяет число подразделений каждого цикла так, что при $N = 1$ подразделение отсутствует; при $N = 2, 9$ цикл логарифмически подразделяется соответственно на 2 и 9 частей.

Характеристики 3 и 4 позволяют в достаточно широких диапазонах варьировать систему координат графической области, что особенно важно при решении различных научно-технических вычислительных задач. Следующий фрагмент документа иллюстрирует использование локального формата для форматирования графической области.

```
[H] X := -2, -1.9..2      Y (X) := X^2      k := 0..100
    d := 10              Trace Types=e      p := 0..d
```

Рис. 26

```
Tk := rnd 100          Intp := p · d      t := 0..d - 1
Trace Types=B          Trace Types=E
```

Рис. 27

```
Trace Types=Lod          Subdivisions=10, 10
```

Рис. 28

```
Z := 1, 1.1..3          Trace Types=LD
Subdivisions=5, 10      Log Cycles=1, 1
                        Subdivisions=2, 2
```

Рис. 29

Trace Types=LD
Log Cycles=2,0
Subdivisions=2, 10

Z := 2, 2.1 .. 5

Trace Types=LD
Log Cycles=2,2
Subdivisions=2,2

Рис. 30

Size=15,25
Trace Types=LD
Log Cycles=1,1
Subdivisions=9,9

Size=15,25
Trace Types=L
Log Cycles=2,2
Subdivisions=9,2

Рис. 31

С учетом сказанного и комментариев фрагмента приведенные в нем примеры [K] особых пояснений не требуют.

На глобальном уровне форматизацию областей плоских графиков позволяет осуществлять команда пакета

PLOTFORMAT | — | —

по которой в правой части экрана появляется меню **глобального формата плоского графика (ГГФ)**, с очевидными отличиями совпадающее по содержанию и смыслу с ЛГФ. После определения нового ГГФ все последующие области плоских графиков документа форматировются согласно ему. Согласно новому формату можно форматировать и существующие графические области, задав в формате действие 'Global Default'. Наконец, по команде пакета 'Configsave' можно сохранить новый ГГФ в файле конфигурации MCAD.MCC пакета (обновление значений параметров у команды 'Plotformat' файла). В последнем случае все последующие загрузки пакета будут использовать именно этот формат плоских графиков по умолчанию как на глобальном, так и на локальном уровнях.

9.2. ИСПОЛЬЗОВАНИЕ ГРАФИКОВ ФУНКЦИЙ ОТ ДВУХ ПЕРЕМЕННЫХ

Наряду с двумерными графиками функций одной переменной, базирующимися на использовании p -переменных, векторов и выходных таблиц, рассмотренных в гл. 2, пакет MathCAD поддерживает работу с трехмерными (**поверхностными**) графиками функций от двух переменных, в основе представления которых лежат двумерные массивы (прямоугольные матрицы). Поверхностный график представляет собой изображение функции от двух действительных или комплексных аргументов. Подобно случаю других типов объектов, с которыми оперирует пакет, для поверхностных графиков также в текущем документе создается область **поверхностного графика (ОПГ)**. Для создания ОПГ курсор устанавливается в нужное место текущего документа и нажимаются клавиши 'Alt + @ (эт)', в результате чего на экране появляется макет ОПГ с одним активным указателем (рис. 32). К ОПГ в полной мере применимы команды копирования 'Cut', 'Copy' и 'Paste' пакета.

Для завершения формирования ОПГ и визуализации графика искомой поверхности вместо активного указателя необходимо определить имя так называемой **функциональной матрицы** или матричного выражения, содержащего эту матрицу (результатом которого также должна быть матрица). Визуализация искомого графика поверхности происходит после нажатия функциональной клавиши 'F9' (в М-режиме пакета) или после вывода курсора из сформированной ОПГ (в А-режиме). Визуализируемая функциональная $(N \times M)$ -матрица Z представляет собой значения некоторой функции $Z = F(X, Y)$ в узлах прямоугольной (X, Y) -сетки.

Следовательно, перед выводом графика поверхности, определяемой функцией $Z = F(X, Y)$, необходимо сформировать функциональную $(N \times M)$ -матрицу Z ее значений в точках (X_i, Y_j) пря-

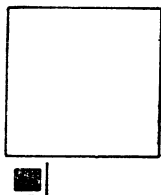


Рис. 32. Макет области поверхностного графика в документе

моугольной сетки ($i = 0, N - 1; j = 0, M - 1$). С этой целью определяются p -переменные $\{i, j\}$, задающие число точек (X, Y) -сетки, и сами точки сетки как некоторые функции $\{X_i, Y_j\}$ этих p -переменных. Наконец, элементы $Z_{i,j}$ функциональной матрицы определяются как функция $F(X_i, Y_j)$ от двух переменных. Для завершения определения ОПГ достаточно вместо активного указателя ввести либо имя функциональной матрицы, либо матричное выражение, которое содержит данную матрицу и результат которого является прямоугольной матрицей. Функциональная матрица Z может быть как действительной, так и комплексной. В комплексном случае

пакет строит поверхность для матрицы $\text{Re}(Z)$, а в чисто мнимом случае для матрицы $\text{Re}(Z - \text{Re}(Z))$, т. е. нулевой функциональной матрицы. Следующий фрагмент документа иллюстрирует построение поверхностного графика для действительных и комплексных функций от двух переменных:

```
[H]  p := -10      ORIGIN := p      H := 0.2      i := p .. -p
      j := p .. -p      Xi := i · H      Yj := j · H
      Zi,j = Xi8 · ej - Yj8 · ei      A := 19.42 - 48i      B := -5.47 + 3.67i
```

Рис. 33

```
X := 0 .. 5      Y := 0 .. 3      ZX,Y :=  $\frac{X^2}{A} - \frac{Y^2}{B}$       M := Re(Z)
```

Рис. 34

Рис. 35

Последние графики представляют собой попарно одинаковые результаты [K] вывода соответственно комплексных и чисто мнимых функциональных матриц.

Из приведенного фрагмента следует, что в отличие от плоских поверхностный график визуально не привязан непосредственно к системе координат и, по сути дела, представляет собой набросок вида поверхности, по которому можно делать определенные выводы о ее качественном поведении. Степень достоверности такого анализа полностью определяется выбранной окрестностью точки (X, Y) , в которой исследуется функция $Z = F(X, Y)$. При этом, функциональная матрица представляет однозначные значения соответствующей функции $Z = F(X, Y)$ в каждой точке (X, Y) -сетки, поэтому поверхностный график представляет только одну из частей функции симметричных относительно плоскости XOY , например, верхнюю, полусферу шара. При необходимости вывода поверхностного графика функции, имеющей в области ее определения особенности, можно использовать подходы, описанные в предыдущем разделе для плоских графиков.

Учитывая, что при создании поверхностного графика пакет для соединения точек функциональной матрицы использует прямые линии, для большей наглядности и удобства качественного анализа рекомендуется создавать графики в небольших окрестностях при достаточно частой (X, Y) -сетке. Однако в таких случаях временные издержки оказываются для низших классов ПК весьма существенными. Данное замечание относится и к выводу поверхностных графиков на печать. На само же качество изображения поверхностного графика существенное влияние оказывает тип монитора, используемого на ПК.

В соответствии с описанным принципом определения ОПГ выводимый график поверхности рассматривается в декартовой системе координат (X, Y, Z) таким образом, чтобы положительная ось X была направлена в сторону от наблюдателя, положительная ось Y — вправо и положительная ось Z — вверх от левого нижнего угла ОПГ. При транспозиции функциональной матрицы оси X и Y меняются местами. В процессе создания и вывода графика поверхности пакет соединяет прямыми линиями выводимые им точки, создавая сеточный образ искомой поверхности. Вид такого изображения зависит от выбора точки обзора поверхности. Для этого используется форматирование поверхностного графика, базирующееся на глобальном и локальном поверхностных форматах. При создании поверхностного графика пакет по умолчанию использует для его вывода глобальный поверхностный формат (ГПФ). Пакет позволяет изменять поверхностный формат как на глобальном, так и на локальном уровнях. На глобальном уровне соответствующим изменениям подвергаются все (как существующие, так и вновь создаваемые) поверхностные графики текущего документа, тогда как на локальном уровне — только график, в области которого был предварительно установлен курсор.

Для изменения ГПФ служит команда пакета

SURFACEFORMAT | — | —

по которой в правой части экрана высвечивается меню формата и

предоставляется возможность изменения значения его характеристик и/или выполнения его действия. Состав меню ГПФ и значения по умолчанию его характеристик полностью соответствуют меню ЛПФ, работа с которым рассматривается ниже. Определив новый ГПФ, пользователь получает возможность создавать согласно ему поверхностные графики в новых ракурсах и/или сохранять новый ГПФ в файле MCAD.MCC конфигурации пакета (обновление значений параметров команды SURFACEFORMAT файла) по команде 'Config-save'. Последняя возможность позволяет использовать новый ГПФ по умолчанию при всех последующих загрузках пакета.

Форматирование дает возможность управлять многими визуальными характеристиками поверхностного графика, иллюстрацию которого рассмотрим на примере наиболее часто используемого ЛПФ, определяемого подобно другим рассмотренным локальным форматам пакета. Для использования ЛПФ курсор устанавливается в нужную ОПГ и нажимается клавиша {f | F}, вызывая появление в правой части экрана меню ЛПФ, содержащего следующие характеристики (с их значениями по умолчанию) и действия:

1. Size = 15, 30
2. Rotation = 10
3. Tilt = 35
4. Hidden Lines = n
5. Vertical Scale = 20
6. Global Default
7. Revert
8. Done

Действия 6—8 формата полностью аналогичны действиям из ЛФР и ЛГФ (разделы 2.2, 2.8, 9.1) и особых пояснений не требуют. Характеристика 1 определяет размер ОПГ соответственно в строках и позициях документа. Значения параметров характеристики 1 должны быть положительными целыми числами, не превышающими 127, т. е. размер ОПГ ориентирован на максимальное использование 128-колонок принтера при выводе поверхностного графика на печать.

Характеристика 2 позволяет обозревать график вращением его вокруг оси Z по часовой стрелке. Угол вращения задается в градусах и изменяется от 0° до 90°. При нулевом значении график обозревается прямо вдоль первого столбца функциональной матрицы, первая строка которой направлена вправо. В результате поворота рисунка на 90° он обозревается прямо вдоль первой строки функциональной матрицы, первый столбец которой направлен влево. Значения характеристики 2, расположенные между 0° и 90°, позволяют обозревать график исследуемой функции с позиций, расположенных между описанными.

Характеристика 3 позволяет перемещать точку обзора графика к зениту. Значение характеристики лежит в пределах от 0° до 90° и изменение его в этих пределах определяет точку обзора графика, начиная с плоскости XOY до точки зенита, когда график поверхности

обозревается точно сверху, т. е. вдоль оси Z декартовой системы координат.

Характеристика 4 определяет видимость (п) или невидимость (у) линий, скрытых другими частями поверхности. Значение 'у' характеристики используется для прояснения или разрежения сети графика, когда выводятся только все видимые его линии. Значение 'п' характеристики используется для уплотнения линий графика, в котором выводятся также части, скрытые другими частями поверхностного графика. Иными словами, для характеристики 4 значение 'п' определяет вывод как видимых, так и невидимых (скрытых другими частями поверхности) линий, а значение 'у' определяет вывод только видимых линий поверхностного графика. Следует отметить, что значение 'у' характеристики 4 приводит к более медленному выводу графика поверхности относительно значения 'п', так как пакет должен анализировать какие линии поверхности будут видимыми, а какие нет. При этом, чем большей является функциональная матрица, тем более значительным будет эффект замедления. Значительные временные затраты имеют место также при выводе поверхностного графика на печать.

Наконец, характеристика 5 определяет растяжение графика поверхности по вертикали. Значением характеристики может быть любое целое положительное число от 0 до 100. При нулевом значении характеристики перепад высот поверхности будет, практически, неразличим (а в ряде случаев исчезнет совсем), а при значении 100 изменения будут так велики, что может оказаться невидимой верхняя часть поверхности. Следующий фрагмент документа иллюстрирует использование ЛПФ:

{H} p:=-8 ORIGIN:=p H:=0.005 i:=p ..-p j:=p ..-p X_i:=i·H
Y_j:=j·H Z_{i,j}:=X_i⁸·eⁱ-Y_j⁸·eⁱ

Использован ЛПФ по умолчанию Rotation=55 Tilt=30
(после загрузки пакета) Vertical Scale=35

Рис. 36 (1)

Рис. 36 (2)

Rotation=35 Tilt=45
Vertical Scale=40

Rotation=0 Tilt=90
Vertical Scale=100

Рис. 37 (1)

Рис. 37 (2)

p:=-5 ORIGIN:=p H:=0.25 i:=p ..-p j:=p ..-p
X_i:=i·H Y_j:=i·H

$$Z_{i,j}:=0.5-X_i^2-Y_j^2$$

Hidden Lines = n

Hidden Lines = Y

Рис. 38 (1)

Рис. 38 (2)

В приведенных графиках указаны только характеристики ЛПФ, значения {K} которых отличаются от ЛПФ по умолчанию.

Из приведенного примера хорошо видно как средство форматизации позволяет получить вполне удовлетворительное представление о поведении исследуемой функции от двух переменных в заданной окрестности начала координат. Наряду с этим пример иллюстрирует также возможность завершения формирования функциональной матрицы исследуемой функции посредством матричного выражения в самой ОПГ, что в целом ряде случаев позволяет существенно упростить определение в документе функции от двух переменных. Область поверхностного графика может выключаться из процесса вычислений в документе посредством команды пакета 'Equation', как это имеет место для любой вычислительной области, исключая текстовые области и зоны.

9.3. ИСПОЛЬЗОВАНИЕ В СРЕДЕ ПАКЕТА ИЗОБРАЗИТЕЛЬНЫХ ФРАГМЕНТОВ ИЗ ДРУГИХ CAD-СИСТЕМ И СИСТЕМ ПРОГРАММИРОВАНИЯ

Средства пакета не позволяют создавать и редактировать непосредственно в его среде изобразительные объекты (рисунки, наброски и т. д.). Поэтому наряду с поддержкой работы с плоскими и поверхностными графиками пакет позволяет использовать в документе **иллюстративные** (рисуночные) фрагменты, предварительно созданные в других CAD-системах (AutoCAD, TurboCAD, AutoSketch, MicroCADAM, CADkey, VersaCAD и др.). В настоящее время такие известные объектно-ориентированные системы программирования, как Object-Oriented Pascal, C++, Smalltalk; Objective-C, Eiffel, Actor и CLOS содержат средства интерфейса с CAD/CAM-системами, что позволяет в их среде также готовить изобразительные объекты различного назначения. Информацию об этих средствах можно найти в превосходной серии из пяти новых журналов [38] по объектно-ориентированному программированию, а также в книге [44].

Используя упомянутые программные средства, следует подготовить дисковый HPGL-файл, содержащий требуемый рисуночный объект и готовый к выводу на плоттер. Такие выходные файлы называются *HPGL-файлами*, получившими свое название от аббревиатуры Hewlett-Packard Graphics Language — универсального языка для работы с изобразительными объектами. Для включения в текущий документ такого HPGL-файла предназначена утилита MCSTRANS, входящая в состав пакета MathCAD и служащая для перевода файла HPGL-формата в файл MCAD-формата. После применения данной утилиты к исходному HPGL-файлу получаем MCAD-файл, готовый к включению в текущий документ по указанному в нем месту.

Для указания местоположения рисуночного объекта в документе служит **область рисунка (ОРИ)**, создаваемая по клавишам «Alt + + %» подобно рассмотренным областям других типов (плоского и поверхностного графиков). Макет ОРИ содержит ограничивающий ее прямоугольник и один активный указатель, вместо которого следует определять имя MCAD-файла, содержащего искомый рисунок, или ассоциированную с ним переменную. Другой способ идентифи-

кации MCAD-файла применяется, если он находится в каталоге, отличном от активного. После завершения определения ОРИ и вывода из нее курсора (А-режим пакета) или нажатия функциональной клавиши 'F9' (М-режим пакета) пакет помещает в определенную область требуемый рисунок. Такова общая схема включения в текущий документ иллюстративных объектов, подготовленных вне среды самого пакета. Данная возможность позволяет существенно расширять изобразительные средства в среде пакета и создает интерфейс с известными CAD-системами и другими подобными программными средствами. Ниже данный вопрос рассматривается более подробно.

Перед погружением требуемого рисунка в среду пакета его необходимо предварительно подготовить в среде некоторой CAD-системы или другого подобного средства и создать на его основе дисковый HPGL-файл, готовый к непосредственному выводу на плоттер. В качестве такой CAD-системы рекомендуется использовать хорошо известные пакеты AutoCAD или AutoSketch фирмы Autodesk Inc., которые в значительной степени являются эталонными для данного типа программных средств. С подробным описанием системы AutoCAD с ориентацией на непрофессионального пользователя можно ознакомиться в книге [28]. В книге представлен широкий круг вопросов, начиная с загрузки системы AutoCAD, работы с дисплеем, вычерчивания и редактирования объектов, до вывода созданных объектов на принтер или плоттер, а также повышения эффективности системы. Большое количество представленных в книге интересных примеров позволяет проиллюстрировать основные возможности системы AutoCAD. Хорошее описание пакета AutoSketch можно найти в фирменном руководстве [37], а также в книге [44].

Для перевода созданного вне среды пакета MathCAD дискового HPGL-файла в MCAD-формат следует использовать утилиту MCSTRANS.EXE, вызов которой в среде MS DOS на выполнение имеет вид

[<УВВ:>] MCSTRANS <СФ1> [<СФ2>]

где параметры 'СФ1' и 'СФ2' определяют соответственно спецификаторы входного HPGL-файла и выходного MCAD-файла, содержащего необходимый рисунок в MCAD-формате. При отсутствии второго параметра утилита пакета создает выходной MCAD-файл с тем же основным именем, что и у входного HPGL-файла (но с расширением имени '.MCS') и на том же томе, где и входной файл. Для входного файла по умолчанию полагается расширение имени '.PLT'. Вызов утилиты можно осуществлять и по команде пакета

DOS [<УВВ:>] MCSTRANS (Параметры)

после выполнения которой по нажатию любой клавиши осуществляется возврат в среду пакета в состояние текущего документа.

После или до создания ОРИ командой пакета

FILENAME <Имя переменной> <СФ>

с MCAD-файлом, определяемым параметром 'СФ' и содержащим ис

комый рисунок, ассоциируется переменная, имя которой определяет первый параметр команды. При определении параметра 'СФ' следует обязательно указывать расширение имени файла, если оно отлично от '.MCS'. Как и в случае файлов данных MCAD-файлы обязательно должны иметь расширение имени.

Для включения рисунка в текущий документ следует в любом его свободном месте создать ОРИ посредством клавиш 'Alt + %', в результате чего на экране возникает макет области с одним активным указателем, аналогичный макету ОПГ (см. рис. 32). Для завершения определения ОРИ вместо активного указателя необходимо ввести имя MCAD-файла или имя ассоциированной с ним переменной. В результате этого пакет отыщет файл согласно его имени (в активном каталоге) или ассоциированной с ним переменной и после успешного поиска поместит содержащийся в нем рисунок в связанную с ним ОРИ. Документ может включать несколько ОРИ, число которых ограничивается размером как доступной памяти ПК, так и самих рисунков. При этом, если MCAD-файл содержит рисунок с более чем 8000 точками, то пакет помещает в ОРИ только часть этого рисунка, определяемую первыми 8000 точками. Рисунки, содержащие большое число точек, требуют значительного размера оперативную память ПК. Поэтому при необходимости включения в документ больших рисунков на основе исходного рисунка рекомендуется создать несколько рисунков меньшего размера и соответственно им несколько дисковых HPGL-файлов, а затем с помощью утилиты пакета MCSTRANS создать несколько MCAD-файлов для непосредственного использования их в текущем документе.

Подобно случаю графических областей для ОРИ пакет предоставляет возможность форматирования содержащихся в них объектов как на глобальном, так и на локальном уровнях. К ОРИ в полной мере применимы команды копирования 'Cut', 'Copy' и 'Paste' пакета. Для изменения формата рисунка на глобальном уровне (в рамках всего текущего документа) служит команда пакета

SKETCHFORMAT | — | —

результатом выполнения которой является вывод на экран меню **глобального формата рисунка** (ГФРИ), идентифицирующего характеристики формата (с их значениями по умолчанию) и его действия, и предоставляющего возможность их изменения. После изменения ГФРИ пакет позволяет создавать в документе все последующие рисунки с этим форматом, изменять формат всех существующих рисунков документа согласно новым значениям его характеристик либо сохранять новый ГФРИ в файле MCAD.MCS (обновление значений параметров команды SKETCHFORMAT файла) пакета посредством команды 'Configsave'. Состав характеристик и действий ГФРИ с очевидными изменениями аналогичен составу **локального формата рисунка** (ЛФРИ), который, в свою очередь, задается аналогично ЛФР, ЛГФ и ЛПФ (по клавише {f | F}).

Для определения ЛФРИ курсор помещается в нужную ОРИ и нажимается клавиша {f | F}. В результате в правой части экрана вы-

свечивается меню ЛФРИ, содержащее следующие его характеристики (с их значениями по умолчанию) и действия:

1. Size = 15, 30
2. Magnification = 1, 1
3. Center = 0.5, 0.5
4. Box = y
5. Global Default
6. Revert
7. Done

Действия 5—7 полностью аналогичны действиям из ЛФР, ЛГФ и ЛПФ (разделы 2.2, 2.8 и 9.2) и особых пояснений не требуют. Характеристика 1 определяет размер ($N \times M$) ОРИ в терминах строк (N) и столбцов (M) документа. Значения для параметров данной характеристики должны быть положительными целыми и не превышать 127. Максимальный размер (127×127) ОРИ ориентирован на максимальное использование возможностей 128-колонного принтера, являющегося наиболее массовым.

Характеристика 2 определяет размер самого рисунка относительно исходного его размера после загрузки в документ, не затрагивая размера самой ОРИ. Значениями параметров данной характеристики являются положительные действительные числа, которые определяют соответственно изменения пропорций рисунка по вертикали и горизонтали. При этом значения (1, 1) определяют тот размер рисунка, с которым он был загружен в документ из MCAD-файла, а значения, меньшие или большие единицы, задают соответственно коэффициенты сжатия или растяжения рисунка по вертикали и горизонтали. Использование нулевых значений особого смысла не имеет, так как рисунок превращается соответственно в горизонтальную или вертикальную прямую линию. Оба нулевых значения превращают рисунок в точку, расположенную в центре ОРИ, определяемом значениями параметров характеристики 3 формата. Например, значения (1.5, 2) параметров характеристики 2 определяют растяжение рисунка по вертикали в 1.5 и по горизонтали в 2 раза относительно его начальных размеров (размеров рисунка после загрузки в документ). Следует иметь в виду, что увеличение размеров рисунка может привести к тому, что обозреться будет только некоторая его часть, остающаяся в ОРИ.

Характеристика 3 служит для центровки рисунка в ОРИ. Значения ее параметров по умолчанию (0.5, 0.5) располагают рисунок по центру ОРИ, тогда как другие значения (Y, X) параметров определяют центр рисунка с координатами $[N(1 - Y), M(1 - X)]$, где N и M — соответственно размеры ОРИ по вертикали и горизонтали. Вычисления нового центра производятся относительно левого верхнего угла ОРИ. Например, значения (0.25, 0.25) параметров характеристики 3 определяют центр рисунка в конце третьей четверти первой диагонали ОРИ.

Наконец, характеристика 4 определяет видимость ограничивающего ОРИ прямоугольника (y — видим, n — невидим) и служит, в основном, для организации лучшего иллюстративного оформления

документа, содержащего рисунки, при выводе его на печать. Рассмотрим теперь некоторые вопросы совместимости при использовании в среде пакета иллюстративного материала, созданного в среде других программных средств.

При использовании программных средств, позволяющих создавать и редактировать иллюстративные объекты (которые предполагается погружать в среду пакета MathCAD), они должны удовлетворять следующим трем основным условиям:

1) допускать вывод на Hewlett-Packard — совместимые плоттеры;

2) допускать переадресацию вывода, предназначенного для плоттера, в выходной дисковый HPGL-файл;

3) вывод на плоттер должен содержать только следующие команды: PA, PD, PR, PU, LB и SP.

Наличие первых двух условий проверяется простым просмотром соответствующих руководств используемого программного средства. Многие типы CAD-систем допускают вывод на плоттеры и большинство из них позволяют помещать в выходной дисковый HPGL-файл выход, предназначенный для плоттера.

Третье условие проверить сложнее, так как сам HPGL-файл содержит много команд. Отмеченные выше шесть команд плоттера являются наиболее общими. Например, известный пакет AutoCAD использует только эти шесть команд плоттера для организации вывода. Если некоторая CAD-система использует дополнительные команды для вывода на плоттер, то утилита MCSTRANS пакета будет игнорировать эти команды при переводе HPGL-файла в соответствующий ему MCAD-файл. Нераспознанные утилитой команды могут как влиять, так и не влиять на визуализацию погруженного в документ рисунка в зависимости от их связи с выводом на плоттер.

Для определения совместимости используемого для создания иллюстративного материала программного средства с пакетом MathCAD рекомендуется создать с его помощью некоторый контрольный рисунок и по описанной технологии попытаться использовать его в среде пакета. Если загруженный в документ рисунок корректно визуализируется, то данное программное средство, по всей вероятности, совместимо с пакетом, в противном случае следует выяснить причину некорректности либо использовать другое подходящее программное средство. При этом, если рисунок содержит текст, созданный на основе встроенного font-файла плоттера, то при погружении рисунка в среду пакета такой текст исчезает. Однако, например, использование пакета AutoCAD позволяет созданные в его среде рисунки, включая текстовую информацию, безболезненно включать в среду пакета MathCAD. В качестве иллюстрации сказанного приведем фрагмент документа, в котором демонстрируется рисунок, подготовленный в среде пакета AutoCAD:

[H] Перед созданием ОРИ и загрузкой в нее рисунка вводятся следующие команды пакета:

Filename Pic C:Picture.MCS

DOS Mcstrans B:PTIP.PLT C:Picture

[K] Формат по умолчанию Center = 0.5, 0.25 Box = n

Вначале пакет командой 'Manual' переводится в М-режим. В данном примере HPGL-файл PTIP.PLT находится в главном каталоге дискового тома на УВВ 'В'. Утилитой MCSTRANS он переводится в MCAD-файл и помещается в главный каталог тома на УВВ 'С'. Файл PTIP.PLT подготовлен в среде пакета AutoCAD, поэтому допустимо его использование в среде MathCAD. Перед созданием ОРИ для рисунка из подготовленного MCAD-файла с именем Picture.mcs по команде Filename пакета с данным файлом ассоциируется переменная 'Pic', чтобы дать возможность адресоваться к файлу, находящемуся в неактивном каталоге. Создав затем по клавишам 'Alt + %' две ОРИ и поместив вместо их указателей имя ассоциированной с MCAD-файлом переменной 'Pic', форматируем ОРИ (одна использует формат по умолчанию, а другая изменяет значения параметров характеристик 'Center' и 'Box'). После этого нажимается функциональная клавиша 'F9' или пакет командой 'Auto' переводится в А-режим и начинается формирование рисунков поочередно в каждой из двух ОРИ. Полученные рисунки можно вывести затем на плоттер или принтер. При этом на печать выводится копия рисунка экрана. Область рисунка подобно любой области, исключая текстовые области и зоны, может командой 'Equation' пакета (раздел 2.5) выключаться из процесса вычислений в текущем документе. Описанная возможность позволяет в среде пакета MathCAD создавать достаточно хорошо иллюстрированные документы, что особенно важно при последующем использовании их, например, в качестве различного рода научно-технических отчетов, рукописей статей, книг и т. д. При этом выключение областей рисунков позволяет резервировать для них место в документе, не имея в данный момент самих готовых к помещению в документ рисунков. Это важно также при отладке и редактировании документа, так как многократная загрузка в документ рисунков требует существенных временных затрат.

Глава 10

ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ ПАКЕТА MathCAD ДЛЯ РЕШЕНИЯ ВЫЧИСЛИТЕЛЬНЫХ ЗАДАЧ

В главе рассматриваются непосредственные прикладные аспекты и возможности пакета MathCAD на примере ряда вычислительных задач. Круг задач определялся интересами авторов и той сферой, в которой проходила апробация пакета. На отбор приведенных примеров оказал влияние также ряд соображений педагогического и методологического характера. Выбирались примеры, достаточно обозримые по размеру и своей внутренней сути, позволяющие однако иллюстрировать как стандартные, так и нестандартные приемы работы в среде пакета MathCAD 2.52.

В предыдущих главах книги также проводилось достаточное число примеров, иллюстрирующих основные свойства и особенности пакета, но они, как правило, носили весьма узкий характер, акцентируя внимание читателя на отдельных аспектах изучаемого пакета. Примеры настоящей главы, включая иллюстрацию особенностей и возможностей пакета, вместе с тем демонстрируют комплексное использование пакета для решения тех или иных задач, что позволяет на качественно ином уровне оценивать его возможности для того или иного применения.

Предлагаемые примеры, во-первых, в основном ориентированы на иллюстрацию особенностей и основных возможностей пакета. Во-вторых, значительная их часть служит для обучения начинающего пользователя основным приемам работы по созданию в среде пакета различного типа документов (вычислительные задачи, графические методы, рукописи научно-технических статей и т. д.). И, наконец, целый ряд примеров, удовлетворяющих перечисленным целям, может быть использован в качестве прообраза или готовых документов в практической деятельности читателя, работающего с пакетом MathCAD на ПК, совместимых с IBM PC/XT/AT или PS/2. Все приведенные примеры снабжены необходимыми комментариями и отлажены в среде пакета MathCAD 2.52 на отечественном ПК ИСКРА 1030 в ПТИ Минстроя ЭССР в течение мая—декабря 1990 г.

Наконец, еще на одном моменте использования пакета стоит остановиться особо. Возможности пакета позволяют проводить в документе достаточно сложный анализ проблемы средствами современной вычислительной и классической математики. Однако как и подавляющее число современных пакетов того или иного назначения MathCAD не имеет универсальных средств организации произвольного вычислительного процесса или алгоритма (вычислительного или другого характера), как это имеет место в случае развитых систем программирования (Turbo-C, Turbo-Pascal, Turbo-Basic, Fortran-77 и др.), что в первую очередь объясняется назначением самого пакета.

Однако средства пакета позволяют в среде текущего документа использовать (при условии достаточной доступной памяти ПК) в случае необходимости другие программные средства для выполнения тех частей алгоритма, которые невозможно или нецелесообразно реализовывать в среде пакета. Это может быть стандартное эффективное средство вычисления функций, отсутствующих в пакете, система программирования, другой прикладной пакет и т. д. С этой целью, находясь в состоянии текущего документа, достаточно выполнить команду 'Dos' пакета с необходимыми параметрами, идентифицирующими требуемое программное средство, либо ввести эту же команду без параметров, дважды нажав клавишу 'Enter'. В первом случае загружается указанное параметрами команды 'Dos' программное средство, выполняет необходимую работу и после завершения ее по любой клавише можно вернуться обратно в состояние текущего документа. Во втором случае осуществляется выход в среду системы MS DOS и пользователь получает возможность использовать необходимые программные средства в любой их очередности (например,

текстовый редактор, Бейсик-программы, некоторые системные средства и т. д.). В этом случае возврат после завершения работы обратно в состояние текущего документа осуществляется по команде 'Exit' операционной системы MS DOS.

С учетом сказанного технология организации работы с использованием пакета MathCAD может принять следующий вид:

- (1) Работа в среде пакета (текущий документ)
↓
- (2) Переход в среду MS DOS и выполнение необходимой работы с сохранением ее результатов в дисковых файлах ASCII-формата и структуры, допускаемой пакетом MathCAD
↓
- (3) Возврат в среду пакета в состояние текущего документа по команде 'Exit', либо по любой клавише, ~~или~~ использовалась команда 'Dos' пакета с конкретным запросом средства
↓
- (4) Продолжение работы в среде пакета с использованием полученных на этапе (2) результатов посредством созданных дисковых файлов (текстовых, числовых или смешанных)

Приведенная схема носит достаточно общий характер и с учетом конкретных условий может весьма эффективно применяться при решении в среде пакета многих задач, значительно расширяя возможности самого пакета и позволяя совместно использовать все преимущества пакета MathCAD и других программных средств различного назначения. Так, для поддержания данной технологии нами была создана на базе системы MINIDOS [10] и пакета MathCAD 2.52 система MINIDOS/MathCAD, ориентированная в первую очередь на отечественный ПК ИСКРА 1030 и зарекомендовавшая себя достаточно хорошо в ряде организаций страны. Данная система позволяет пользователю на ПК без «Винчестера» и с двумя НГМД без дополнительной переконфигурации магнитных томов использовать в среде текущего документа пакета такие программные средства, как команды MS DOS, текстовый редактор, систему программирования Бейсик и некоторые дополнительные сервисные средства.

Для автоматизации обработки документов в среде пакета можно использовать механизм командных файлов, создаваемых текстовым процессором или иным допустимым способом. По умолчанию для такого файла полагается расширение имени '.MCC'. В качестве простого примера приводится система из двух последовательно зависящих MCC-файлов 'PTI.MCC' и 'PTIP.MCC', автоматизирующая выполнение в среде пакета MathCAD некоторый вариант технологии решения задач, ввод исходных данных для которых реализован в среде языка программирования БЕЙСИК. Оба командных файла имеют следующий набор команд пакета:

PTI.MCC	PTIP.MCC
Manual	Load MINSTROJ
Clear	Append SERVIS
Dos Basic DATAVOD	Process
Execute PTIP Print	Print

Согласно приведенным текстам MCC-файлов после выполнения в среде пакета команды 'Execute PTI' происходит следующее. Пакет переводится в М-режим, память ПК очищается от текущего документа с перезагрузкой конфигурации пакета, после чего пакет командой 'Dos' переводится в состояние MS DOS с загрузкой БЕЙСИКа и выполнением в его среде программ 'DATAVVOOD', подготавливающей файлы исходных данных для документа 'MINSTROJ'. После завершения данной программы по любой клавише осуществляется возврат в среду пакета MathCAD с последующим выполнением команды 'Execute', инициирующей выполнение следующего командного файла PTIP. MCC. В результате чего загружается документ 'MINSTROJ', дополняется библиотекой 'SERVIS' функций пользователя, командой 'Process' осуществляется вычисление всего текущего документа, а затем он выводится на печать.

Переходим теперь непосредственно к примерам использования пакета для решения вычислительных задач. Каждый пример содержит необходимый минимум пояснений, помещенных в сам документ, содержащий тот или иной пример, а само содержание примера полностью отвечает соответствующему отлаженному документу пакета.

1. Графическое дифференцирование и интегрирование функций на заданном интервале [A, B]:

A: = -2 B: = 2 H: = 0.1 X: = A, A + H .. B P: = -10 ..30

$$Y(X) := X^3 + 3 \cdot X^2 - 2 \cdot X + 5 \quad Z(X) := \frac{d}{dX} Y(X)$$

$$R(X) := \frac{d}{dX} \frac{d}{dX} Y(X) \quad W(X) := \int_{-2}^X Z(X) dX \quad V(X) := \int_{-2}^X \int_{-2}^X R(X) dX dX$$

Команда 'Plotformat': Size = 15,25
Trace Types = LLLDP

Рис. 40

В левой графической области выведены графики соответственно функции $Y(X)$ (сплошная линия), ее первой (пунктирная линия) и второй (плюс — линия) производных, а также оси декартовой системы координат; в правой — соответственно графики функции $Y(X)$, а также одно- и двукратный интегралы соответственно от первой и второй производных данной функции. Как и должно быть, все три кривые совпали. Перед выводом графиков обе области были отформатированы командой 'Plotformat' единым образом.

2. Табулирование вычисления сумм числового ряда с заданной точностью E:

E: = 10^{-6} ∞: = 10^8 N: = 0 .. ∞ P: = (4 6 8)

$$R(N, P) := \left[\frac{(-1)^N}{(2 \cdot N + 1)^{P/2}} \right]$$

$$j := 0 \dots \sum_N \text{until } (|R(N, P)| - E, N \cdot \Phi(E - |R(N, P)|))$$

$$S(P) := \sum_j R(j, P) \quad S(P)^T = (0.988945 \quad 0.998685 \quad 0.99985)$$

$$P := \begin{bmatrix} 3 \\ 4 \end{bmatrix} \quad R(N, P) := \left[\frac{1}{(2 \cdot N + 1)^P} \right]$$

$$j := 0 \dots \sum_N \text{until} (|R(N, P)| - E, N \cdot \Phi(E - |R(N, P)|))$$

$$S(P) := \sum_j R(j, P) \quad S(P) = \begin{bmatrix} 1.051776 \\ 1.014678 \end{bmatrix}$$

Примеры числовых рядов взяты из книги [24]. Для вычисления сумм в зависимости от параметра P с точностью $E = \text{TOL}$ использовались векторные представления значений параметра и векторизация функции, представляющей n -й член ряда. Для определения конца суммирования с заданной точностью использована специальная конструкция на основе встроенных функции 'until' и функции Хэвисайда. Данная и подобные ей конструкции весьма эффективны при организации ряда вычислительных процессов с заданным условием выхода. Другие подобные примеры будут приведены ниже.

3. Вычисление нулевых корней уравнения по их начальным приближениям с точностью, определяемой переменной TOL пакета:

$X := -4, -1 \dots V := i$ Определение начальных приближений

$$F(X) := X^6 - 4.21 \cdot X^5 - 29.89 \cdot X^4 + 71.63 \cdot X^3 + 60.53 \cdot X^2 - 89.54 \cdot X + 45.18$$

$$\text{root}(F(V), V) = 0.46997 + 0.42322i \quad \text{root}(F(X), X)$$

—4.67061
—1.42066
2.47041
6.89092

Ввиду действительных коэффициентов уравнения $F(X) = 0$ шестым корнем является значение, сопряженное корню $X = 0.46997 + 0.42322i$, т. е. $X^* = 0.46997 - 0.42322i$

При использовании встроенной функции 'root' пакета для вычисления нулевых корней функции рекомендуется использовать для определения начальных значений p -переменные, а не вектора или матрицы с последующей векторизацией функции 'root', ибо возможны аварийные ситуации с диагностикой 'Underflow', или 'Out of MEMORY'. Данный пример в качестве прототипа можно использовать при решении линейных однородных и неоднородных дифференциальных уравнений n -го порядка с постоянными коэффициентами, решение которых сводится к решению соответствующего характеристического полинома n -й степени.

4. Табуляция эллиптических интегралов первого и третьего типов с точностью, определяемой переменной TOL пакета:

$k := 0.1, 0.2 \dots 0.3 \quad \varphi := 0.1 \cdot \pi, 0.6 \cdot \pi \dots 2 \cdot \pi \quad n := 3$

$$A(\varphi) := \sin(\varphi)^2$$

$$F(\varphi, k) := \int_0^\varphi \frac{1}{\sqrt{1 - k^2 \cdot A(\varphi)}} d\varphi$$

$$\Pi(\varphi, n, k) = \int_0^{\varphi} \frac{1}{[1 + n \cdot A(\varphi)] \cdot \sqrt{1 - k^2 \cdot A(\varphi)}} d\varphi$$

F(φ, k)	Π(φ, n, k)
0.31421	0.28817
0.31436	0.28830
0.31462	0.28852
1.89044	0.86763
1.90729	0.87287
1.93685	0.88202
3.46370	1.86160
3.48810	1.86979
3.53071	1.88401
5.03993	2.44106
5.08103	2.45436
5.15295	2.47751

Приведенный пример табуляции интегралов пояснений не требует.

5. Вычисление несобственных интегралов с точностью, определяемой переменной TOL пакета. Примеры интегралов взяты из книги [24], где даются их значения.

а). Определяются исходные данные для интегрирования:

$$E := TOL \quad P := 1 \quad H := 10^P \quad \infty := 10^{P+8} \quad Y := H, H + P \dots \infty$$

$$F(X) := \ln[1 + e^{-X}] \text{ — Подынтегральная функция}$$

$$\int_0^{\infty} F(X) dX = ? \text{ — Искомое значение} \quad J(Y, Z) := \int_Z^Y F(X) dX$$

Формула вычисления интеграла с проверкой условия выхода

$$J := \sum_Y \text{until} (|J(Y + P, Y)| - E, J(Y + P, 0) \cdot \Phi(E - |J(Y + P, Y)|)) \cdot 0.5$$

Результат вычисления следует делить на 2, ибо ранжирование по переменной Y идет дважды, удваивая получаемый результат

$$\frac{\pi^2}{12} = 0.82247 \text{ Точное значение; } J = 0.82246 \text{ Вычисление}$$

б). A := 48 B := 43 M := 0.89 Определение констант

$$F(X) := \ln \left[\frac{A^2 + X^2}{B^2 + X^2} \right] \cdot \cos(M \cdot X) \quad J(Y, Z) := \int_Z^Y F(X) dX$$

$$J_1 = \sum_Y \text{until} (|J(Y+P, Y)| - E, J(Y+P, 0) \cdot \Phi(E - |J(Y+P, Y)|)) \cdot 0.5$$

$$\pi \cdot \frac{e^{-B \cdot M} - e^{-A \cdot M}}{M} = 8.36004 \cdot 10^{-17} \quad \text{Точное значение: } J = 0$$

с). A: = 43 B: = 48 C: = 23 Определение констант

$$F(X) := \frac{1}{A \cdot X^4 + 2 \cdot B \cdot X^2 + C} \quad J(Y, Z) := \int_Z^Y F(X) dX$$

$$J_1 = \sum_Y \text{until} (|J(Y+P, Y)| - E, J(Y+P, 0) \cdot \Phi(E - |J(Y+P, Y)|)) \cdot 0.5$$

$$\frac{\pi}{2 \cdot \sqrt{C} \cdot \sqrt{2 \cdot B + 2 \cdot \sqrt{A \cdot C}}} = 0.026 \quad \text{Точное значение: } J = 0.026$$

д). Особенность на обоих концах интервала интегрирования:

$$F(X) := \ln \left[\frac{1 - e^{-X}}{1 + e^{-X}} \right] \quad \text{Особенность на обоих концах интервала}$$

$$J(Y, Z) := \int_0^\infty F(X) dX \quad \text{Определение интеграла как функции пределов интегрирования}$$

A: = 1 Выбор точки, делящей интервал интегрирования

Вычисление второй части интеграла на интервале $[A, \infty]$:

$$J_2 = \sum_Y \text{until} (|J(Y+P, Y)| - E, J(Y+P, A) \cdot \Phi(E - |J(Y+P, Y)|)) \cdot 0.5$$

Вычисление первой части интеграла на $[0, A]$:

H: = A P: = -10^{-4} Y: = H, H + P .. 0 Деление подынтервала на шаги

$$J_1 := \sum_Y \text{unti} (|J(Y, Y+P)| - E, J(Y+P, A) \cdot \Phi(E - |J(Y, Y+P)|)) \cdot 0.5$$

$$-\frac{\pi^2}{4} = -2.4674 \quad \text{Точный результат: } J_1 + J_2 = -2.46729$$

Если вычисляемый интеграл расходится, то можно либо уменьшить значение переменной TOL пакета, определяющей точность интегрирования, либо увеличить значение переменной P, определяющей величину шага при вычислении условия конца интегрирования. Приведенные примеры вычисления несобственных интегралов предлагают определенную методику, но не могут служить аналогом для всех возможных случаев. Например, в ряде случаев может потребоваться выбор значения для бесконечности (∞), чтобы не вызвать ситуации 'Overflow', либо дополнительный графический анализ поведения подынтегральной функции $F(X)$. Однако предложенный подход может быть положен в основу вычисления подобного типа интегралов. Он достаточно прозрачен и его эффективность вполне удовлетворительна.

6. Решение системы линейных уравнений в матричном виде:

$$A := \begin{bmatrix} 0.42 & 0.47 & 0.67 \\ 1.89 & -3.2 & 8.3 \\ -6.4 & 5.62 & 9.83 \end{bmatrix} \quad B := \begin{bmatrix} 1.48 \\ 2.43 \\ 3.23 \end{bmatrix} \quad \begin{array}{l} \text{Матрица коэффициентов} \\ \text{и столбец свободных} \\ \text{членов} \end{array}$$

Вводятся команды: Filename X C: Matrix. A; Filename Y B: Vector. B
WRITEPRN (X): = A WRITEPRN (Y): = B Запись данных в СТ-файлы

Решение системы ищется в матричном виде: $R = A^{-1}B$

$R := [\text{READPRN}(X)^{-1} \cdot \text{READPRN}(Y)]^T$ Результат решения

$\text{WRITEPRN}(Z) := R$ Сохранение транспонированного результата

$\text{READPRN}(Z) = (1.342 \quad 1.27 \quad 0.4766)$ Визуализация результата

Примечание: Так как в функции 'if' пакета использование функций доступа к данным не всегда корректно, то в предложенном алгоритме решения системы затруднительно осуществить проверку на невырожденность матрицы A коэффициентов системы. Для осуществления такого контроля функцию 'if' следует использовать в конструкции, позволяющей решать систему с контролем на сингулярность:

$$R := \text{if } [|A|, A^{-1} \cdot B, 0], R = \begin{bmatrix} 1.342 \\ 1.27 \\ 0.477 \end{bmatrix}$$

В такой конструкции функция возвращает вектор-решение системы, если матрица A невырождена, в противном случае возвращается нулевое скалярное значение.

7. Определение в среде пакета функций алгебры логики. Пакет непосредственно не предоставляет средств для использования булевых функций, однако приводимые ниже определения основных (базовых) функций алгебры логики через встроенные функции пакета позволяют легко устранить это ограничение

$$X := 0 \dots 1 \quad Y := 0 \dots 1 \quad Z := 0 \dots 1$$

$$\text{NOT}(X) := \text{if}(X, 0, 1) \quad \text{EQV}(X, Y) := \delta(X, Y) \quad \text{AND}(X, Y) := X \cdot Y$$

$$\text{PRS}(X, Y) := \text{if}(X + Y, 0, 1) \quad \text{SEF}(X, Y) := \text{if}(\text{if}(X \cdot Y, 1, 0), 0, 1)$$

$$\text{XOR}(X, Y) := \text{mod}(X + Y + 2, 2) \quad \text{IMP}(X, Y) := X \cdot Y + \text{if}(X, 0, 1)$$

$$\text{OR}(X, Y) := \Phi(X + Y - 1)$$

Определенные таким образом функции NOT, EQV, AND, PRS, SEF, XOR, IMP и OR задают соответственно булевы функции отрицания, эквивалентности, конъюнкции, стрелки Пирса, штрих Шеффера, разделительной дизъюнкции, импликации и дизъюнкции, о чем свидетельствуют их таблицы истинности, приводимые ниже:

NOT (X)	EQV (X, Y)	PRS (X, Y)	SEF (X, Y)	XOR (X, Y)
1	1	1	1	0
0	0	0	1	1
	0	0	1	1
	1	0	0	0
OR (X, Y)	AND (X, Y)	IMP (X, Y)	EQV (AND (X, Y), PRS (X, Y))	
0	0	1	0	
1	0	1	1	
1	0	0	1	
1	1	1	0	

AND (OR (X, Y), OR (NOT (X), NOT (Y)))

0
1
1
0

OR (AND (X, NOT (Y)), AND (NOT (X), Y))

0
1
1
0

На основе определенных восьми функций в среде пакета легко реализуется алгебра логики для K -местных ($K \geq 2$) булевых функций. В системе MINIDOS / MathCAD данные булевы функции, определенные глобально, составляют файл BOOL.MCD, который по мере надобности по команде 'Arpend' пакета может дописываться в конец текущего документа, позволяя использовать средства алгебры логики.

8. Определение некоторых полезных функций пользователя:

а). Функция целочисленного деления аргументов:

$$\text{Idiv}(X, Y) \equiv \text{floor} \left[\frac{\text{floor}(X) + \text{floor}(2 \cdot \text{mod}(X, \text{floor}(X)))}{\text{floor}(Y) + \text{floor}(2 \cdot \text{mod}(Y, \text{floor}(Y)))} \right].$$

б). Функция вычисления k -й десятичной цифры числа X :

$$\text{Nfig}(X, K) \equiv \left\lfloor \text{floor} \left[\frac{\text{floor}[X \cdot 10^K]}{10} \right] - \frac{\text{floor}[X \cdot 10^{K-1}]}{10} \right\rfloor \cdot 10$$

Определение номера K десятичной цифры числа X производится согласно схеме: $-K \dots -2 -1 0 \mid \dots$ (точка) $\mid 1 2 3 \dots K \dots$

с). Решение системы линейных уравнений с контролем сингулярности матрицы коэффициентов:

$\text{Sol}(A, B) \equiv \text{if} [|A|, A^{-1} \cdot B, 0]$ Функция возвращает вектор-решение, если $|A| \neq 0$, и 0 в противном случае

д). Вычисление собственного значения квадратной матрицы:

$F(A, \lambda) \equiv \text{if} (\text{rows}(A) - \text{cols}(A), 0, \text{root}(|A - \lambda \cdot \text{identity}(\text{rows}(A))|, \lambda))$

При приемлемом начальном значении для λ возвращает величину собственного значения в случае квадратной матрицы A и ноль, если матрица не является квадратной.

е) Вычисление K -го эмпирического центрального момента вектора произвольной выборки X ($K > 1$):

$$\text{Mom}(X, K) \equiv \left[\frac{\sum (X - \text{mean}(X))^K}{K - 1} \right]$$

ф). Функция знака аргумента (Sign):

$$\text{Sign}(X) \equiv \text{if} \left[X, \text{if} \left[\frac{X}{|X|} + 1, 1, -1 \right], 0 \right]$$

$X = 589.5497 \quad Y = 2.78 \quad \text{Idiv}(X, Y) = 196 \quad \text{Nfig}(X, -2) = 5$

$$A = \begin{bmatrix} 0.42 & 0.47 & 0.67 \\ 3.6 & -5 & 1.7 \\ -7.9 & -2.3 & 7 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad X = \begin{bmatrix} 1.4 \\ 1.52 \\ 1.77 \end{bmatrix}$$

$$\begin{aligned} F(A, 0) &= 2.36 \\ \text{Mom}(X, 3) &= 0.002 \quad \text{Sol}(A, B) = \begin{bmatrix} 0.448 \\ 0.27 \\ 1.023 \end{bmatrix} \\ \text{Sign}(F(A, 0)) &= 1 \end{aligned}$$

g). Функция Е-толерантности аргумента (Tol):

$$\text{Tol}(X, E) \equiv \text{if}(\Phi(E - |X|), 0, X)$$

$$E: = 0.0025 \quad \text{Tol}(0.001, E) = 0 \quad \text{Tol}(0.0026) = 0.0026$$

Функция возвращает нулевое значение при $|X| \leq E$ и X в противном случае, т. е. определяет Е-толерантность аргумента X .

h). Ступенчатые функции действительного аргумента:

$$U(X) := \text{if}(X, \text{if}(|X| + X, 1, 0), 0.5) \quad U_p = \text{if}(X, \text{if}(|X| - X, 0, 1), 0)$$

$$U_m := \text{if}(X, \text{if}(|X| + X, 1, 0), 1) \quad K := -1 \dots 1$$

U (K)	U _p (K)	U _m (K)
0	0	0
0.5	0	1
1	1	1

Подробную информацию по данным функциям можно найти в книге [26].

i). Некоторые часто используемые интегральные функции [26]:

$$\text{Si}(X) := \text{if}\left[X, \int_{\text{TOL}^2}^X \frac{\sin(X)}{X} dX, 0\right] \quad \text{Интегральные синус и косинус}$$

$$\text{Ci}(X) := \text{if}\left[X, 0.577216 + \ln(X) - \int_{\text{TOL}^2}^X \frac{1 - \cos(X)}{X} dX, 0\right] \quad (X > 0)$$

$$\text{Ei}(X) := \text{if}\left[X, \int_{-30}^X \frac{e^X}{X} dX, \infty\right] \quad (X < 0) \quad \text{Интегральная экспонента}$$

$$\text{Li}(X) := \text{if}\left[X, \int_{\text{TOL}^2}^X \frac{1}{\ln(X)} dX, 0\right] \quad (X > 0) \quad \text{Интегральный логарифм}$$

Данные функции полностью заменяют соответствующие таблицы [26].

j). Функция определения принадлежности аргумента отрезку $[A, B]$:

$$\rho(X, A, B) := \text{if}(\Phi(X - B), 0, \text{if}(\Phi(X - A), 1, 0))$$

$$A: = 5 \quad B: = 9 \quad \rho(0, A, B) = 0 \quad \rho(5, A, B) = 1$$

$$\rho(8, A, B) = 1 \quad \rho(9, A, B) = 0 \quad \rho(15, A, B) = 0$$

Функция возвращает единицу, если значение аргумента принадлежит отрезку $[A, B]$ (исключая правый его конец), и ноль в противном случае. Функция полезна во многих математических расчетах. Так с ее помощью легко вводятся определения кусочно заданных функций, например:

$$A: = \infty \quad B: = 0 \quad C: = 5 \quad D: = 50$$

$$F(X) := \text{if}[\rho(X, A, B), \sin(X), \text{if}[\rho(X, B, C), \cos(X), \text{if}[\rho(X, C, D), \ln(X), e^X]]]$$

$$F(-6) = 0.27942 \quad F(3) = -0.98999 \quad F(6) = 1.79176$$

9. Дифференцирование интегралов по параметру и неявных функций: $A: = 1 \quad B: = 3$

$$U(\lambda) := \sin(\lambda) \quad W(\lambda) := \lambda \cdot \cos(\lambda) \quad F(X, \lambda) := \ln(X + \lambda)$$

$$Y(\lambda) := \frac{d}{d\lambda} \int_A^B F(X, \lambda) dX \quad Z(\lambda) := \frac{d}{d\lambda} \int_{U(\lambda)}^{W(\lambda)} F(X, \lambda) dX$$

$$Y(1) = 0.693 \quad Z(2) = -0.815 \quad R(X, Y) = \ln(X + Y) - \sin(X \cdot Y) + X \cdot Y$$

$$D(X, Y) = (-1) \cdot \frac{\frac{d}{dX} R(X, Y)}{\frac{d}{dY} R(X, Y)} \quad D(1, 2) = 2$$

10. Решение систем векторных уравнений разного типа:

$$P: = 48 \quad R: = 43 \quad S: = 23 \quad A(B \times C) = 0 \quad \langle - \text{Условие}$$

$$A: = \begin{bmatrix} -0.42 \\ 0.47 \\ 0.67 \end{bmatrix} \quad B: = \begin{bmatrix} 1 \\ -2 \\ 3 \end{bmatrix} \quad C: = \begin{bmatrix} 1.48 \\ 2.43 \\ -3.89 \end{bmatrix} \quad D: = \begin{bmatrix} 1.3 \\ -4.5 \\ 2.2 \end{bmatrix}$$

Уравнения и системы

Решения в среде пакета

$$XA = P$$

$$X: = A \cdot \frac{P}{A \cdot A} \quad X = 48$$

$$XA = P$$

$$XB = R$$

$$X: = \frac{P \cdot (B \times C) + R \cdot (C \times A) + S \cdot (A \times B)}{A \cdot (B \times C)}$$

$$XC = S$$

$$AX + BY + CZ + D = 0 \quad X: = -\frac{D \cdot (B \times C)}{A \cdot (B \times C)} \quad Y: = -\frac{A \cdot (D \times C)}{A \cdot (B \times C)}$$

$$Z: = -\frac{A \cdot (B \times D)}{A \cdot (B \times C)}$$

$$\sqrt{B \times C}X + (C \times A)Y + (A \times B)Z + D = 0 \quad X: = -\frac{D \cdot A}{A \cdot (B \times C)} \quad Y: = -\frac{D \cdot B}{A \cdot (B \times C)}$$

$$Z: = -\frac{D \cdot C}{A \cdot (B \times C)}$$

11. Исследование систем функциональных неравенств на основе решающих блоков пакета:

$$X: = 2 \quad Y: = 2 \quad F(X, Y) = X^2 + Y^2 + 0.42 \cdot X \cdot Y$$

GIVEN

$$0.48 \cdot X + 0.43 \cdot X \cdot Y \leq 1.42 \cdot \int_0^X e^X dx$$

$$0.23 \cdot X \cdot Y + 1.7 \cdot Y < 1.89 \cdot \int_0^Y \sqrt{Y} dY$$

$$\left| \left[\frac{d}{dX} \frac{d}{dX} F(X, Y) \quad \frac{d}{dX} \frac{d}{dY} F(X, Y) \right] \right| > 0$$

$$R_{avz}(X, Y) = \text{FIND}(X, Y) \quad [R_{avz}]^T = (2 \quad 3.475)$$

12. Исследование системы параметрических нелинейных уравнений с помощью решающих блоков:

Начальные условия для искомого решения и параметров системы

$$X: = 0 \quad Y: = 0 \quad Z: = 0 \quad A: = 4.3 \quad B: = 1.7 \quad C: = 2.3$$

Решающий блок, содержащий исследуемую систему

$$\text{Given} \quad X^4 + Y^4 + Z^4 \approx B$$

$X \cdot Y + X^2 \cdot Z + \sin(X + Y + Z) \approx A \quad e^{X+Y} - 3 \cdot Y \cdot Z + X \approx C$
 Sol (A, B, C):= FIND (X, Y, Z) Примеры решения данной системы:
 Sol (-2.3, 6.7, 3.8)^T = (1.095 -0.038 -1.515)
 Sol (-1.5, 8, 4.7)^T = (0.81 0.219 -1.659)
 Sol (0.5, 2.2, 5)^T =

did not find solution

Для более широкого исследования подобных систем уравнений можно использовать данный подход, сочетая пользовательские функции блока на основе как функции FIND, так и MINERR. Сказанное относится и к системам нелинейных параметрических неравенств.

13. Исследование задач линейного и нелинейного программирования методом решающих блоков. Задачи данного класса, наряду с сугубо теоретическим, представляют большой прикладной интерес. При этом, если в случае линейного программирования имеются хорошо развитые методы, то задачи нелинейного программирования, за малым исключением, поддаются лишь численным методам решения, что делает весьма актуальным поиск таких методов. В приводимых примерах иллюстрируется возможность использования для этих целей средств пакета MathCAD:

а). Вычисление минимума целевой функции, удовлетворяющей определенному условию

Начальные значения для переменных и условий

A:=100 D:=90 X:=1 Y:=1 Z:=1 T:=1 U:=1

$Q_{\min}(X, Z) := -52 \cdot X + 35 \cdot Z + 109$ Целевая функция, для которой ищется минимум R с точностью E = 0.01 при $A \leq R \leq B$

Решающий блок Given $X \geq 0 \quad Y \geq 0 \quad Z \geq 0 \quad T \geq 0$

$3 \cdot X + Y + 2 \cdot Z \approx 5 \quad X - 4 \cdot Z + T \approx 1 \quad U \geq 0$

$-X + 3 \cdot Z + U \approx 9 \quad Q_{\min}(X, Z) \leq A \quad Q_{\min}(X, Z) \geq D$

R (X, Y, Z, T, U, A, D):= FIND (X, Y, Z, T, U). Параметрическая функция, позволяющая путем варьирования параметра A отыскивать минимум целевой функции. Первый этап ранжирования параметра A:

j:=0..10 B_j:=100-j MR_(j):=R [X, Y, Z, T, U, B_j, D]

QR_j:=Q [MR_{0,j} MR_{2,j}] Определение матрицы MR решений и вектора QR значений целевой функции при этих решениях

QR^T=(100 99 98 97 96 95 94 93 92 91) При A = 90 решения нет

Второй этап ранжирования: B_j:=90.5 - 0.1 · j

QR^T=(90.5 90.4 90.3) При A = 90.2 решения блока нет

Третий этап (на уровне точности E): B_j:=90.25 - 0.01 · j

QR^T=(90.25 90.24 90.23). При A = 90.22 решения блока нет

Следовательно минимум целевой функции Q равен QR₂=90.23 в точке [MR⁽²⁾]^T=(0.951 0.391 0.877 3.557 7.32).

б). Минимизация целевой функции методом решающего блока и сравнение результата с решением задачи симплекс-методом:

A:=1 X:=1 Y:=1 Z:=1 U:=2 Нач. значения

$Q_{\min}(X, U) := 0.25 \cdot X + 0.33 \cdot U$ Целевая функция

Given $5 \cdot X - Y + U \approx 5 \quad 2 \cdot X - Z + 5 \cdot U \approx 10$

$X \geq 0 \quad Y \geq 0 \quad Z \geq 0 \quad U \geq 0$

$Q_{\min}(X, U) \leq A \quad Q_{\min}(X, U) \geq 0$ /

$R(X, Y, Z, U, A) := \text{FIND}(X, Y, Z, U)$. Для получения искомого минимума целевой функции исходя из общей ее характеристики и линейных ограничений организуем просмотр решений блока на первом уровне ранжирования параметра A функции блока:

$$j := 0..2 \quad B_j := 1 - 0.1 \cdot j \quad MR^{(j)} := R[X, Y, Z, U, B_j]$$

$$QR_j := Q_{\min}[MR_{0,j}, MR_{3,j}] \quad QR^T = (0.823 \ 0.9 \ 0.8)$$

Решение блока при значении $A = 0.7$ отсутствует. Затем подобно предыдущему примеру (а) проводятся аналогичным образом ранжированные по параметру A просмотры решений блока до требуемой точности $E = 0.001$ и в результате такого анализа выясняется наличие решения блока при $A = 0.737$ в точке $R(X, Y, Z, U, 0.737)^T = (0.652 \ 2.062 \cdot 10^{-10} \ 4.034 \cdot 10^{-10} \ 1.739)$, т. е. минимум целевой функции равен $Q_{\min}(0.652, 1.739) = 0.737$. Результат применения симплекс-метода для решения данной задачи дает минимум целевой функции равный $Q_{\min} = 0.743$, т. е. абсолютная разница не превышает вполне удовлетворительной величины $\varepsilon = 0.006$.

с). Задача целочисленного программирования. Требуется найти минимум целевой функции при условии, что переменные могут принимать только неотрицательные целые значения. Для решения нам потребуется определить целочисленную функцию $\text{Int}(X)$ аргумента следующим образом:

$$D(X) := X - \text{floor}(X) - 0.5 \quad L(X) := X - D(X) - 0.5 \quad X := 1 \quad Y := 1 \quad A := 0$$

$$\text{Int}(X) := \text{if}(D(X), L(X) + \Phi(D(X), \text{if}(\text{mod}(L(X), 2), L(X) + 1, L(X)))$$

$$Q_{\min}(X, Y) := -(X + Y) \quad \text{Целевая функция: } X, Y - \text{целые числа}$$

$$\text{Given } 2 \cdot \text{Int}(X) + \text{Int}(Y) \leq 4 \quad 2 \cdot \text{Int}(X) + 3 \cdot \text{Int}(Y) \leq 6$$

$$X \geq 0 \quad Y \geq 0 \quad Q_{\min}(\text{Int}(X), \text{Int}(Y)) \leq A$$

$R(X, Y, A) := \text{FIND}(X, Y)$ Принимая во внимание целочисленность значений целевой функции при целочисленности ее переменных, легко организовать ранжирование параметра A блока с целью анализа получаемых решений при целочисленных переменных X и Y .

$$B := -2 \quad \text{ORIGIN} := B \quad A := 0, -1 \dots B \quad MR(A) := R(X, Y, A)$$

$\text{Min}_A := Q_{\min}[\text{Int}[MR_B, A], \text{Int}[MR_{B+1}, A]]$ Значения функции Результат p -вычисления по параметру A решающего блока:

$$\overrightarrow{\text{Int}(MP)} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \text{Min}^T = (-2 \ -2 \ -2)$$

Таким образом, минимальное значение функция принимает в точке $(X, Y) = (1, 1)$. Геометрический анализ легко подтверждает данный результат.

d). Задача нелинейного программирования. Требуется найти максимум нелинейной функции при нелинейных условиях с точностью $E = 0.01$. Решение ищется также в среде пакета MathCAD:

$$X := 0.25 \cdot \sqrt{2} \quad Y := 0.25 \cdot \sqrt{2} \quad AB := 0 \quad Z_{\max}(X, Y) := \sin(X + Y)$$

$$\text{Given } X^2 + Y^2 \leq 1 \quad X \geq 0 \quad Y \geq 0 \quad Z_{\max}(X, Y) \geq AB$$

$R(X, Y, AB) := \text{FIND}(X, Y)$ Решение ищется на основе анализа поэтапного ранжирования блока по параметру AB подобно описанным выше способам:

$$j := 0..8 \quad A_j := 0.1 \cdot (1 + j)$$

$$MR^{(j)} := R[X, Y, A_j] \quad \text{Max}_j := Z_{\max}[MR_{0,j}, MR_{1,j}]$$

$$MR = \begin{bmatrix} 0.354 & 0.354 & 0.354 & 0.354 & 0.354 & 0.354 & 0.42 & 0.552 & 0.683 \\ 0.354 & 0.354 & 0.354 & 0.354 & 0.354 & 0.354 & 0.356 & 0.376 & 0.437 \end{bmatrix}$$

$$\text{Max } T = (0.65 \ 0.65 \ 0.65 \ 0.65 \ 0.65 \ 0.65 \ 0.7 \ 0.8 \ 0.9)$$

При $j = 9$ ($A = 1$) решение блока отсутствует; 2-й этап ранжирования:

$$j := 0..7 \quad A_j := 0.9 + 0.01 \cdot (1 + j)$$

$$MR^{(j)} := R[X, Y, A_j] \quad \text{Max}_j := Z_{\max}[MR_{0,j}, MR_{1,j}]$$

$$MR = \begin{bmatrix} 0.704 & 0.719 & 0.723 & 0.735 & 0.749 & 0.762 & 0.778 & 0.788 & 0.683 \\ 0.44 & 0.449 & 0.472 & 0.487 & 0.505 & 0.525 & 0.547 & 0.582 & 0.437 \end{bmatrix}$$

$$\text{Max}^T = (0.91 \ 0.92 \ 0.93 \ 0.94 \ 0.95 \ 0.96 \ 0.97 \ 0.98 \ 0.9)$$

Таким образом, с точностью $E = 0.01$ максимум целевой функции достигается в точке $(X, Y) = (0.788, 0.582)$ и равен $Z_{\max} = 0.98$.

14. Вычисление экстремумов функций многих переменных.

а) Максимум и минимум функции двух переменных. Требуется вычислить локальные максимум и минимум функции

$$F(X, Y) := 3 \cdot X^3 - X + Y^3 - 3 \cdot Y^2 - 1$$

Для этого используем решающий блок с двумя параметрами А, В и условием существования экстремума функции от двух переменных

$$A := 1 \quad B := 1$$

Given

$$\frac{d}{dX} F(X, Y) \approx 0 \quad \frac{d}{dY} F(X, Y) \approx 0 \quad A \cdot \frac{d}{dX} \frac{d}{dX} F(X, Y) > 0$$

$B \cdot Y \geq B$ $R(A, B) := \text{FIND}(X, Y) \quad m := R(1, B) \quad M := R(-1, B)$
 Параметр А позволяет разделять минимум и максимум функции, а параметр В служит для получения всех экстремальных точек в исследуемой области (m — минимум; M — максимум). С учетом сказанного следующие выражения достаточно очевидны:

$$\text{Max} := F[M_0, M_1] \quad \text{Min} := F[m_0, m_1] \quad \text{Min} = -5.222 \quad \text{Max} = -4.778$$

$m^T = (0.333 \ 2) \quad M^T = (-0.333 \ 2)$ Изменяем значение параметра В для отделения полученных экстремальных точек:

$$m := R(1, -1) \quad M := R(-1, -1) \quad \text{Max} := F[M_0, M_1]$$

$$\text{Min} := F[m_0, m_1] \quad \text{Min} = -1.222 \quad \text{Max} = 0.778 \quad m^T = (0.333 \ 0) \quad M^T = (-0.303 \ 0)$$

Таким образом, локальный минимум функции Min = -5.222 в точке (0.333, 2), а максимум Max = 0.778 в точке (-0.333, 0).

б). Вычисление и анализ экстремума функции двух переменных:

$$X := 0 \quad Y := 0 \quad M := 10^{250} \quad m := 10^{200}$$

$$F(X, Y) := 0.5 \cdot X^2 - 4 \cdot X \cdot Y + 9 \cdot Y^2 + 3 \cdot X - 14 \cdot Y + 0.5 \quad \text{Функция}$$

Given

$$\frac{d}{dX} F(X, Y) \approx 0 \quad \frac{d}{dY} F(X, Y) \approx 0$$

$$R := \text{FIND}(X, Y) \quad X := R_0 \quad Y := R_1 \quad X = 1 \quad Y = 1$$

Функция 'if' используется для определения типа экстремума, ибо решение блока возвращает только самую экстремальную точку:

$$\text{if} \left[\left| \left[\begin{array}{l} \frac{d}{dX} \frac{d}{dX} F(X, Y) \quad \frac{d}{dX} \frac{d}{dY} F(X, Y) \\ \frac{d}{dY} \frac{d}{dX} F(X, Y) \quad \frac{d}{dY} \frac{d}{dY} F(X, Y) \end{array} \right] \right| \right] > 0, \text{if} \left[\frac{d}{dX} \frac{d}{dX} F(X, Y) > 0, \right.$$

$$\left. F(X, Y) \cdot m, \text{if} \left[\frac{d}{dX} \frac{d}{dX} F(X, Y) < 0, F(X, Y) \cdot M, 2i \right] \right], i = -5 \cdot 10^{200}.$$

При такой организации вычислений, если функциональный определитель имеет нулевое значение, то возвращается результат «i»; иначе (если вторая производная функции равна нулю) возвращается результат «2i»; максимальное значение идентифицируется коэффициентом $k_1 = 10^{350}$ и минимальное коэффициентом $k_2 = 10^{200}$. При значениях «i», «2i» требуется дополнительное исследование функции. Таким образом, в точке (1, 1) функция имеет Min = -5.

15. Замена переменных в дифференциальных выражениях. Приведем представление оператора (δ) Лапласа: $X := 1$ $Y := 1$ некоторой функции

$$\omega(X, Y) := X \cdot \ln(X) + Y \cdot \sin(Y)$$

$$\delta\omega := \frac{d}{dX} \frac{d}{dX} \omega(X, Y) + \frac{d}{dY} \frac{d}{dY} \omega(X, Y) \quad \delta\omega = 1.23913$$

в полярных координатах $\rho := \sqrt{2}$ $\varphi := 0.25 \cdot \pi$

$$X := \rho \cdot \cos(\varphi) \quad Y := \rho \cdot \sin(\varphi) \quad F(\rho, \varphi) := \omega(\rho \cdot \cos(\varphi), \rho \cdot \sin(\varphi))$$

$$\delta\omega := \frac{d}{d\rho} \frac{d}{d\varphi} F(\rho, \varphi) + \frac{1}{\rho^2} \cdot \frac{d}{d\varphi} \frac{d}{d\varphi} F(\rho, \varphi) + \frac{1}{\rho} \cdot \frac{d}{d\rho} F(\rho, \varphi).$$

Вычисленное значение $\delta\omega = 1.23913$ в полярных координатах совпадает со значением оператора функции в декартовых координатах.

16. Вычисление криволинейных интегралов. В качестве примера предлагается определить массу и центр тяжести циклоиды. Если масса гладкой кривой L распределена с линейной плотностью $\rho(X, Y, Z)$, то полная масса кривой вычисляется по формуле $M := \int_L \rho(X, Y, Z) dS$, а координаты центра тяжести есть:

$$A := \frac{1}{M} \cdot \int_L X \cdot \rho \, ds \quad B := \frac{1}{M} \int_L Y \cdot \rho \, ds \quad C := \frac{1}{M} \cdot \int_L Z \cdot \rho \, ds$$

Вычисляется масса и координаты центра тяжести циклоиды, заданной в полярной системе координат следующими уравнениями: $R := 3$ $X(t) := R \cdot (t - \sin(t))$ $Y(t) := R \cdot (t - \sin(t))$ при $(0 \leq t \leq 2 \cdot \pi)$ и с равномерно распределенной массой $\rho(X, Y) = 1$. Используя связь между криволинейными и обычными интегралами, получаем:

$$F(t) := \sqrt{\left[\frac{d}{dt} X(t)\right]^2 + \left[\frac{d}{dt} Y(t)\right]^2} \quad M := \int_0^{2 \cdot \pi} F(t) \, dt$$

$$A := \frac{1}{M} \cdot \int_0^{2 \cdot \pi} X(t) \cdot F(t) \, dt \quad B := \frac{1}{M} \cdot \int_0^{2 \cdot \pi} Y(t) \cdot F(t) \, dt \quad M = 24$$

$$A = 9.425 \quad B = 4 \quad t: = 0, 0.1 \dots 2 \cdot \pi$$

Рис. 41

На графике приведен один виток циклоиды и крестиком отмечен центр его тяжести.

17. Замена переменных в двойных интегралах. Нахождение площади S области, ограниченной астроидой:

$$A: = 1 \quad X: = A \cdot \cos(t)^3 \quad Y: = A \cdot \sin(t)^3 \quad (0 \leq t \leq 2 \cdot \pi)$$

с помощью криволинейных координат

$$X(u, v): = u \cdot \cos(v)^3 \quad Y(u, v): = u \cdot \sin(v)^3$$

и замены переменных в двойном интеграле

$$S(A): = \int_0^A \int_0^{2 \cdot \pi} \left| \begin{vmatrix} \frac{d}{du} X(u, v) & \frac{d}{dv} X(u, v) \\ \frac{d}{du} Y(u, v) & \frac{d}{dv} Y(u, v) \end{vmatrix} \right| dv du$$

$S(A) = 1.1781$. Аналогично осуществляется замена переменных и в интегралах кратности, большей двух.

18. Вычисление тройных интегралов. Нахождение объема эллипсоида, заданного уравнением

$$\frac{X^2}{A^2} + \frac{Y^2}{B^2} + \frac{Z^2}{C^2} = 1$$

$$V(A, B, C) = 8 \cdot \int_0^A \int_0^{\frac{B}{A} \cdot \sqrt{A^2 - X^2}} \int_0^{\sqrt{1 - \frac{X^2}{A^2} - \frac{Y^2}{B^2}}} 1 dZ dY dX$$

$$V(1, 2, 3) = 25.133$$

19. Вычисление поверхностных интегралов. Подобно случаю криволинейных интегралов поверхностные интегралы бывают также двух родов, каждый из которых сводится к обычному двойному интегралу. Если поверхность задана параметрически в виде:

$$X: = X(u, v) \quad Y: = Y(u, v) \quad Z: = Z(u, v)$$

и переменные u, v пробегают область R плоскости (u, v) , то

$$\iint_{(S)} T(X, Y, Z) dS: = \iint_{(R)} T(X(u, v), Y(u, v), Z(u, v)) \times \\ \times \sqrt{EG - F^2} du dv$$

где

$$A = 48 \quad X(u, v) = A \cdot \sin(u) \cdot \cos(v)$$

$$Y(u, v) = A \cdot \sin(u) \sin(v) \quad Z(u, v) = A \cdot \cos(u)$$

$$E(u, v): = \left[\frac{d}{du} X(u, v) \right]^2 + \left[\frac{d}{du} Y(u, v) \right]^2 + \left[\frac{d}{du} Z(u, v) \right]^2$$

$$G(u, v): = \left[\frac{d}{dv} X(u, v) \right]^2 + \left[\frac{d}{dv} Y(u, v) \right]^2 + \left[\frac{d}{dv} Z(u, v) \right]^2$$

$$F(u, v): = \frac{d}{du} X(u, v) \frac{d}{dv} X(u, v) + \frac{d}{du} Y(u, v) \frac{d}{dv} Y(u, v) + \\ + \frac{d}{du} Z(u, v) \frac{d}{dv} Z(u, v)$$

$$S: = 4 \cdot \int_0^{\frac{\pi}{2}} \int_0^{\pi} \sqrt{E(u, v) \cdot G(u, v) - F(u, v)^2} dudv \quad S = 5260.459$$

20. Вариационные задачи. Требуется приближенно найти экстремум функционала

$$J(Y) = \int_0^1 \left[\left[\frac{d}{dX} Y \right]^2 + Y^2 + 2 \cdot X \cdot Y \right] dX$$

при условии $Y(0) = 0$ и $Y(1) = 0$. Решение отыскивается на основе приближенного метода Рунге [26]:

Определяются три координатные функции U , V и Z

$$U(X) := X^2 - X \quad V(X) := X^3 - X^2 \quad Z(X) := X^4 - X^3$$

$$Y(A, B, C, X) := A \cdot U(X) + B \cdot V(X) + C \cdot Z(X) \quad \text{Эмпирическая функция}$$

Условия $Y(A, B, C, 0) = 0$ и $Y(A, B, C, 1) = 0$ удовлетворены

Исследуемый функционал при данных предположениях принимает вид

$$J(A, B, C) = \int_0^1 \left[Y(A, B, C, X)^2 + \left[\frac{d}{dX} Y(A, B, C, X) \right]^2 + 2 \cdot X \cdot Y(A, B, C, X) \right] dX$$

С помощью решающего блока отыскивается минимум функционала J по его параметрам A , B и C :

$A := 0 \quad B := 0 \quad C := 0 \quad \text{Given}$

$$\frac{d}{dA} J(A, B, C) \approx 0 \quad \frac{d}{dB} J(A, B, C) \approx 0 \quad \frac{d}{dC} J(A, B, C) \approx 0$$

$R := \text{FIND}(A, B, C) \quad n := 0 \dots 2$

Эмпирическая функция принимает следующий вид:

$$Y(X) := \sum_n R_n \cdot [X^{n+2} - X^{n+1}]$$

Табулирование эмпирической функции и точной функции, дающей минимум функционалу, на отрезке $[0, 1]$ дает следующие результаты:

$$X := 0, 0.2 \dots 1 \quad Y(X) \quad \frac{e}{e^2 - 1} \cdot [e^X - e^{-X}] - X$$

0.00002	0
-0.02859	-0.02868
-0.05023	-0.05048
-0.05821	-0.05826
-0.04421	-0.04429
0.00001	0

Точность полученных результатов вполне удовлетворительна. Учитывая тесную связь вариационных задач с задачами оптимального управления, пакет может быть использован и для численного решения задач данного класса.

21. Вывод эмпирических формул. Задачи данного типа часто возникают при обработке эмпирических числовых данных (результатов

наблюдений, измерений и т. д.). Для решения таких задач существует ряд методов, из которых рассмотрим в качестве примера метод наименьших квадратов, разработанный еще Гауссом. Рассматриваемая задача состоит в следующем. Упорядоченные в порядке возрастания экспериментальные данные находятся в двух структурированных файлах А (соответствует оси X) и В (соответствует оси Y), при этом между данными в обоих файлах существует некоторая функциональная связь, для которой требуется определить эмпирическую формулу. Краткий алгоритм решения задачи состоит в следующем:

A := READPRN(A) B := READPRN(B) Чтение файлов в
length (A) = 13 length (B) = 13 вектора A и B
TOL := 0.01 m := length (A) — 1 i := 1..m
j := B₁, 0.5..B_m

Рис. 42

Получение основной информации по файлам и подготовка к выводу на основе их данных графика зависимости.

В результате очевидных преобразований полученного графика появляется предположение о выборе кусочно-определенной эмпирической формулы, заданной на интервалах [A₁, A₃], [A₃, A₄], [A₄, A₅] и [A₈, A_m]. Для упрощения вычислений вводится функция ρ (X, A, B), определяющая принадлежность X полуинтервалу [A, B).

$$\rho(X, A, B) = \text{if } (\Phi(X - B), 0, \text{if } (\Phi(X - A), 1, 0))$$

где Φ — встроенная функция Хэвисайда.

Для получения эмпирической формулы из вида графика на первом, втором и последнем отрезках для сглаживания данных выбираются прямые линии, а на третьем отрезке кривая $Q(X) = GX^4 + H$. Для получения прямых линий в виде $Y = k_1X + k_2$ используем решающий блок, параметрами которого являются координаты точек концов отрезков прямых на графике, а выходом — вектор, содержащий соответственно значения для величин A и B искомой прямой.

$k_1 := 1 \quad k_2 := 1$
Given $k_1 \cdot X_1 + k_2 \approx Y_1 \quad k_1 \cdot X_2 + k_2 \approx Y_2$
 $K[X_1, Y_1, X_2, Y_2] := \text{FIND}[k_1, k_2] \quad W := K[A_1, B_1, A_3, B_3]$
 $V := K[A_3, B_3, A_4, B_4] \quad R := K[A_8, B_8, A_m, B_m]$

Таким образом, искомые прямые принимают следующий вид:

$$y(X) := W_0 \cdot X + W_1 \quad u(X) := V_0 \cdot X + V_1 \quad z(X) := R_0 \cdot X + R_1$$

Кривую для третьего отрезка ищем по методу наименьших квадратов: $p := 4$
 $n := 4..8$

$$Q(G, H, p, X) := G \cdot X^p + H \quad \omega(G, H) := \sum_n [Q(G, H, p, A_n) - B_n]^2$$

и используем для определения параметров G и H решающий блок при начальных условиях $G := 0 \quad H := 0$

Given

$$\frac{d}{dG} \omega(G, H) \approx 0 \quad \frac{d}{dH} \omega(G, H) \approx 0$$

$C := \text{FIND}(G, H)$

$$Q(X) := C_0 \cdot X^p + C_1$$

Теперь все входящие в эмпирическую формулу частные функции определены и формула принимает следующий вид:

$$Y(X) := \text{if} [\rho[X, A_1, A_3], y(X), \text{if} [\rho[X, A_3, A_4], u(X), \text{if} [\rho[X, A_4, A_8], Q(X), z(X)]]]$$

Для проверки качества сглаживания данных полученной эмпирической формулой вычисляем абсолютные значения разностей между данными файла В и полученными по формуле $Y(X)$ в одинаковых точках отрезка $[A_1, A_m]$, а также строим график эмпирической функции:

$$|Y[A_i] - B_i|$$

0
0.01
0
0.004
0.004
0.003
0.005
0
0.002
0.005
0.003
0

Рис. 43

Проведенная проверка показывает весьма хорошую степень сглаживания исходных экспериментальных данных эмпирической формулой.

22. Физические задачи. Баланс энергии при изменении поверхности жидкости. Требуется определить количество высвобождаемой энергии при слиянии мелких ртутных капель радиусом r в одну большую каплю радиусом R . Для решения данной задачи требуется определить некоторую систему единиц. При решении широкого круга физических задач рекомендуется помещать в текущий документ (в его начало или конец) один из имеющихся в составе пакета файлов систем единиц 'MKS.MCD' (СИ) или 'CGS.MCD', либо специальный пользовательский файл единиц измерений. Для наших же целей

вполне достаточно определить в документе только необходимые базовые и производные единицы:

$$\text{cm:} = 1\text{L} \quad \text{gm:} = 1\text{M} \quad \text{sec:} = 1\text{T} \quad \text{Базовые единицы}$$

$$g: = 980.665 \cdot \frac{\text{cm}}{\text{sec}^2} \quad \text{dyne:} = \text{gm} \cdot \frac{\text{cm}}{\text{sec}^2} \quad \text{erg:} = \text{dyne} \cdot \text{cm}$$

$$\text{mm:} = 0.1 \cdot \text{cm} \quad \text{joule:} = 10^7 \cdot \text{erg} \quad \text{Производные единицы}$$

Исходные данные: $g: = 2 \cdot 10^{-3} \cdot \text{mm}$ $R: = 2 \cdot \text{mm}$ Радиусы капель
 $\alpha: = 540 \cdot \frac{\text{dyne}}{\text{cm}}$ (Коэффициент поверхностного натяжения ртути
 при $t = 20^\circ \text{C}$)

Решение: Пусть N есть число мелких капель. Тогда общая поверхность $S(N)$ всех N мелких капель

$$S(N): = 4 \cdot \pi \cdot r^2 \cdot N$$

Так как поверхность большой капли есть $S_0: = 4 \cdot \pi \cdot R^2$, то выделяемая при слиянии мелких капель энергия вычисляется по формуле

$$E(\alpha, N): = 4 \cdot \pi \cdot [r^2 \cdot N - R^2] \cdot \alpha$$

Число N легко определяется из того соображения, что суммы объемов мелких капель и одной крупной должны быть равны

$$N: = \left[\frac{R}{r} \right]^3$$

Подставляя это значение в предыдущую формулу, получаем

$$Q: = 4 \cdot \pi \cdot R^2 \cdot \left[\frac{R}{r} - 1 \right] \cdot \alpha$$

Теперь легко получить значение выделяемой энергии

$$Q = 0.02712 \cdot \text{joule}$$

Таким образом, за счет уменьшения поверхности жидкости при слиянии мелких капель в одну большую освобождается энергия Q , которая идет на нагревание одной полученной большой капли.

23. Дифференциальные уравнения. В качестве примера рассмотрим простую задачу численного решения дифференциального уравнения первого порядка

$$\frac{d}{dX} Y(X) \approx F(X, Y) \quad Y(A) \approx B \quad \square$$

методом Эйлера. По методу Эйлера решение определяется рекуррентной формулой: $A: = 1$ $H: = 0.01$ $N: = 5$ $j: = 0..N$ $X_1: = A + H \cdot j$ $B: = 1$ $Y_0: = B$ $F(X, Y): = X \cdot Y^{1/3}$ $Y_{j+1}: = Y_j + H \times F[X_j, Y_j]$ (Конкретное ур-ние), позволяющей получить приближенное решение уравнения в узлах X_j . Для данного случая

оценим приближенное решение и точное в указанных узлах:

$$Y = \begin{bmatrix} 1 \\ 1.01 \\ 1.02 \\ 1.03 \\ 1.041 \\ 1.051 \\ 1.062 \end{bmatrix} \quad \begin{array}{|c|} \hline 1 \\ \hline 1.01 \\ \hline 1.02 \\ \hline 1.031 \\ \hline 1.041 \\ \hline 1.052 \\ \hline \end{array}$$

Использование более сильных методов позволяет получить вполне удовлетворительные результаты численного решения дифференциальных уравнений в среде пакета MathCAD.

24. Функции комплексного аргумента. Определение комплексной функции:

$F(Z) := A(X, Y) + i \cdot B(X, Y)$ $W := F(Z)$ $Z := X + i \cdot Y$

Значения независимой переменной Z ставятся в соответствие единственной точке (X, Y) комплексной Z -плоскости, а значения $W := A + i \cdot B$ таким же образом ставятся в соответствие точкам (A, B) W -плоскости. Из сказанного следует, что $A(X, Y) := \operatorname{Re}(F(Z))$ и $B(X, Y) := \operatorname{Im}(F(Z))$. Рассмотрим графическое представление

комплексной функции в W -плоскости:

$$P := -3 \quad H := 0.4 \quad i := 0 \dots -\left[\frac{2 \cdot P}{H}\right] \quad J := 0 \dots -\left[\frac{2 \cdot P}{H}\right] \quad X_i := P + H \cdot i \quad Y_j := P + H \cdot j \quad Z_{i,j} := X_i + i \cdot Y_j \quad F(Z) := \sin(Z) \cdot e^Z \\ \operatorname{Re}_{i,j} := \operatorname{Re}[F[Z_{i,j}]] \quad \operatorname{Im}_{i,j} := \operatorname{Im}[F[Z_{i,j}]]$$

Рис. 44

Графики представляют отдельно действительную и мнимую части заданной функции $F(Z)$ комплексного аргумента Z .

25. Вычисление эквивалентного уровня шума. Непостоянный шум оценивается в эквивалентных уровнях звука $L_{\text{экв}}$ в децибелах (дБ). Под эквивалентным уровнем звука понимается среднестатистический уровень звука непостоянного шума, оказывающего такое же воздействие на человека, как и постоянный шум того же уровня. Эквивалентный уровень звука рассчитывается согласно руководству [42], а входными данными для расчета являются набор замеренных уровней звука и общее время замера. Исходя из упомянутых выше особенностей пакета MathCAD, для решения поставленной задачи применяется следующая типичная технология. Простая БЕЙСИК-программа «VVD.BAS» организует ввод исходных данных (значения уровней звука и общее время замера), на основе которых формирует

файлы данных (VVOD.DAT — замеренные значения уровней звука; ND.DAT — количество полученных значений и TIME.DAT — общее время замера в секундах). Подготовленные программой ввода файлы используются соответствующим документом в среде пакета MathCAD. Таким образом, наиболее слабая сторона пакета (ввод данных) обеспечена БЕЙСИК-программой, тогда как вся расчетная часть задачи реализована средствами самого пакета. Документ расчетной части задачи имеет следующий вид:

[Н] $p := 0..17$ $L_{p,0} := 23 + 5 \cdot p$ $L_{p,1} := 27 + 5 \cdot p$
 L — (18×2) -матрица диапазонов уровней звука
 $T := \text{READ}(\text{TIME})$ $j := 0.. \text{READ}(\text{ND}) - 1$
 $V_j := \text{READ}(\text{VVOD})$ Чтение в массив V файла исходных данных

$$N_p := \sum_j \text{if } [L_{p,0} \leq V_j], 1, 0] \cdot \text{if } [L_{p,1} \geq V_j], 1, 0]$$

$$\delta_T := \frac{T}{\text{length}(V)} \quad \delta_p := N_p \cdot \delta_T \quad K_p := \delta_p \cdot \frac{100}{T}$$

$$L_{\text{экв}} = 10 \cdot \log \left[0.01 \cdot \sum_p K_p \cdot 10^{\frac{L_{p,0} + L_{p,1}}{20}} \right] L_{\text{экв}} = 81.01$$

[К] Значение эквивалентного уровня шума вычислено в дБ.

Уже из приведенных в главе примеров следует, что пакет представляет достаточно эффективные и выразительные средства для решения весьма широкого класса научно-технических задач из различных областей человеческой деятельности, которые можно сформулировать в классических математических понятиях. При этом, результаты не только представляются в обычном математическом написании, но и снабжаются текстовой, графической и другой иллюстративной информацией, что позволяет в удобном виде оформлять результаты численного исследования той или иной проблемы. Пользователю пакета MathCAD рекомендуется в процессе его эксплуатации создавать и расширять собственные библиотеки функций пользователя наряду с библиотеками документов. Функции примеров 7 и 8, определенные глобально, включены в систему MINIDOS / MathCAD в качестве соответственно файлов BOOL.MCD и USERLIB.MCD. Пользователю системы по мере необходимости командой 'Append' пакета можно дописывать в текущий документ требуемые MCD-файлы и использовать определенные в них функции и фрагменты стандартных документов.

ЗАКЛЮЧЕНИЕ

Компьютеризация многих направлений человеческой деятельности — это настоящее, но еще в большей мере — будущее развития человеческого общества в целом. Многоаспектность практического влияния компьютеризации на общество — науки, образования, производства, медицины, управления, вклад в решение многих глобальных проблем обуславливает значительные перспективы данного

направления, позволяя отнести его к наиболее приоритетным направлениям настоящего времени. С полной уверенностью можно сказать, что будущее развитие общества во многом будет зависеть от его информационного обеспечения, реализованного, в первую очередь, на основе новейших вычислительных средств и технологий. В этом отношении особая роль отводится классу персональных компьютеров, позволяющих вовлечь в процесс компьютеризации общества как профессиональных, так и непрофессиональных пользователей вычислительной техники (ВТ).

Одним из важнейших направлений компьютеризации является массовое освоение основ компьютерной грамотности, а также наиболее перспективного из программных средств различного назначения, созданных в развитых компьютерных странах за последние десятилетия. Данная проблема предполагает наличие различного уровня литературы, доступной широким кругам потенциальных пользователей ВТ. Это актуально не только в плане классического обеспечения специалистов технической литературой, но и в связи с весьма своеобразной ситуацией, имеющей место в нашей стране.

Вступив на путь широкой компьютеризации позже развитых компьютерных стран (США, Япония, Англия, ФРГ, Франция и др.), наша страна испытывает при решении данной проблемы целый ряд существенных затруднений: недостаток и низкое качество отечественной ВТ, отсутствие в нужном количестве специальной литературы, недостаточные уровень и методология подготовки массового пользователя, отсутствие серьезной национальной программы в этом направлении и т. д. Кроме этого, расширение международных связей открыло путь, по которому стали поступать в страну как современная персональная ВТ, так и многочисленные программные средства различного назначения (операционные системы, системы программирования, пакеты прикладных программ, системы автоматизации различного типа и т. д.). При этом поступление значительной части программных средств носит стихийный характер, а сами они плохо или совсем недокументированы. Все это вносит стихийность в выбор необходимых пользователю программных средств, учитывая отсутствие подготовки пользователей и специальной литературы.

В связи с этим задумана серия книг по программному обеспечению ПК, совместимых с хорошо известными системами IBM PC/XT/AT и PS/2. Первой из этой серии явилась книга [10], в которой с ориентацией на отечественный ПК ИСКРА 1030 были рассмотрены такие программные средства широкого назначения, как операционная система MS DOS, система программирования БЕЙСИК, текстовый процессор и дан обзор наиболее популярных зарубежных программных средств согласно обширной классификации. При создании данной серии книг предполагается поддерживать творческие отношения с наиболее известными зарубежными фирмами-разработчиками программных средств, ориентироваться на современные мировые тенденции в разработке и использовании ПК и их программного обеспечения, использовать многолетний опыт, полученный при разработке программного обеспечения для различного класса ЭВМ,

подготовке специалистов различного уровня по программированию, а также при издании книг по программному обеспечению ЭВМ серий ЕС и СМ.

Предлагаемая читателю книга является второй из указанной серии и посвящена хорошо известной за рубежом САД-системе MathCAD 2.52, ориентированной на автоматизацию решения различных вычислительных задач. Она рассчитана на тех читателей, которые не желают останавливаться на сведениях для начинающих, а хотят получить хорошее представление о возможностях пакета MathCAD и перспективных сферах его применения. С этой целью изложение построено таким образом, что читателю нет необходимости знакомиться с фирменной документацией по пакету, тем более, что материалы книги существенно перекрывают фирменную документацию, отражая особенности использования и выполнения пакета, а также его слабые и сильные стороны, не получившие отражения в фирменной документации и в изданных за рубежом поданному пакету книгах. Опыт апробации и эксплуатации пакета MathCAD показал его перспективность для многих пользователей ПК в различных сферах человеческой деятельности, связанных с решением задач, носящих ярко выраженный вычислительный характер. Наряду с этим настоящая книга может оказаться полезной также читателям, использующим в своей профессиональной деятельности подобные программные средства.

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ 1

ТИПЫ ПРИНТЕРОВ И ПЛОТТЕРОВ, ПОДДЕРЖИВАЕМЫХ ПАКЕТОМ MathCAD 2.52

В соответствии с идеологией пакета поддерживаемые им принтеры и плоттеры разбиты на две группы: **стандартные** и **дополнительные**. Драйверы для устройств первой группы находятся в файле MCAD.MCP, а второй группы в файле MORE.MCP. Оба файла с драйверами поставляются стандартно в составе пакета. По команде 'Selectprinter' из предлагаемого меню можно выбрать для вывода документов любой из стандартных драйверов. Во вторую группу включены драйверы, поддерживающие те же устройства, что и драйверы первой группы, но с более высокой разрешающей способностью или другими модификациями характеристик. Драйверы данной группы позволяют выводить документы лучшего качества, но существенно медленнее.

Вторая группа содержит также два чисто текстовых драйвера: один для работы с текстовыми файлами в коде ASCII и другой для расширенного кода IBM. Для использования дополнительных драйверов устройств можно использовать один из трех основных подходов:

- 1) любым текстовым редактором или системной командой COPY объединить файлы MCAD.MCP и MORE.MCP под именем первого файла;
- 2) любым текстовым редактором включить в файл MCAD.MCP из файла MORE.MCP необходимые драйверы;
- 3) переименовать файл MCAD.MCP в MORE.MCP и наоборот.

Первый подход делает стандартным любой драйвер из объединения файлов MCAD.MCP и MORE.MCP, но требует значительных затрат памяти ПК, так как каждый драйвер становится резидентным. Второй подход сокращает затраты памяти и делает стандартными только дополнительно требуемые драйверы. Наконец, при третьем подходе драйверы обоих системных файлов меняются по их

использованию пакетом в обратном порядке. На практике рекомендуется поступать следующим образом. При настройке пакета на конкретные ресурсы ПК в файл MCAD.MCP включать только драйверы реально существующих устройств вывода, помещая все остальные в файл MORE.MCP. Это поможет существенно сократить память ПК, занимаемую ядром пакета.

Перечень принтеров и плоттеров, имеющих

стандартные драйверы

PostScript: 10-pitch/12-pitch	IBM Gr/Epson Intl. (132-col)
Epson FX/JX/EX/LX (132-col)	Epson LQ: draft (132-col)
HP ThinkJet: native mode	HP ThinkJet: IBM mode
HP QuietJet: native 96-col	HP QuietJet: IBM mode
HP QuietJet + : native 158-col	HP QuietJet+: IBM mode
HP PaintJet	HP DeskJet
HP LaserJet: 75 dpi	HP LaserJet Ser II: 75 dpi
Toshiba P351: 96-col (158-col)	Okidata 84s2/92/93: 96-col
Okidata 84: w/IBM Plug & Play	Okidata 192+
Okidata 193 + : 136-col	C. Itoh. C-315: 132-col
NEC P5	NEC P6: 10-pitch (12-pitch)
JDL 850: 132-col (Плоттер)	Diconix 150: 85-col
TI 850/855	TI 865: 132-col
Dataproducts 8072 : 132-col	Fujitsu DL-2400
Gemini (132-col)	HP Plotter (HPGL)

дополнительные драйверы

Post Script : half-size	IBM/Ep Intl: dbl-dens. (132-col)
Epson FX : dbl-dens. (132-col)	Epson LQ: 180 doi (132-col)
HP ThinkJet: native hi-dens.	HP ThinkJet: IBM mode hi-d
HP QuietJet: native hi-qual	HP QuietJet: IBM mode hi-q
HP LaserJet: 150 dpi	HP LaserJet: 300 dpi
HP LsrJet Ser II: 150 dpi	HP LsrJet Ser II: 300 dpi
Okidata 192 + : hi-quality	Oki. 193+: hi-quality 136-col
TI 855: hi-quality	Diconix 150: IBM mode 66x86
Text: ASCII-code	Text: IBM extended code

В приведенном перечне устройств вывода, если не указано противного, драйвер предназначен для 80-колонного вывода, поэтому для принтеров с широкой кареткой следует выбирать соответствующие драйверы, помеченные как 132-col, 136-col или 158-col. Эти же данные в скобках говорят о дополнительной возможности соответствующего драйвера. Краткая характеристика каждого устройства дана после его названия через двоеточие и соответствует общепринятым сокращениям. Более полные технические характеристики перечисленных устройств можно найти в фирменной документации [22], а тексты самих драйверов легко получить из файлов MCAD.MCP и MORE.MCP с помощью любого текстового редактора или системными командами COPY или PRINT.

ПРИЛОЖЕНИЕ 2

ПРИМЕРЫ ДОКУМЕНТОВ, ПОСТАВЛЯЕМЫХ С ПАКЕТОМ MathCAD 2.52

В приложении приведены MCD-файлы, содержащие документы решения вычислительных задач, информационные, графические и т. д. Для использования любого из этих файлов можно загрузить его либо при запуске пакета MathCAD, либо по команде пакета 'Load'. Командами COPY или PRINT MS DOS, а также любым текстовым процессором можно вывести содержимое MCD-файлов на экран или печатать в MathCAD-формате. В перечне MCD-файлов (общим числом 34) приводятся только их главные имена, а общее для всех расширение имени '.MCD' опускается (табл. 16).

Имя файла	Назначения MathCAD-документа и данного файла
CGS	Система мер CGS: сантиметр — грамм — секунда. Все остальные физические единицы, включая электрические, определены в терминах этих единиц
MKS	Система мер (СИ): метр — килограмм — секунда — кулон
USCUSTOM	Система мер США: фут — фунт — секунда — кулон
ANHARM	Анализ колебаний негармонического осциллятора
BEAM	Расчет эластичного изгиба нагруженного бруса, опирающегося на оба своих конца
BOOLEAN	Определение функций логики в среде пакета
NPV	Вычисление текущего чистого дохода
OITRIG	Набросок рисунка, подготовленного в TurboCAD
SHUTTLE	Эскиз корабля многоразового использования типа «Columbia», подготовленного в AutoCAD
POLAR	Представление графиков в полярных координатах
QUADRATI	Решение общего квадратичного уравнения
SIMULT	Анализ распространения эпидемии
SPIRAL	Вычерчивание логарифмической спирали
STAIR	Демонстрация использования пакета для проектирования лестничных строительных конструкций
STARS	Вычерчивание вписанных в окружность звездообразных фигур при использовании в качестве координат комплексных чисел
SURFACE	Различные примеры вычерчивания поверхностей
TRIANGLE	Вычисление элементов плоского треугольника
DIRECTPY	Краткая информация по всем файлам пакета
README	Новейшая информация по пакету MathCAD 2.52., включая дополнения и корректировки к технической документации пакета
BRAKE	Анализ процесса торможения автомобиля
COMPFUNC	Вычерчивание графика функции комплексного аргумента; действительные и мнимые части представляются отдельно
CONFORM	Комформное отображение линейным преобразованием в комплексной плоскости
CONTOUR	Вычисление комплексного контурного интеграла
CURVEFIT	Полиномиальное сглаживание кривой; подход может использоваться и для других типов сглаживания, а также для множественной регрессии
DECAY	Вычерчивание кривых радиоактивного распада
DIFFUSE	Решение дифференциального уравнения диффузии методом конечных разностей
FFTFILT	Обработка сигналов методом быстрого дискретного преобразования Фурье
FLUID	Определение размера трубы для турбулентного потока жидкости при заданных условиях
FRACTAL	Анализ самоквадрируемых кривых в комплексной плоскости, инвариантных относительно квадратичного преобразования
HEATTRAN	Расчет передачи тепла через двухслойную стену
HISTO	Гистограмма результатов подбрасывания монеты
LISSAJOU	Вычерчивание плоских фигур Лиссажу
MORTGAGE	Анализ состояния кредитной задолженности
MOTION	Расчет баллистической траектории снаряда

Следует отметить, что MCD-файлы CGS и MKS можно включать в текущие документы и использовать входящие в них преобразования единиц измерений для вычислений физико-технического характера.

ОСНОВНЫЕ ОСОБЕННОСТИ РАБОТЫ В СРЕДЕ ПАКЕТА MathCAD 2.52, ВЫЯВЛЕННЫЕ В ПРОЦЕССЕ ЕГО ЭКСПЛУАТАЦИИ И АПРОБАЦИИ

В приложении кратко описаны особенности использования пакета, а также перспективы дальнейшего его развития, сформулированные в процессе апробации и эксплуатации при решении широкого класса задач вычислительного характера. Наряду с этим в приложении представлены ранее не упоминавшиеся возможности и особенности пакета, что является вполне естественным при создании подобного типа сложных программных систем. На протяжении книги представляемые особенности обсуждались с различной степенью детализации, но их сведение в приложении дает возможность пользователю легче ориентироваться в процессе освоения пакета MathCAD 2.52.

1. При записи на диск по команде 'ConfigSave' командного файла не выдается диагностика при отсутствии на томе свободного места. На том записывается указанный в команде файл с нулевой длиной. В случае записи файла на неготовое или неисправное УВВ происходит длительная попытка записи, но также не выдается никакой диагностики и пакет возвращается в состояние документа. Отсутствие данной диагностики требует в общем случае проверки успешности записи файла на том.

2. Если в команде MCAD указан параметр загрузки ' $\backslash R \langle CF \rangle$ ', то при возникновении особой ситуации (отсутствие файла, пустой файл, дефектность УВВ) диагностика не выдается и при наличии используется именно пакетный файл MCAD.MCC.

3. В командном режиме пакет не для всех команд допускает использование комментария. Так команда 'Execute : \langle Комментарий \rangle ' игнорируется, а по команде 'Dos : \langle Комментарий \rangle ' делается попытка чтения несуществующего файла. В командном же файле пакета (MCC-файле) комментарий можно помещать только отдельной строкой, начинающейся либо точкой с запятой (;), либо каким-нибудь другим символом или словом, не являющимся командой пакета. Во втором случае при выполнении КФП с таким комментарием выдается сообщение 'Undefined command', в ответ на которое следует для продолжения выполнения пакета нажать клавишу 'Enter'. В результате неверная команда игнорируется, выполнение КФП продолжается. Любая информация, стоящая в КФП после команды 'Execute', игнорируется.

4. При ожидании выбора устройства печати по команде 'Select' невозможно выйти из пакета по клавишам 'Ctrl + Q'.

5. Символ бесконечности (∞), определяемый значением 10^{307} и декларируемый как наибольшее число в среде пакета, на самом деле таковым не является. На самом деле в качестве максимального пакет допускает значение $2,2 \times 10^{307}$. Сам символ бесконечности не соответствует принятому в математике, о чем свидетельствуют следующие соотношения, имеющие место в среде пакета: $\infty / \infty = 1$, $\infty / 5 = 2 \times 10^{306}$, $5 \times \infty =$ Переполнение. Поэтому обращаться с данным символом следует осмотрительно. Перед использованием данной предопределенной переменной в документе рекомендуется переопределить ее как $\infty := 2.2 \cdot 10^{307}$.

6. Пакет допускает использование комплексных чисел вида $A + Bi$, где числа A и B могут быть любого из трех типов: действительные с плавающей точкой, а также целые 8- или 16-ричные. Формат результатов вычислений с участием таких чисел определяется по умолчанию (D-формат) или задается командой 'Format'.

7. В конструкциях типа $0 * X$ или $0/X$ выражение X не вычисляется, но проверяется на наличие ошибок, что не соответствует описанию ситуации, данному в фирменных руководствах [19, 20].

8. По команде 'Equation' пакета можно выключать в документе как любые выражения, графики и выходные таблицы, так и определения переменных или функций пользователя, т. е. любую вычислительную область документа.

9. Пакет формирует диагностическое сообщение 'Unable to run Dos' не только при попытке выполнения команды 'Dos' без доступа к системному файлу COMMAND.COM, но и при нехватке оперативной памяти ПК, остающейся от текущего документа.

10. Пакет идентифицирует аварийную ситуацию 'MEMORY' при наличии

свободной памяти ПК для текущего документа в пределах 7К, а не 10К, как указано в фирменном руководстве [19].

11. В случае глобального и локального определения одной и той же переменной или функции пользователя для каждого их применения в документе используется ближайшее сверху определение (локальное или глобальное), в противном случае — последнее глобальное определение.

12. По команде 'Equation' можно выключать любой вычисляемый объект (определение, выражение, график и т. д.). При этом выключаемые выходные таблицы или графики соответствующим образом помечаются, исключаются из процесса вычислений, но из документа не удаляются. При загрузке документа, содержащего такие объекты, в них сохраняется только информация, внесенная пользователем, а не полученная в результате вычислений.

13. Сообщение 'Illegal ORIGIN' возникает в случае присвоения предопределенной переменной ORIGIN значения большего, чем 15099, а не 25 000 как указано в документации [21].

14. Сообщение 'Index too large' возникает при попытке использовать верхний или нижний индекс со значением большим, чем $ORIGIN + 7999$ (предельным для пакета). В документации [21] указан только частный случай при $ORIGIN = 0$.

15. При определении p -переменной $X := \langle AB1 \rangle, [\langle AB2 \rangle] .. \langle AB3 \rangle$ при наличии параметра $\langle AB2 \rangle$ шаг ее изменения определяется по формуле $\Delta = [(AB3) - (AB1)] \times [(AB2) - (AB1)] / | (AB3) - (AB1) |$ при соблюдении условия $(AB1) < (AB2) \leq (AB3)$.

16. При определении p -переменной посредством входной таблицы элементами ее могут быть любые допустимые выражения, не вычисляемые в самой таблице. Однако в векторном представлении данной таблицы в качестве элементов вектора будут уже вычисленные значения этих выражений.

17. При корректировке таблицы (если ее изменения касаются элементов с номерами в диапазоне определенного ранее p -индекса) производится актуализация связанных с нею вычислений. Однако увеличение размера таблицы сверх максимума ее p -индекса не влечет за собой такой актуализации. Во избежание возможных ошибок при таких корректировках следует одновременно корректировать также и диапазон изменения p -индекса таблицы.

18. Пакет допускает использование массивов размерности не большей двух.

19. При работе с большими массивами, p -выражениями и другими сложными объектами пакет оказывается довольно медленным, поэтому весьма желательно наличие его Турбо-версии или высокопроизводительных ПК.

20. Ряд аспектов редактирования выражений не полностью соответствует фирменным материалам [19], читателю рекомендуется подробнее ознакомиться с разделом 2.7 книги и апробировать изложенные сведения в практической работе на ПК.

21. При обработке выражений пакет не приводит подобные члены, например $Y := X + X + X$ не преобразуется в эквивалентное $Y := 3 \cdot X$. Данная особенность пакета имеет свои отрицательные и положительные стороны.

22. Описание применения оператора продолжения выражения (клавиши 'Ctrl + Enter') в документации [19] не полностью охватывает ряд его аспектов. Для подробного ознакомления с данным оператором рекомендуется обратиться к разделу 2.7.

23. Редактирование текста в области или зоне: по клавишам 'Ctrl + L' курсор перемещается в конец текущей строки; после центровки пустой строки курсор вместе с идентифицирующим ее символом (◀) устанавливается в центр строки; после центровки текста в строке курсор устанавливается в ее конец.

24. Для групп команд 'Copy', 'Cut', 'Paste', 'Incory', 'Incut', 'Inpaste' пакет использует различные типы ВПП.

25. В результате расширения текстовая зона на экране перекрывает следующие за ней области и другие зоны. После вывода курсора из зоны или по клавишам 'Ctrl + N' пакет раздвигает документ, оставляя свободное место (количество пустых строк не фиксировано) между зоной и следующими за ней областями и зонами.

26. В отличие от областей по клавишам 'Ctrl + V' текстовая зона выделяется вертикальной линией в ее нулевом столбце.

27. Типы ВП П для команд 'Incopy', 'Incut', 'Inpaste' (группа 'In-region') относительно их объектов обработки (помеченных блоков и выражений) различны, тогда как ВПП для команд 'Cory', 'Cut', 'Paste' (отличаясь от ВПП команд группы 'In-region') идентична относительно обрабатываемых ими объектов (области и зоны). Содержимое всех типов ВПП сохраняется при очистке памяти от текущего документа по команде 'Clear', что существенно повышает гибкость использования команд копирования объектов документа и может служить своеобразным интерфейсом между различными документами в среде пакета.

28. Диагностика пакета [21] не является полной, поэтому рекомендуется обращаться к прил. 5 данной книги.

29. При сохранении в дисковом файле документа, содержащего выключенные графические области и выходные таблицы, с последующей загрузкой его в память такие объекты после загрузки имеют следующий вид: графическая область, кроме самой кривой графика, сохраняет все свои элементы; от выходной таблицы сохраняется только ее идентификатор со знаком равенства.

30. При попытке сохранения MCD-файла или PRN-файла текущего документа на диске с недостатком свободного места в первом случае выдается неточная диагностика 'Error writing file', а во втором пакет «зависает» без выдачи какой-либо диагностики и требуется перезагрузка системы по клавишам 'Ctrl + Alt + Del' или кнопке 'System'. В обоих аварийных ситуациях на диске создается пустой файл нулевой длины с указанным именем.

31. Вывод текущего документа на принтер или в дисковый файл можно прервать по клавишам 'Ctrl + A' или 'Ctrl + Break'. Во втором случае принтер после прекращения печати прогоняет бумагу.

32. При всей наглядности, допускаемой пакетом при работе с производными и интегралами (а в ряде случаев и с ранжированными суммированием и произведением), вычисления данных конструкций требуют весьма существенных временных затрат.

33. К недостаткам пакета следует отнести также отсутствие в его среде доступной пользователю службы времени, основанной на работе таймера ПК.

34. В качестве переменной дифференцирования можно использовать p -переменную. Результатом дифференцирования является выходная таблица, с помощью которой можно строить график производной функции.

35. При вычислении определенного интеграла в качестве пределов интегрирования, а также переменных и констант в подынтегральном выражении можно использовать p -переменные. Результатом интегрирования является выходная таблица, с помощью которой можно строить график первообразной функции.

36. Для различных массивов документа можно использовать разные значения предопределенной переменной ORIGIN, область действия которой зависит от типа определения (локального или глобального) подобно обычным переменным. Однако в этом случае следует быть внимательным, применяя один и тот же массив в различных областях определений переменной ORIGIN.

37. Определения всех переменных решающего блока, включая и его ведущие переменные, следует вводить глобально в любом месте документа или локально выше данного блока. В противном случае пакет полагает их для блока неопределенными и идентифицирует соответствующие ведущие переменные функции MINERR или FIND диагностическим сообщением 'Undefined'.

38. При использовании операции векторизации для функции пользователя, определенной на основе функции FIND или MINERR, переменная ERR пакета не определяется и появление ее приводит к ошибке с сообщением 'Undefined'.

39. Тело решающего блока может содержать ранжированные, векторные и матричные выражения (если они вычисляемы в нем), текстовые области и зоны, а также графические области.

40. Функции FIND и MINERR конца решающего блока нельзя использовать непосредственно в выражениях и в качестве аргумента других функций. В этом плане они несколько отличны от других функций, поддерживаемых пакетом. Например, конструкции вида $\ln(\text{find}(x))$ или $\cos(x) + \text{minerr}(x, y)$ недопустимы. В отличие от FIND использование функции MINERR в подобных конструкциях не всегда вызывает аварийные ситуации, хотя результат в общем случае некорректен.

41. Для функции пользователя, заданной в терминах функций FIND или MINERR конца решающего блока, не допускается глобальное определение.

В противном случае первое использование такой функции приводит к аварийной ситуации с диагностикой 'Illegal context'.

42. Последним возвращаемым значением встроенной функции until (expr1, expr2) является значение выражения 'expr2', вычисленное в момент первого отрицательного значения ранжированного выражения 'expr1'.

43. Применение встроенных функций доступа READ и READPRN к существующим файлам производит их полное обновление без запроса санкции пользователя на проведение такой операции. По функциям доступа APPEND и APPENDPRN дописывается информация в конец существующего файла данных или создается новый файл соответствующего типа также без запроса санкции пользователя.

44. Использование функций доступа READ и READPRN требует наличия у имени файла его расширения. По умолчанию предполагается расширение имени соответственно '.DAT' и '.PRN'. При отсутствии у файла данных расширения имени он является недоступным для встроенных функций доступа пакета, вызывая особую ситуацию 'File not found'.

45. Команда 'Filename' пакета применима как к существующим, так и к новым файлам данных любого типа. С ее помощью с каждым файлом данных можно ассоциировать в документе ряд переменных, но не наоборот. Максимальное количество файлов, используемых текущим документом, не должно превышать 15.

46. В качестве разделителей записей в неструктурированном файле данных допускаются пробелы, запятые, точки с запятой, символы табуляции и перевода строки. Для структурированного файла данных в качестве заполнителей свободных мест шаблона допускаются пробелы, запятые и точки с запятой.

47. Использование для функций доступа глобальных определений требует глобального определения и всех связанных с ними переменных и выражений, включая и предопределенные переменные пакета, если они переопределяются в текущем документе.

48. При копировании выражений в ВПП и обратно в документ (команды группы 'In-region') операции векторизации могут не сохраняться, если они не были указаны при копировании в качестве старших. Это обстоятельство следует учитывать при редактировании выражений в документе.

49. В качестве правой части функций доступа WRITE и APPEND допускаются ранжированные, векторные, матричные и индексированные действительные или комплексные выражения. В качестве правой части функций доступа WRITEPRN и APPENDPRN допускаются наряду с векторными и матричными также ранжированные и индексированные действительные и комплексные выражения. Особенности такого использования рассматриваются в гл. 7.

50. Функцию доступа READ можно использовать со структурированными данными, возвращая корректные результаты, тогда как применение функции доступа READPRN к неструктурированному файлу данных возвращает в основном некорректный результат, имеющий единичное нулевое значение.

51. Функции доступа READ и READPRN в ряде случаев можно использовать в выражениях подобно другим функциям пакета, однако в общем случае такие операции некорректны.

52. Использование в команде 'Goto' управления курсором отрицательного значения для параметра строки (столбца) приводит к установке курсора в нулевую строку (столбец) текущего документа.

53. Команда 'Replace' замены цепочки символов имеет формат 'Replace [Текст]', где параметр 'Текст' задает заменяемую цепочку символов. Цепочка для замены определяется в ответ на запрос 'Replace with:' после ввода команды. Приведенное в фирменных руководствах описание команды пакета 'Replace' не совсем корректно.

54. Пакет допускает работу с документами только в двухоконном режиме экрана. Определяя для дополнительного окна экрана А-режим, все создаваемые впоследствии дополнительные окна экрана также будут формироваться пакетом в А-режиме.

55. Сфера действия команд пакета: Format, Plotformat, Surfaceformat, Sketchformat, Set, Redraw, Selectprinter, Memory, Dimensions и Randomize распространяется на оба окна экрана.

56. В результате разметки документа по страницам посредством команд 'Pagelength' и 'Pagebreak' пакет выводит размеченный документ на печать постранично, но остановка после каждой страницы не производится. Такая организация вывода на печать существенно затрудняет использование бланковой печати документов.

57. В качестве функциональной можно использовать и комплексную матрицу Z , но в этом случае поверхностный график строится для функциональной матрицы $M: = \operatorname{Re}(Z)$, а если матрица Z мнимая — для функциональной матрицы $M: = \operatorname{Re}(Z - \operatorname{Re}(Z))$, т. е. нулевой матрицы.

58. Вместо активного указателя области поверхностного графика (ОПГ) можно помещать не только имя функциональной матрицы, но и матричное выражение с этой или другой предварительно определенной матрицей, результатом которого также должна быть матрица. Данная возможность позволяет деформировать функциональную матрицу в самой ОПГ, что в ряде случаев существенно упрощает определение в документе исследуемой функции от двух переменных.

59. При форматизации рисунка по команде 'Sketchformat' координаты его нового центра вычисляются по формулам $N(1 - Y)$ и $M(1 - X)$, начиная с левого верхнего угла области рисунка, где N и M — соответственно размеры области по вертикали и по горизонтали в строках и столбцах документа; Y и X — значения параметров характеристики 'Center' формата рисунка.

60. По команде 'Equation' можно выключать из процесса вычислений в документе любую вычислительную область (кроме текстовой), области плоского и поверхностного графиков, а также область рисунка. Включение этих областей в процесс вычислений происходит после ввода в них курсора и повторной команды 'Equation'.

61. При автомасштабировании области плоского графика следует быть внимательным, так как при округлении предельных значений диапазонов по осям координат возможно существенное искажение характера поведения графика (его расположение относительно осей координат). По возможности рекомендуется прямое задание диапазонов по осям координат.

62. При создании нескольких плоских графиков в одной графической области количество элементов в списках выражений для осей Y и X , определяющих пары для выводимых графиков, не обязательно должно быть одинаковым.

63. В определении p -переменной вида $j: = A, B \dots C$ должно соблюдаться соотношение $(C - A)/(B - A) \leq 10^9$, в противном случае возникает аварийная ситуация с диагностикой 'Illegal range'.

64. В функции 'if' пакета в общем случае использование функций доступа к данным любого типа некорректно.

65. Операции дифференцирования и интегрирования не векторизуются по клавишам (Alt + «—»).

66. При возникновении особой ситуации при использовании функции пользователя, определенной в терминах решающего блока, в p -выражении не выводятся вычисленные до данной ситуации значения, что существенно ограничивает применение указанного подхода при исследовании, например, задач линейного и нелинейного программирования.

67. В одной области поверхностного графика допускается определение только одной поверхности, что в ряде случаев является существенным ограничением.

68. Недопустимость использования оператора (\neq) неравенства в решающих блоках пакета является относительной. Данный оператор нельзя применять в качестве соединительного для левой и правой частей условия, но можно, например, использовать с функцией 'if' пакета в любой из частей условия.

69. В случае недостатка свободного места на диске при выводе текущего документа в PRN-файл (файл в формате принтера) возможно «зависание» пакета, требующее его перезагрузки.

70. Для выключения (команда 'Equation') решающего блока необходимо выключить в нем по крайней мере выражение, содержащее функцию FIND или MINERR конца блока. Выключение только заголовка 'GIVEN' не исключает блок из процесса вычислений в текущем документе.

71. В ранжированном выражении допускается использование условных операторов ранжирования, определяющих те значения p -переменной, по которым производится ранжирование. Для операции ранжированного произведения

условные операторы ранжирования не имеют особого смысла, так как почти всегда дают нулевой результат.

72. Использование в операции ранжированного суммирования массива **M** без индексов приводит в результате к массиву **M1** того же типа с элементами, представляющими собой значения соответствующих элементов массива **M**, умноженных на $[rows(M) \times cols(M)]$.

73. При выводе HELP-информации по пакету (клавиша 'F1') некорректно определена возможность 'S-Boolean', предназначенная для характеристики возможностей работы в среде пакета с булевой алгеброй. Стандартно такая возможность пакетом не поддерживается и для ее обеспечения создан специальный файл **BOOL.MCD** (см. гл. 10 и прил. 13).

74. Если в текущем документе какой-либо массив определен глобально, то для изменения начального значения его индексов переменную **ORIGIN** пакета также следует определять глобально. В противном случае возможно несоответствие диапазонов изменения индексов массивов в документе.

75. При выгрузке документа по команде 'Save' в дисковый **MCD**-файл пакет не диагностирует аварийные ситуации (отсутствие свободного места на томе, неисправность УВВ и т. д.), что требует повышенного внимания при проведении данной операции.

ПРИЛОЖЕНИЕ 4.

КЛАВИАТУРА ПК, СОВМЕСТИМЫХ С IBM PC/XT/AT

В качестве примера международной клавиатуры компьютеров, совместимых с **IBM PC/XT/AT**, приводится 101-клавишный пульт ПК **IBM PC/AT** (рис. 45). Другие возможные реализации клавиатур, имея тот же основной набор клавиш, могут различаться исполнением. По своему функциональному назначению все клавиши можно разделить на пять основных групп:

- 1) функциональные клавиши (ФК);
- 2) алфавитно-цифровая клавиатура (АЦК);
- 3) специальные клавиши (СК);
- 4) клавиши управления курсором (КУК);
- 5) цифровая клавиатура (ЦК) и КУК.

Кратко рассмотрим состав и назначение клавиш в каждой из этих групп.

Группа 1 (ФК) содержит 12 функциональных клавиш **F1 — F12** и одну специальную клавишу «ESC». Назначение этих клавиш определяется той программной средой, под управлением которой в данный момент функционирует ПК. Например, в **MS DOS** клавиша «ESC» отменяет вводимую команду, а в среде **MathCAD 2.52** служит для переключения из состояния команд в состояние документа и наоборот, а также для отмены ранее введенного запроса на обслуживание. Назначение клавиш этой группы в среде пакета **MathCAD** подробно рассматривается в настоящей книге.

Группа 2 (АЦК) является основной частью клавиатуры и содержит алфавитно-цифровые клавиши (подобные клавишам обычной пишущей машинки) и ряд специальных клавиш: «CapsLock», «BackSpace (Bksp)», «Ctrl», «Alt», «Tab» и «Shift». АЦК позволяет вводить символы латинского алфавита, символы национальных алфавитов (включая кириллицу), десятичные цифры 0—9 и ряд специальных символов (#, %, &, !, ? и др.). Специальные клавиши АЦК имеют следующее основное назначение:

CapsLock — для переключения регистров прописных и строчных букв;

Tab — клавиша табуляции;

Ctrl — совместно с другими клавишами для запроса специальных функций; назначение может определяться программной средой;

Alt — совместно с другими клавишами АЦК и ЦК запрашивает специальные функции, которые могут определяться программной средой;

Shift — для переключения верхнего и нижнего регистров клавиатуры;

BackSpace — сдвиг курсора на позицию влево; может при этом стирать встречные символы;

Enter — завершение ввода информации (конец текста).

Подробно использование клавиш данной группы в среде пакета **MathCAD 2.52** рассматривается в настоящей книге.

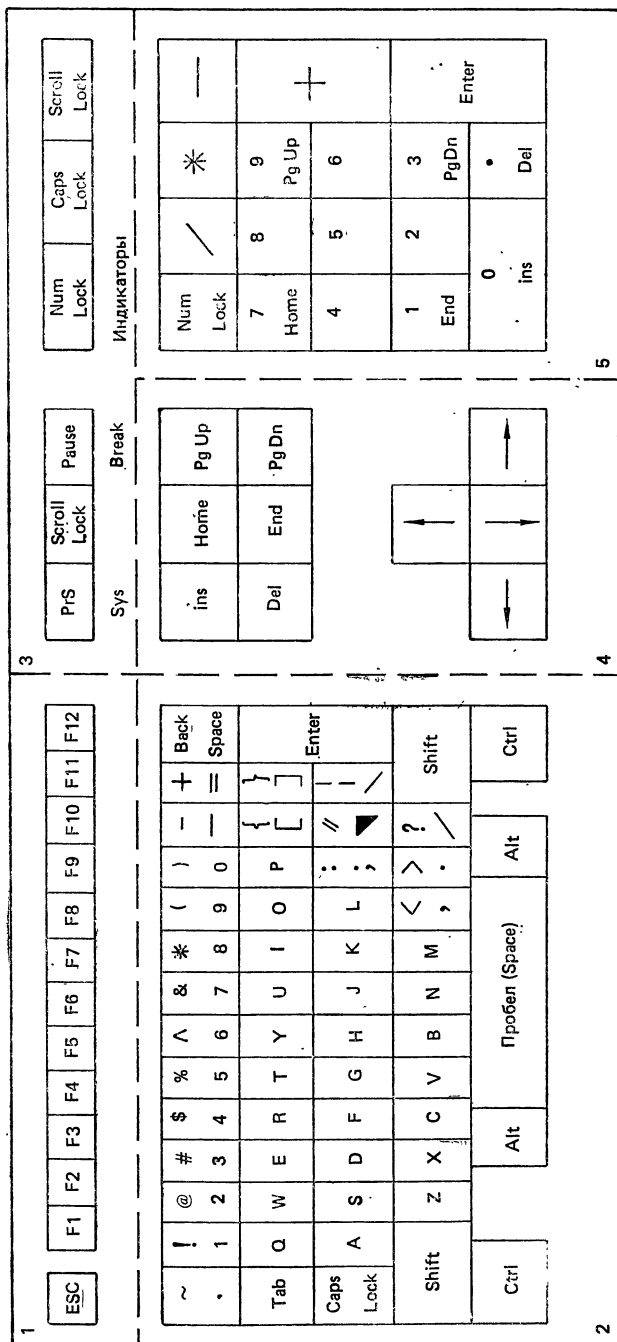


Рис. 45. Принципиальная структура 101-клавишного пульта международного стандарта для ПК IBM/PC/XT/AT

Группа 3 (СК) содержит специальные управляющие клавиши, имеющие следующее основное назначение:

Sys — системная клавиша; используется согласно описанию MS DOS или в зависимости от программной среды;

PrtS — на верхнем регистре выводит на печать копию экрана, а на нижнем — регистр символ (*);

ScrollLock — перемещает текст на экране на одну строку вниз или вверх.

К этой же группе относятся световые индикаторы управляющих клавиш «NumLock», «CapsLock» и «ScrollLock».

Группа 4 (КУК) содержит клавиши управления курсором и по назначению полностью перекрывается возможностями клавиш пятой группы. Использование клавиш данной группы имеет смысл, когда клавиши группы 5 находятся в числовом режиме (по клавише «NumLock»).

Наконец, *группа 5* (ЦК) содержит цифровую клавиатуру и КУК, режим работы которой (цифровой или управления курсором) определяется управляющей клавишей «NumLock». В числовом режиме данная группа клавиш используется как обычная ЦК для ввода десятичных цифр 0—9 и знаков арифметических операций {., | + | — | * | /}, тогда как в режиме управления курсором — следующим образом:

Home — перемещение курсора в начало экрана;

End — перемещение курсора в правый конец строки символов на экране;

PgUp — переход к предыдущей странице экрана;

PgDn — переход к следующей странице экрана;

Del — удаление символа в позиции курсора;

Ins — перевод в режим «вставки», когда каждый ввод нового символа раздвигает строку текста.

Назначение остальных КУК легко следует из нанесенных на них стрелок, указывающих направление перемещения курсора на одну позицию экрана.

Одновременное использование группы клавиш позволяет в зависимости от программной среды выполнять специальные функции, например:

Ctrl + Alt + Del — перезагрузка операционной системы на ПК;

Ctrl + Break — прерывание выполнения команды или программы;

Ctrl + Alt + F1 — переключение клавиатуры на US-стандарт;

Ctrl + Alt + F2 — переключение клавиатуры на национальный алфавит.

Подробное описание клавиатуры (которая не соответствует международному стандарту) отечественного ПК ИСКРА 1030, совместимого с серией IBM PC/XT, можно найти в книге [10].

ПРИЛОЖЕНИЕ 5

ПЕРЕЧЕНЬ ОСОБЫХ И АВАРИЙНЫХ СИТУАЦИЙ, ИДЕНТИФИЦИРУЕМЫХ ПАКЕТОМ MathCAD 2.52

Приводится полный перечень идентифицируемых основных ошибок и особых ситуаций, возникающих в процессе создания, вычислений и обработки документов в среде пакета. По мере необходимости приводятся поясняющие примеры и основные рекомендации по локализации и устранению ошибок. Обнаруженная ошибка в использовании функции пользователя связывается с именем функции, а не с ее определением, поэтому данная ситуация требует анализа определения функции пользователя. Предлагаемая диагностика апробирована и уточнена в процессе эксплуатации пакета MathCAD 2.52.

1. **Array size mismatch.** Несоответствие размеров массивов в векторной или матричной операции. Многие операции этого типа требуют одинаковых размеров участвующих в них операндов или их компонент. Например, операция умножения двух матриц требует одинакового количества столбцов первой матрицы и строк второй матрицы.

2. **Cannot be defined.** Локальное или глобальное определение недопустимой конструкции, например, $X^2 := X$. Допустимые конструкции, определяемые в среде пакета, приведены в гл. 2.

3. **Cannot take subscript.** Использование нижнего индекса для конструкции, не являющейся векторной или матричной.

4. **Cannot take superscript.** Использование верхнего индекса для конструкции, не являющейся матричной.

5. **Definition stack overflow.** Использовано слишком много вложенных определений функции.

6. **Did not find solution.** Невозможно решить блок при заданных условиях. Для оценки результатов решения блока следует в его конце использовать вместо функции 'Find' функцию 'Minegr'.

7. **Dimension to nonreal power.** Возведение выражения с размерностью в комплексную степень. Единицы измерений могут возводиться только в действительную скалярную степень, иначе пакет не может определить размерность результата.

8. **Domain error.** Попытка вычисления функции с недопустимым значением аргумента, например, $\ln(0)$.

9. **Duplicate.** Попытка повторного определения переменной в одном и том же определении.

10. **Equation too large.** Попытка ввести слишком большое выражение. Рекомендуется выражение представить в виде нескольких меньших.

11. **Error in constant.** Недопустимое представление константы, например, $X := 525B$.

12. **Error in list.** Недопустимый список аргументов при определении или использовании функции, например, $F(X, 48) := X$.

13. **Error in solve block.** Возникает при выполнении функции пользователя, определенной в терминах решающего блока, и при наличии ошибки в таком блоке. Для локализации ошибки следует проверить соответствующий решающий блок.

14. **File error.** Ошибка при попытке чтения из файла данных посредством функции READ или READPRN. Возможно неверное использование форматов данных, например, READ(Command.com).

15. **File not found.** Пакет не может найти посредством функции READ или READPRN файл данных. Рекомендуется проверить его наличие путем вывода содержимого каталогов используемых томов. У искомого файла данных возможно отсутствует расширение имени.

16. **Illegal array operation.** Попытка применить скалярную функцию или операцию к вектору или матрице. Например, деление двух матриц или $X := : = X^2$, где X — p -переменная.

17. **Illegal context.** Использование функции или операции в недопустимом пакете контексте. Например, использование имени функции в качестве имени переменной или наоборот $[r(r) := r \cdot X]$.

18. **Illegal factor.** Недопустимое выражение для указателя в конце вычисляемой конструкции. Указатель в конце вычисляемой конструкции требует действительного скалярного выражения. Это сообщение часто появляется совместно с сообщением 46.

19. **Illegal function name.** Пакет интерпретирует конструкцию как функцию, но имя ее недопустимо, например, $e(5)$ или $\pi(x)$.

20. **Illegal ORIGIN.** Предопределенной переменной ORIGIN присвоено нецелое или большее 15 099 число. Ошибка идентифицируется только при первом использовании какого-либо элемента массива.

21. **Illegal range.** Использование недопустимого определения ранжированной переменной, например, $P := 1,1..10$.

22. **Illegal tolerance.** Возникает при вычислении выражений, содержащих конструкции (интеграл, производная и т. д.), которые требуют использования предопределенной переменной 'TOL', не удовлетворяющей условию $0 < \text{TOL} < 1$. Для локализации и устранения ошибки следует изменить значение 'TOL' на допустимое.

23. **Incompatible units.** Неверные операции над размерными величинами, например, сложение времени и массы. Проверить структуру выражения, использование единиц и системы измерений.

24. **Indeterminate dimension.** Попытка возведения размерного выражения в степень, содержащую ранжированную переменную или вектор. Пакет в этом случае не может определить размерность результата. Реорганизовать выражение так, чтобы степень была действительным числом, либо исключить из него единицы измерений.

25. **Index out of bounds.** Верхний или нижний индекс получает значение, лежащее вне допустимого диапазона, например, $X_{-2} = 5$ при $ORIGIN = 0$.
26. **Index too large.** Попытка использовать нижний или верхний индекс со значением большим, чем $ORIGIN + 7999$ (предельным для пакета).
27. **Interrupted.** Информировует о том, что вычисления в документе были прерваны по клавишам 'Ctrl + A' или 'Ctrl + Break'. Для пересчета прерванного выражения следует установить в него курсор и нажать клавишу 'F9' (Вычисление).
28. **List too long.** Попытка ввести слишком много элементов списка, разделенных запятыми.
29. **Misplaced comma.** Некорректное использование запятой, которая может кодироваться только в списке аргументов функции, а также для разделения ранжированных переменных, элементов списка по осям графика, элементов входной таблицы и нижних индексов в матричных переменных. В остальных случаях использование запятой полагается пакетом некорректным.
30. **Missing operand.** При вводе выражения пропущен операнд, место которого идентифицируется указателем. Например, $X/' = '$.
31. **Missing operator.** В выражении отсутствует знак операции, например, $(X + 5)/(Y - 7) =$.
32. **Must be 3-vector.** Использование операции векторного произведения для выражений, не являющихся трехэлементными векторами.
33. **Must be array.** Использование операции, требующей массива, а не скаляра. Например, попытка определить переменную с верхним индексом как скаляр.
34. **Must be dimensionless.** Некорректное использование единиц измерений, например, $\sin(1M)$.
35. **Must be increasing.** Использование вектора (чьи элементы не упорядочены в порядке возрастания) в качестве аргумента функции, требующей строгого упорядочения.
36. **Must be integer.** Выражение принимает нецелые значения, тогда как требуются целые. Например, для значений нижних и верхних индексов.
37. **Must be nonzero.** Использование недопустимого нулевого значения для аргумента встроенной функции, например, для функции Бесселя $Y0(0)$.
38. **Must be positive.** Попытка использования для вывода графика нулевых или отрицательных значений для логарифмической оси, допускающей только положительные ненулевые значения.
39. **Must be range.** Вместо требуемой ранжированной переменной использована другая конструкция, например, в качестве нижнего индекса.
40. **Must be real.** Вместо требуемого действительного значения использовано комплексное, например, для значений индексов.
41. **Must be scalar.** Вместо требуемого скалярного выражения использовано векторное или матричное. Например, в качестве аргумента тригонометрических функций.
42. **Must be square.** Недопустимое использование неквадратной матрицы, например, в операциях обращения или возведения в степень.
43. **Must be vector.** Использование скаляра или матрицы вместо требуемого вектора, например, в качестве нижнего индекса — матрица.
44. **Nested solve block.** Используются вложенные решающие блоки, недопустимые пакетом. Однако пакет допускает использование внутри блока функций пользователя, определенных в терминах решающих блоков.
45. **No matching Given.** Несовместимое использование 'Given' с функцией 'Find' или 'Minerr'. Каждый решающий блок, оканчивающийся функцией 'Find' или 'Minerr', должен начинаться управляющим словом 'Given'.
46. **Non-scalar value.** Вместо требуемого скаляра использовано векторное или ранжированное выражение. Например, $X = K$, где K — ранжированная переменная.
47. **Not a name.** Вместо требуемого имени использована недопустимая комбинация символов, например, $\text{root}(X^2 + X + 1, 48)$.
48. **Not converging.** При определенном значении предопределенной переменной 'TOL' пакет не в состоянии вычислить интеграл, производную и другие выражения.
49. **Overflow.** При вычислении получено значение, превышающее максимальное для данного пакета $2,2 \times 10^{307}$.

50. **Significance {lost | reduced}**. Попытка вычислить функцию для значения аргумента, выходящего из диапазона точности для данной функции, например, $\cos(10^{48})$.

51. **Singularity**. Вычисление выражения, функции или выполнение операции с недопустимым значением. Например, обращение сингулярной матрицы (ее определитель равен нулю) или деление на ноль.

52. **Stack overflow**. Переполнение внутреннего стека пакета при вычислении выражения. Рекомендуется упростить выражение, разбить его на несколько частей или изменить алгоритм.

53. **Too few arguments**. Использование функции с меньшим (чем указано в ее определении) количеством аргументов. Для встроенных функций оно фиксировано, а для пользовательских — задается при определении в документе. Пример ошибочного использования функции `root(X)`.

54. **Too few constraints**. В решающем блоке использовано меньше условий, чем требуется для вычисления значений всех ведущих переменных. Рекомендуется добавить в блок фиктивные условия или сократить количество ведущих переменных.

55. **Too few elements**. Попытка применить дискретное преобразование Фурье, кубическую сплайн-интерполяцию или функцию линейной интерполяции к вектору с небольшим количеством элементов. Так, для преобразования Фурье вектор должен иметь не менее 4 элементов, а для двух остальных — не менее двух элементов.

56. **Too few subscripts**. Используется один индекс для идентификации элемента матрицы вместо необходимых двух.

57. **Too large to display**. Попытка вывести на экран вектор или матрицу размера большего, чем допускает пакет.

58. **Too many arguments**. Использование функции с большим (чем указано в ее определении) количеством аргументов, например, `sin(5, 8)`.

59. **Too many constraints**. В решающем блоке находится более 50 условий.

60. **Too many subscripts**. Использование более одного индекса для идентификации элемента вектора.

61. **Too many points**. Попытка начертить точек больше, чем пакет может обработать для одного графика.

62. **Undefined**. Идентифицирует неопределенную переменную или функцию любого типа. Указанную конструкцию следует определить в документе локально или глобально. При анализе подобной ситуации следует помнить, что она может быть следствием ошибки, возникшей в одной из предыдущих конструкций.

63. **Underflow**. При вычислении получено значение, меньшее числа $1,8 \times 10^{-307}$, например, « $Q := 10^{307} \cdot 1.7/Q =$ ».

64. **Unmatched parenthesis**. В выражении количество левых и правых скобок неодинаково. Добавить недостающие или убрать лишние скобки.

65. **Wrong size vector**. Неверное количество элементов аргумента в функциях преобразования Фурье. Функция 'ifft' требует в качестве аргумента вектор с $2^n + 1$ ($n > 0$) элементами, а три другие — вектор с 2^n ($n > 1$) элементами, где n — целое положительное число.

66. **Range not allowed**. Недопустимое использование ранжированных переменной или выражения. Например, при попытке в качестве границы диапазона графической области указать p -переменную.

67. **Error writting file**. Ошибка записи в файл. Например, при попытке сохранить MCD-файл на дисковом томе, на котором недостаточно свободного места.

68. **Unable to open**. Пакет не может открыть указанный файл. Например, введена команда «Save ✕.✕».

69. **Can't find file MCAD.OVL**. Резидентная часть пакета не может найти оверлейный файл. Для продолжения работы следует указать к нему полный путь или нажать клавишу 'Enter', если файл установлен на активном устройстве.

70. **Too many files**. Количество используемых документом файлов данных превышает допустимое пакетом (15). Например, по команде 'Filename' пакета для встроенных функций доступа определено более 15 различных файлов данных.

71. **Error writting file**. Ошибка при записи информации в дисковый файл. Как правило, данная ситуация возникает при нехватке места на диске при по-

пытке создания файла данных (по функциям доступа WRITE, APPEND, WRITEPRN, APPENDPRN) или при попытке сохранить на диске текущий документ (по команде 'Save').

72. **Illegal command.** Недопустимое использование команды. Например, попытка удалить в документе командой 'Deleteline' непустую строку документа.

73. **No file loaded.** Отсутствие загруженного документа. Данная ситуация возникает при попытке использовать команду 'Append' к пустому документу.

74. **Printing ABORTED.** Прекращен вывод документа на печать или в дисковый PRN-файл. Данное сообщение появляется, например, после нажатия клавиш 'Ctrl + Break', 'Ctrl + A', либо после отмены команды 'Print' пакета по клавише 'ESC'.

75. **Unrecognized command.** Пакет не распознает введенную команду. Причина, как правило, лежит в неверном кодировании имени команды, либо недопустимом его сокращении. Следует повторить ввод нужной команды.

76. **Too close to a region.** Недостаточно места для области. Для устранения ситуации курсор следует переместить в свободную достаточную по размеру часть документа и повторить процедуру создания требуемой области.

77. **Not enough memory.** Не хватает оперативной памяти. Например, при динамическом формировании массива количество его элементов, не достигнув максимальной величины для пакета (8000), использовало всю доступную для текущего документа память ПК.

78. **Ambiguous command.** Введенная команда нераспознаваема, например, в случае недопустимого сокращения ее имени. Допустимые сокращения имен команд пакета приведены в прил. 10.

79. **Printer not ready.** Принтер не готов к работе. Проверить и включить принтер, после чего повторить команду 'Print'.

80. **[F9] To recalc plot.** Требуется произвести (по клавише 'F9' или командами 'Calculate', 'Process' перевычисление области плоского графика. Например, после введения или отмены логарифмической системы координат в области графика.

81. **Out of MEMORY.** Превышение текущим документом объема доступной памяти. Предоставляется возможность сохранить текущий документ по функциональной клавише 'F6' и/или выйти из пакета по клавишам 'Ctrl + Q'. Другие действия пакетом игнорируются.

82. **Unable to open MCAD.HLP.** Пакет не находит в активном каталоге свой HELP-файл. Необходимо данный файл либо сделать доступным пакету, либо игнорировать запрос HELP-функции по клавише 'Enter'.

83. **Invalid.** Недопустимая запись конструкции. Например, в ранжированную конструкцию неверно введен условный оператор ($n \neq j$) · [APPENDPRN(R): = M⁽ⁿ⁾].

84. **Not on placeholder.** Попытка поместить конструкцию по клавишам 'Ctrl + F4' (команда 'Inpaste') не по месту указателя выражения.

ПРИЛОЖЕНИЕ 6

КЛАВИШИ ПК, УПРАВЛЯЮЩИЕ ДВИЖЕНИЕМ КУРСОРА В ПРЕДЕЛАХ ТЕКУЩЕГО ДОКУМЕНТА ПАКЕТА

Приводятся клавиши, управляющие движением обычного и дополняющего курсоров, а также результат их применения. При описании использованы введенные ранее понятия, определения и соглашения. Области математических конструкций, текста и графиков обозначены соответственно «А», «В» и «С» (табл. 17).

ПРИЛОЖЕНИЕ 7

ВСТРОЕННЫЕ ФУНКЦИИ ПАКЕТА MathCAD 2.52

Приводится полный перечень встроенных функций пакета, сгруппированных в 13 групп. Кодирование имен функций является буквозависимым и соответствует приводимому написанию. При описании встроенных функций будут использованы следующие обозначения: X, Y; Z; N, N1, N2, N3; V, VX, VY, VS

Клавиши	Результат применения клавиш
←, ↓, ↑, →	Сдвиг курсора на одну строку или столбец в заданном клавишей направлении (цифровая клавиатура)
Tab	В «В» перемещение на слово вперед; в «А» и «С» — к следующему указателю; между областями движение вправо с шагом 10 позиций
Shift + Tab	В «В» перемещение на слово назад; в «А» и «С» — перемещение к предыдущему указателю; между областями движения влево с шагом 10 позиций
PgUp	Сдвиг вверх на 5 строк за одно нажатие
PgDn	Сдвиг вниз на 5 строк за одно нажатие
Ctrl + PgUp	Сдвиг вверх на 80 % страницы за одно нажатие
Ctrl + PgDn	Сдвиг вниз на 80 % страницы за одно нажатие
Ctrl + "←"	Сдвиг влево на 10 позиций (обратная табуляция)
Ctrl + "→"	Сдвиг вправо на 10 позиций (прямая табуляция)
Ctrl + E	Сдвиг влево на 80 позиций за одно нажатие
Ctrl + G	Сдвиг вправо на 80 позиций за одно нажатие
Home	Сдвиг в начало текущей области; если уже в начале, то сдвиг к предыдущей области документа
End	Сдвиг в конец текущей области; если уже в конце, то сдвиг к следующей области документа
Ctrl + Home	Выход на начало документа; перемещение курсора в начало первой области документа
Ctrl + End	Выход на конец документа; перемещение курсора на начало последней области документа
Enter	В «В» переход на новую строку; в «А» и «С» выход курсора на следующую строку за область; между областями сдвиг в начало следующей строки

и М — соответственно действительные, комплексные (или действительные), целочисленные, векторные и матричные выражения; exrg — выражение; var — имя переменной; fname — имя файла или ассоциированной с ним переменной (табл. 18).

Встроенные функции групп 1—5 и 11, а также ряд функций из групп 10 и 13 имеют широко известный математический смысл и особого пояснения не требуют. Функции групп 6; 7, 8, 10, 13; 9 и 12 рассмотрены соответственно в главах 7; 6; 4 и 5.

ПРИЛОЖЕНИЕ 8

ОСНОВНЫЕ ОПЕРАЦИИ, ПОДДЕРЖИВАЕМЫЕ ПАКЕТОМ MathCAD 2.52

Приводится полный перечень основных операций, поддерживаемых пакетом, с представлением их изображения, способа кодирования и краткой характеристики. Операции приводятся в порядке убывания приоритета. Более специальные операции, например, векторные и матричные, рассматриваются в соответствующих главах книги (табл. 19).

ПРИЛОЖЕНИЕ 9

ОСНОВНЫЕ УПРАВЛЯЮЩИЕ И ДОПОЛНИТЕЛЬНЫЕ КЛАВИШИ

ПАКЕТА MathCAD 2.52

Приведен перечень клавиш, предназначенных для создания областей и зон в документе, их форматирования, а также управляющих работой пакета в основных режимах (табл. 20). В перечень не включены клавиши, управляющие пере-

Функция пакета	Краткая характеристика функции
1. Тригонометрические функции (аргумент и результат в радианах)	
sin (Z)	Синус
cos (Z)	Косинус
tan (Z)	Тангенс
asin (Z)	Арксинус
acos (Z)	Аркосинус
atan (Z)	Арктангенс
angle (X, Y)	Угол в радианах (0—2 π) от оси X до точки с координатами (X, Y)
2. Гиперболические функции	
sinh (Z)	Синус
cosh (Z)	Косинус
tanh (Z)	Тангенс
asinh (Z)	Арксинус
acosh (Z)	Аркосинус
atanh (Z)	Арктангенс
3. Логарифмические и экспоненциальные функции	
exp (Z)	Экспонента e^Z
ln (Z)	Натуральный логарифм (по основанию e)
log (Z)	Десятичный логарифм
4. Функции Бесселя	
J0 (X)	Функция $J_0(X)$
J1 (X)	Функция $J_1(X)$
...	...
Jn (n, X)	Функция $J_n(X)$; $n > 0$
Y0 (X)	Функция $Y_0(X)$; $X > 0$
Y1 (X)	Функция $Y_1(X)$; $X > 0$
...	...
Yn (n, X)	Функция $Y_n(X)$; $X, n > 0$
5. Функции комплексного аргумента	
Re (Z)	Действительная часть переменной Z
Im (Z)	Мнимая часть переменной Z
arg (Z)	Аргумент Z: значение в радианах угла ($-\pi < \varphi \leq \pi$); $Z = A + Bi$, $\varphi = \arccos (A/ Z)$
6. Функции доступа к файлу данных с именем «fname»	
READ (fname)	Чтение очередного значения из файла
WRITE (fname)	Запись очередного значения в файл
APPEND (fname)	Запись значения в конец файла
READPRN (fname)	Чтение матрицы из структурированного файла (Ст-файл)
WRITEPRN (fname)	Запись матрицы в Ст-файл
APPENDPRN (fname)	Запись матрицы в конец Ст-файла

Функция пакета	Краткая характеристика функции
7. Функции интерполяции	
linterp (VX, VY, X)	Линейно интерполированное значение X на основе векторов VX и VY
cspline (VX, VY)	Вектор коэффициентов кубической сплайн-интерполяции на основе векторов данных VX и VY
lspline (VX, VY)	Вектор коэффициентов линейной сплайн-интерполяции на основе векторов данных VX и VY
pspline (VX, VY)	Вектор коэффициентов параболической сплайн-интерполяции на основе векторов данных VX и VY
interp (VS, VX, VY, X)	Интерполированное значение в точке X на основе кубической сплайн-интерполяции с векторами коэффициентов VS и векторами данных VX и VY
8. Функции быстрого дискретного преобразования Фурье	
fft (V)	Быстрое преобразование Фурье вектора V из 2^n действительных значений
ifft (V)	Обратное fft (V) преобразование; количество элементов V должно быть $2^n + 1$
cfft (V)	Быстрое преобразование Фурье вектора V из 2^n комплексных значений
icfft (V)	Обратное cfft (V) преобразование; количество элементов V должно быть 2^n
9. Функции векторного (V)-и матричного аргументов (M)	
length (V)	Число элементов вектора V
last (V)	Значение индекса последнего элемента V
max ({V M})	Максимальный элемент {вектора матрицы}
min ({V M})	Минимальный элемент {вектора матрицы}
rows (M)	Число строк матрицы M
cols (M)	Число столбцов матрицы M
identity (N)	Единичная (N×N)-матрица
tr (M)	След квадратной матрицы M; сумма элементов ее главной диагонали
augment (M1, M2)	Конкатенация матриц M1 и M2
augment (V1, V2)	Конкатенация векторов V1 и V2; возможна конкатенация матрицы и вектора
10. Статистические функции	
mean (V)	Средняя значений элементов вектора V
stdev (V)	Среднеквадратичное отклонение значений элементов вектора V
var (V)	Дисперсия значений элементов вектора V
corr (VX, VY)	Корреляция по Пирсону векторов VX и VY
slope (VX, VY)	Наклон линии регрессии для векторов значений VX и VY
intercept (VX, VY)	Отрезок, отсекаемый на оси Y линией регрессии для векторов VX и VY
Г (Z)	Гамма-функция Эйлера
erf (X)	Функция ошибок
cnorm (X)	Нормальная функция распределения
hist (V1, V2)	Гистограмма вектора значений V2 на основе интервалов, представленных V1
rnd (X)	Вычисление псевдослучайного числа на интервале [0,X]

Функция пакета	Краткая характеристика функции
11. Функции упорядочивания элементов в массивах	
<code>sort (V)</code>	Сортировка элементов вектора V в порядке возрастания (ПВ)
<code>{c r} sort (M, N)</code>	Сортировка (строк столбцов) матрицы M в ПВ ее (столбца строки) с номером N
<code>reverse ({M V})</code>	{Матрица вектор} с обратным порядком {строк матрицы M элементов вектора V }; <code>reverse (sort(V))</code> — сортировка элементов вектора V в порядке убывания

12. Функции решения уравнений и блоков

<code>root (expr, var)</code>	Нулевой корень var_0 уравнения $expr=0$
<code>FIND ((<i>список</i>))</code>	Возвращает результат решения блока
<code>MINERR ((<i>список</i>))</code>	Возвращает значения ведущих переменных, минимизирующих вектор отклонений

13. Другие встроенные функции пакета

<code>floor (X)</code>	Наибольшее целое, не превышающее X
<code>ceil (X)</code>	Наименьшее целое, не меньшее X
<code>if (cond, Z1, Z2)</code>	Значение $Z2$ при $cond=0$, иначе — $Z1$
<code>Ф (X)</code>	Ступенчатая функция Хэвисайда: значение 1 при $X \geq 0$ и 0 в противном случае
<code>mod (X1, X2)</code>	Остаток от деления $X1$ на $X2$; знак результата совпадает со знаком $X1$
<code>δ (N1, N2)</code>	Дельта-функция Кронекера: значение 1 при $N1=N2$ и 0 в противном случае
<code>ε (N1, N2, N3)</code>	Полностью антисимметричный тензор ранга 3: $N1, N2, N3$ — целые в интервале $0-2$ или в интервале $ORIGIN-ORIGIN+2$ при $ORIGIN=0$; возвращает 0, если любые два аргумента совпадают, возвращает 1 для четных перестановок и «-1» для нечетных
<code>until (X1, X2)</code>	Возвращает $X2$ до тех пор, пока $X1 \geq 0$; затем прекращает итеративный процесс

Изображение	Кодирование	Назначение операции
(X)	'X или (X)	Операция взятия скобок
X_i	X [i	Операция нижнего индекса
$M^{(k)}$	X'Alt+ ^'k	k-й столбец матрицы M
V	V'Alt+ "—" 'k	Векторизация выражения V
N!	N!	Операция факториала
M^T	M'Alt+!'	Операция транспозиции
X^Y	X^Y	Степень; обращение матрицы $Y=-1$
$-X$	-X	Операция отрицания знака
$\sum V$	'Alt+ \$ 'V	Сумма векторных элементов
\sqrt{X}	\ X	Извлечение квадратного корня
$ X $	X	Модуль, определитель, норма
$\frac{X}{Y}$	X/Y	Операция деления
$X \cdot Y$	X * Y	Скалярное произведение
$X \times Y$	X'Alt+ * 'Y	Векторное произведение
$\sum_i X$	i \$ X	Ранжированная сумма элементов
$\prod_i X$	i # X	Ранжированное произведение
$\int_a^b F(X) dX$	X & F(X)	Определенный интеграл
$\frac{d}{dX} F(X)$	X ? F	Производная функции F(X)
$X + Y$	X + Y	Операция сложения
$X - Y$	X - Y	Операция вычитания
$X \dots + Y$	X'Ctrl'Y	Операция продолжения с линейным прерыванием выражения
$X > Y$	X > Y	X больше, чем Y
$X < Y$	X < Y	X меньше, чем Y
$X \geq Y$	X'Alt+) 'Y	X не меньше, чем Y
$X \leq Y$	X'Alt+ ('Y	X не больше, чем Y
$X \neq Y$	X'Alt+ # 'Y	X не равно Y
$X \approx Y$	X'Alt+ " = " 'Y	X реляционно равно Y
$X, Y \dots Z$	X, Y; Z	Операция ранжирования

мещением курсора и служащие для просмотра документа. Эти клавиши представлены в прил. 6. Не представлены здесь и клавиши ввода команд пакета, которые приведены в прил. 10.

В круглых скобках приведены числа, набираемые на цифровой части (ЦК) клавиатуры и представляющие собой порядковые номера соответствующих символов в кодовой таблице ПК.

ПРИЛОЖЕНИЕ 10

СВОДКА КОМАНД ПАКЕТА MathCAD 2.52

Приводится полная сводка 61 команды пакета в принятой в настоящей книге нотации с их кратким описанием. Полное описание команд можно найти в соответствующих разделах книги. В первом столбце (табл. 21) приведено полное имя команды и максимально допустимая его аббревиатура. Команды по имени вводятся после перевода пакета клавишей 'ESC' в командный режим. Во втором

Клавиши	Назначение клавиш
Ctrl + Break	Прекращение операции вычисления или вывода
Ctrl + {A a}	Прекращение операции вычисления или вывода
Ctrl + {D d}	Изменение цвета дисплея типа EGA или VGA
Ctrl + {J j}	Эквивалентно клавишам 'Ctrl+Enter'
Ctrl + {U u}	Ввод символа параграфа (§)
Ctrl + {T t}	Создание в документе текстовой зоны
Ctrl + {V v}	Визуализация областей и зон документа
Ctrl + {G g}	Очистка экрана монитора
Ctrl + {H h}	Перемещение курсора на позицию влево
Ctrl + {I i}	Эквивалентно клавиша 'ESC'
Ctrl + {H h}	Эквивалентно клавише табуляции 'Tab'
Ctrl + {M m}	Перемещение курсора на строку вниз
" (кавычки)	Создание текстовой области в документе
@ (эт)	Создание области плоского графика
{F f}	Изменение локального формата области любого типа; курсор устанавливается в область
Alt + %	Создание в документе области рисунка
Alt + @	Создание области поверхностного графика
Alt + F1	Вывод в ССП максимально доступного размера памяти для текущего документа (ТД)
Alt + F2	Вывод в ССП размера используемой ТД памяти
Alt + F3	Вывод в ССП распределения памяти под ТД
Alt + (15)	Эквивалентно команде пакета 'Print'
Alt + (17)	Эквивалентно команде пакета 'Quit'
Alt + (18)	Эквивалентно команде пакета 'Redraw'
Alt + (20)	Эквивалентно команде пакета 'Text'
Alt + (22)	Эквивалентно клавишам 'Ctrl+V'
Alt + (27)	Эквивалентно действию клавиши 'ESC'

столбце даны клавиши, по которым вводится команда из состояния документа. В третьем столбце приведены две буквы, по которым предоставляется возможность ввести команду через командное меню, которое вызывается функциональной клавишей 'F10'. Например, команду 'Load' загрузки документа в память можно ввести любым из следующих способов:

перевести пакет клавишей 'ESC' в командный режим и в ССП ввести имя 'Load' с необходимым параметром или без;

нажать функциональную клавишу 'F5' и ввести на запрос пакета в ССП нужный ответ;

нажать функциональную клавишу 'F10', а затем подряд две клавиши 'F' и 'L'; пакет выведет командное меню с высвеченной командой; после нажатия клавиши 'Enter' в ССП появится запрос, требующий соответствующего ответа.

Как видно из приведенной сводки, не все команды пакета допускают ввод всеми возможными способами. Наконец, в четвертом столбце дана краткая характеристика команд пакета.

В описании команд пакета приняты следующие сокращения: ТД — текущий документ, НТД — непустой ТД, ВПП — внутренняя память пакета, ДОЭ — дополнительное окно экрана, а под «текстом» понимается любая текстовая область или зона.

Имя команды	Клавиши	Меню	Характеристика команды
Ap [pend]	—	F A	Дописывание файла в конец НТД
Au [tomatic]	—	C A	Перевод пакета в А-режим
Back [ward]	Ctrl+B	T B	Установление обратного порядка перемещений в тексте
Cal [culate]	F9	C C	Вычисление видимой части ТД
Cen [ter]	—	T C	Центровка строк текста
Cl [ear]	—	F C	Удаление ТД из памяти
Con [figsave]	—	S C	Сохранение текущей конфигурации пакета в файле MCAD.MCC
Cop [y]	F2	E C	Копирование области или зоны из ТД в ВПП
Cut	F3	E C	То же, но с удалением области или зоны из ТД
Del [eteline]	Ctrl+F10	E D	Удаление пустой строки ТД
Dim [ensions]	—	C D	Изменение имен размерности
Dos	—	S D	Выполнение команды MS DOS
Equ [ation]	—	C E	Выключение областей из вычислений (кроме текстовых)
Exe [cute]	—	S E	Выполнение файла MCAD.MCC
Fil [ename]	—	F F	Задание ассоциированной переменной для файла данных
Form [at]	—	C F	Установка глобального формата результатов вычислений
Forw[ard]	Ctrl+F	T F	Установление прямого порядка перемещений в тексте
Go[to]	—	E G	Установка курсора в заданные строку и столбец ТД
H[elp]	F1	S H	Help-информация по пакету
Inco[py]	Ctrl+F2	I C	Копирование части выражения или текста из ТД в ВПП
Incu[t]	Ctrl+F3	I C	То же, но с удалением из ТД
Inp[aste]	Ctrl+F4	I P	Копирование из ВПП в ТД
Ins[ertline]	Ctrl+F9	E I	Вставка пустой строки в ТД
Just[ify]	Ctrl+N	T J	Выравнивание текста
Line[length]	—	W L	Установка длины строки ТД
Lo[ad]	F5	F L	Загрузка документа из файла
Man[ual]	—	C M	Перевод пакета в М-режим
Marg[in]	—	W M	Установка левого поля для вывода ТД на принтер
Mark	Ctrl+Y	E M	Выделение блока областей ТД
Mat[rrix]	Alt+M	C G	Создание вектора или матрицы
Me[mory]	—	S M	Мониторинг объема памяти ПК
Mo[ve]	—	E M	Сдвиг ТД на заданное число строк и столбцов
Pageb[reak]	—	W I	Установка жесткой линии раздела страниц в ТД
Pagsel[ength]	—	W P	Установка длины страницы ТД
Pas[te]	F4	E P	Копирование из ВПП в ТД
Plo[tformat]	—	—	Установка глобального формата для плоских графиков
Pri[nt]	Ctrl+O	S P	Вывод ТД на печать или в файл
Pro[cess]	—	C P	Вычисление всего ТД
Q[uit]	Ctrl+Q	S Q	Завершение пакета и выход в операционную систему MS DOS
Ra[ndomize]	—	C R	Установка начального значения для генератора псевдослучайных чисел
Red[raw]	Ctrl+R	S R	Восстановление экрана

Имя команды	Клавиши	Меню	Характеристика команды
Rep[lace]	Ctrl+F6	E R	Замена одной части текста ТД на другую
Res[et]	—	F R	Перезагрузка пакета
Sa[ve]	F6	F S	Сохранение ТД в файле
Sea[rch]	Ctrl+F5	E F	Поиск заданного текста в ТД
Sel[ectprinter]		S S	Выбор принтера или плоттера
Sep[arate]		E S	Разделение пересекающихся областей и зон документа
Set	—	—	Задание новых значений для четырех переменных пакета
Ske[tchformat]	—	—	Установка глобального формата для рисунков ТД
Skip[l ine]	Ctrl+L	—	Переход к предыдущей / следующей строке текста
Skipp[aragraph]	Ctrl+P	—	Переход к предыдущему / следующему абзацу текста
Skips[entence]	Ctrl+S	—	Переход к предыдущему / следующему предложению текста
Skipw[ard]	Ctrl+W	—	Переход к предыдущему / следующему слову текста
Sp[lit]	F7	W S	Создание дополнительного окна экрана (ДОЭ)
Su[rfaceformat]	—	—	Установка глобального формата поверхностных графиков ТД
Sw[itch]	F8	W J	Переключение активного окна экрана
Textb[and]	Ctrl+T	T T	Создание текстовой зоны в ТД
Textm[ark]	Ctrl+X	T M	Выделение части текста
U[nsplit]	Ctrl+F7	W U	Отмена ДОЭ
W[idth]		T W	Установка длины текста

ПРИЛОЖЕНИЕ 11

АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ ДЛЯ РЕШЕНИЯ КОНСТРУКТОРСКИХ И ТЕХНОЛОГИЧЕСКИХ ЗАДАЧ

Предлагается краткая характеристика ряда оригинальных автоматизированных систем для эффективного решения различных конструкторских и технологических задач. Функционирование данных систем осуществляется на ПК семейств IBM PC/XT/AT и PS/2 и совместимых с ними компьютерах под управлением MS DOS версий 2.0 и выше. При этом наличие сопроцессора необязательно, но позволяет существенно повысить производительность функционирующих систем.

Система <Screen MUMPS>

является дальнейшим развитием универсального языка программирования MUMPS, знакомого пользователям ОС ДИАМС для СМ ЭВМ, и предоставляет следующие дополнительные возможности:

- универсальный Turbo-интерфейс для создания и отладки программ;
- поддержка и ведение библиотек программ;
- наличие справочной системы для различных режимов работы, включая информацию по самому языку MUMPS;
- встроенный оконный редактор;
- поддержка прямого и последовательного способов доступа к файлам;
- командный процессор для загрузки и выполнения программ.

Система <Screen MUMPS> соответствует стандарту ANSI MUMPS и может использоваться в качестве СУБД для создания информационно-поисковых и справочных систем, работы с текстовыми документами, а также для создания систем управления и проектирования.

Система <КОМПАС>

является графической системой, обеспечивающей конструктора и технолога современным инструментом по разработке и выпуску чертежно-графической документации, а также прикладных программистов — эффективным инструментом по разработке проблемно-ориентированных задач САПР. Система предоставляет пользователям широкие возможности формирования графической информации средствами:

«электронного кульмана»;

стандартной и пользовательской библиотек параметрических графических элементов;

формирования графических элементов из произвольной части текущего объекта и внесения их в другой объект (чертеж);

полуавтоматического оформления чертежа согласно требованиям ЕСКД (размеры, допуски, технологические объекты, технические требования, таблицы, обозначения, тексты, штриховки).

Система имеет развитые средства хранения и вывода графической информации, а также поддерживает интерфейс с программами, написанными на языках Pascal и Fortran. Наряду с этим в системе предусмотрен аппарат вспомогательных построений, облегчающих работу в режиме «электронного кульмана», а встроенный вычислитель позволяет в процессе работы производить необходимые расчеты. Средства фрагментации системы обеспечивают возможность разделения сложных объектов. Система <КОМПАС> достаточно проста и ориентирована на непрофессионального пользователя.

Система <КРОКУС>

представляет собой инструментальную систему для выполнения расчетных программ под ее управлением, а также анализа расчетного модуля с помощью средств машинной графики. Система предоставляет следующие возможности:

выбор необходимого расчета, т. е. программы из набора ППП;

ввод, просмотр и корректировку исходных данных;

ведение архива исходных данных;

выполнение требуемых расчетов и просмотр их результатов;

вычисление функции в произвольной точке заданного отрезка;

построение графиков зависимостей между входными и выходными (независимыми и зависимыми) переменными при заданных шаге изменения независимой переменной и количестве точек на графике;

вывод результатов расчета и графиков на принтер.

Система <КРОКУС> открыта и позволяет подключать автономно отлаженные расчетные модули, реализованные на любых языках программирования. Она может быть использована в качестве подсистемы САПР, оказывая существенную помощь конструктору на всех этапах разработки изделия.

Система <FORMA>

обеспечивает сокращение сроков проектирования пресс-форм и литейных форм, изготовление комплекта конструкторской документации, повышение качества разработки, уменьшение трудоемкости рутинных операций по привязке форм для термопластавтоматов (литейных машин). Объектами проектирования системы являются формы, съем деталей и отливок на которых осуществляется плитой или толкателями. При этом на деталях отсутствуют боковые отверстия и углубления, которые оформляются ползунами. Обрамляющая часть формы дорабатывается конструктором в диалоговом режиме. В диалоге конструктор может также отменять предлагаемые системой решения и заменять их своими, исходя из опыта, интуиции или принятых типовых решений на данном предприятии. Для своего функционирования система требует графический адаптер типа EGA или VGA и плоттер. Система <FORMA> использует диалоговую графическую систему <КОМПАС>, обеспечивающую вычерчивание произвольных деталей и компоновку чертежей любых объектов.

Система <ASPECT>

обеспечивает следующий комплекс работ:

1. Проектирование маршрутно-операционной технологии. Проектирование единичных, типовых и групповых технологических процессов может выполняться в любых отраслях машино- и приборостроения для различных типов и видов производства.

2. Нормирование трудозатрат. Расчет нормативов времени на выполнение операций и переход от операции к операции производится на основании технически обоснованных норм, действующих на конкретном предприятии.

3. Нормирование материалов при использовании проката в механогазотвительном производстве. Выполняются расчеты нормы расхода основных материалов, коэффициента использования материала, массы исходной заготовки.

4. Автоматизированный выбор технологической оснастки. В процессе проектирования выполняется поиск как универсальной, так и специальной оснастки в зависимости от обозначения детали, ее параметров и выполняемой технологической операции.

5. Формирование комплекта технологических документов. Состав комплекта разрабатывается по требованию заказчика.

6. Сохранение технологического процесса в архиве. Записанный в архив технологический процесс можно выводить на печать, корректировать или удалять по мере необходимости.

Система поддерживает удобные режимы работы с базами данных: их удобное наполнение, корректировку, быстрый поиск в базах, использование справочной информации, распечатку баз данных и их отдельных фрагментов.

Система <ASTRA>

относится к группе программных средств автоматизации технологической подготовки производства деталей, где в качестве исходных заготовок используются листовой, сортовой и фасонный прокат. Система предназначена для автоматизированного раскроя листовых материалов на полосы или штучные прямоугольные заготовки, рулонных материалов на ленты, других линейных материалов на мерные отрезки с целью их оптимального использования. Система обеспечивает следующий комплекс работ:

выбор вариантов схем раскроя проката из имеющегося сортамента с оптимальными расходными характеристиками с учетом вида разделительной операции и ограничениями по направлению волокон проката;

расчет характеристик выбранного варианта раскроя в зависимости от массы заготовки, количества требуемых деталей, нормы расхода материала и коэффициента использования материала;

учет отходов, используемых в производстве;

вывод на печать технологических документов по раскрою материалов согласно ГОСТ 3.1402—84.

Система <ASTRA> предусматривает интерфейс с САПР <ASPECT>, а полученная информация используется при проектировании технологических процессов холодно-штамповочного производства, при этом выполняется автоматическое проектирование и техническое нормирование заготовительных операций согласно выбранному варианту раскроя. Комплексная система <ASTRA-ASPECT> выбирает оптимальный вариант раскроя листового материала по критерию минимума себестоимости.

Система <STAMP>

обеспечивает сокращение сроков проектирования штампов и изготовления комплекта конструкторской документации, повышение качества разработки, уменьшение трудоемкости рутинных операций (поиск характеристик прессового оборудования и штампуемого материала, определение необходимого зазора между пуансоном и матрицей, расчет и подбор стандартных деталей штампа, включая стандартные пуансоны, а также устранение ручных копировальных работ). Объектами проектирования системы являются:

1) штампы простого и последовательного действия, однорядные, без твердо-сплавных элементов, с неподвижным съемником для деталей размером 200 × 200 мм и толщиной 0,5—10 мм, обеспечивающие отрезку, вырубку и пробивку в полосе или штучной заготовке с различными способами прижима полосы и ее фиксации (грибковый, цилиндрический, разовый упор, шаговый нож, лопатель, противоотжим);

2) гибочные штампы V-образных деталей с различными углами между планками и длиной полосок до 100 мм, П-образных деталей с длиной полок до 100 мм и минимальным расстоянием между полками более 25 мм.

САПР <STAMP> использует диалоговую графическую систему <КОМПАС>, обеспечивающую вычерчивание произвольных деталей и компоновку чертежей любых объектов, и требует для своего функционирования наличия графического адаптера типа EGA или VGA.

По вопросам получения более подробной информации и поставки описанных систем следует обращаться по адресу:

290005 Львов, ул. Мартовича 2, а/я 6629

НПП <АГРАФ>, телефон: (8—0322) — 724—001

ПРИЛОЖЕНИЕ 12

НОВЫЕ ЭФФЕКТИВНЫЕ СРЕДСТВА СЖАТИЯ И ВОССТАНОВЛЕНИЯ ФАЙЛОВ ДАННЫХ И ПРОГРАММ

В связи с обилием программных средств, большими объемами данных, особенно связанных с использованием различного рода СУБД, и относительно ограниченными ресурсами внешней памяти ПК (НГМД и НМД) остро стоит вопрос о более эффективном использовании магнитных носителей. Наряду с этим использование ПК в системах телеобработки и сетях ЭВМ выдвигает требования по минимизации обменных операций (с программами и данными). В связи с этим фирмами-разработчиками ПО были созданы программные средства (ПС), предназначенные для сжатия и восстановления программ и данных [33, 34], которые в настоящее время являются одними из наиболее часто используемых средств пользователей ПК и других классов ЭВМ. Стоимость таких ПС не превышает, как правило, 50 долл. Они безотносительно типа ПК и его внешней памяти позволяют существенно экономить память, повышая тем самым эффективность использования самих ПК. Данные ПС используют разнообразные алгоритмы для сжатия как бинарных файлов, так и файлов в коде ASCII. При этом наиболее популярными алгоритмами сжатия информации являются эффективные алгоритмы Шеннона — Фано, Хармана, Лемпеля — Зива, Лемпеля — Зива — Уэлша и их модификации [35, 36].

Из ПС сжатия и восстановления файлов наиболее популярными являются ARC 6.02, ICE 1.14 (известное также под именем LHARC), PAK 2.1, ZOO 2.01 и PKWARE 1.1. Каждое из перечисленных средств имеет свои особенности выполнения и использования. Так, например, программа ICE 1.14 наиболее популярна в Японии и обладает самыми высокими характеристиками сжатия файлов среди ПС подобного типа, однако она весьма медленна и неудобна, частично из-за недостаточной документированности. Поэтому данная программа не может быть рекомендована для ПК класса IBM PC/AT также незначительно меняет суть дела.

Пакет PKWARE фирмы PKware Inc. (США) является явным лидером по сжатию и восстановлению файлов, выполняя все основные функции сжатия и восстановления лучше аналогичных ему ПС. Пакет эффективно сжимает файлы данных и программ при весьма умеренных временных затратах. Так, для ряда популярных ПС и связанных с ними файлов данных (ФД) пакет дает следующие результаты по сжатию (объем приводится в байтах):

ПС и ФД	Полный объем	Сжатый объем	% сжатия
Ядро MathCAD 2.52	289.664	142.287	51
Пакет PKWARE 1.1	297.536	140.116	53
Turbo C 2.0	2.243.572	1.201.059	54
Windows 2.1	1.374.491	756.116	45
dBase 4	2.840.002	1.651.734	42

Файлы dBase 4	217.394	46.026	79
ASCII-файлы	230.578	83.326	65
ФД Lotus 1—2—3	262.761	95.400	64
Текст рукописи настоящей книги	639.104	278.909	56.4

Из приведенных данных, в частности, следует, что наибольший эффект, как правило, можно получить от сжатия ФД. Результаты сжатия ПС в определенной мере могут характеризовать качество их реализации и/или сложность и организацию их алгоритмов, тогда как результаты сжатия ФД — структуру содержащихся в них данных.

Пакет PKWARE хорошо известен среди пользователей, имеющих дело с телеобработкой, и фактически является стандартом для IBM PC совместимых систем телеобработки. В своем составе пакет PKWARE содержит 13 файлов, из которых основными являются следующие:

Файл	Назначение файла
PKZIP.EXE	Основная программа сжатия файлов (ZIP-файлы)
PKUNZIP.EXE	Основная программа восстановления ZIP-файлов
PKZIPFIX.EXE	Программа восстановления разрушенных ZIP-файлов
PKSFX.EXE	Создание самовосстанавливающихся ZIP-файлов
ZIP2EXE.EXE	Создание бинарных самовосстанавливающихся файлов
APPNOTE.TXT	Описание форматов ZIP-файлов, алгоритмов сжатия, восстановления и криптования файлов
ADDENDUM.DOC	Описание новых дополнений к версии 1.1 пакета
README.DOC	Руководство по восстановлению самого пакета из ZIP-файла PKZ110.EXE системной дискеты
MANUAL.DOC	Руководство пользователя по работе с пакетом

Пакет PKWARE 1.1 поставляется в виде самовосстанавливающегося ZIP-файла PKZ110.EXE и функционирует на ПК, совместимых с IBM PC/XT/AT и PS/2, под управлением MS DOS версий 2.0 и выше или OS/2. Так как пакет использует команду SHARE, поддерживаемую MS DOS версий 3.0 и выше, то рекомендуется работать с пакетом под управлением этих версий ОС, хотя это и необязательно. Для работы пакет требует не менее 85 К ОП, НГМД и/или НМД, при этом наличие последнего существенно повышает производительность пакета. Пакет может сжимать в один ZIP-файл до 3900 исходных файлов. Возможность создания самовосстанавливающихся ZIP-файлов не требует для их восстановления наличия самого пакета. Восстановление происходит путем запуска такого ZIP-файла. Программой ZIP2EXE пакета можно создавать бинарные самовосстанавливающиеся ZIP-файлы, которые особенно эффективны при использовании в системах телеобработки и сетях ЭВМ. Эти и другие возможности пакета PKWARE 1.1 делают его весьма полезным для самого широкого круга пользователей ПК, совместимых с IBM PC/XT/AT и PS/2.

ПРИЛОЖЕНИЕ 13

ХАРАКТЕРИСТИКИ СИСТЕМЫ MINIDOS/MathCAD

Приводятся основные сведения, подробнее описанные в соответствующих главах книги. Система MINIDOS/MathCAD была создана в ПТИ Минстроя ЭССР на основе ранее разработанной системы MINIDOS [10] для отечественного ПК ИСКРА 1030 и пакета MathCAD фирмы MathSoft Inc. версии 2.52. Созданная система является эффективной, надежной и компактной, располагается на двух дискетах емкостью 360 К. Согласно установке в карманы дисководов будем называть их соответственно А-дискетой и В-дискетой системы. Каждая из системных дискет содержит следующий набор файлов:

А-дискета

Основные файлы MINIDOS
MCAD.OVL; MCSTRANS.EXE
MCAD.MCP; MCAD.HLP
MCAD.MCC; MC12X6.MCF
MC8X8.MCF; MC9X6.MCF
MCADPS.MCF

В-дискета

MCAD.EXE
BASIC.COM
FORMAT.COM
SORT.EXE
BOOL.MCD
USERLIB.MCD

Файл BOOL.MCD содержит восемь основных булевых функций, позволяющих использовать в среде пакета MathCAD средства алгебры логики: NOT (отрицание), EQV (эквивалентность), AND (конъюнкция), PRS (стрелка Пирса), SEF (штрих Шеффера), XOR (разделительная дизъюнкция), IMP (импликация) и OR (дизъюнкция).

Файл USERLIB.MCD содержит функции пользователя, позволяющие во многих случаях существенно упрощать организацию вычислительного процесса в текущем документе. Файл включает следующие функции:

Idiv (X, Y) — целочисленное деление X на Y;
Nfig (X, K) — нахождение K-й цифры значения аргумента X;
B (X, Y) — Бета-функция;
Sol (A, B) — решение системы линейных уравнений $AX = B$;
F (A, λ) — вычисление собственного значения матрицы A;
Mom (X, K) — вычисление K-го эмпирического центрального момента выборки X;
Tol (X, E) — E-толерантность аргумента X;
U (X) — ступенчатая симметричная единичная функция;
 $U_p(X)$, $U_m(X)$ — асимметричные единичные функции;
Si (X) — интегральный синус;
Ci (X) — интегральный косинус ($X > 0$);
Ei (X) — интегральная показательная функция;
Li (X) — интегральный логарифм ($X > 0$);
 $\rho(X, A, B)$ — определение принадлежности X отрезку [A, B];
Int (X) — округление аргумента X до целого.

Функции в файлах BOOL.MCD и USERLIB.MCD определены глобально и по мере надобности могут дополнять текущий документ по команде 'Append' пакета и использоваться согласно их описаниям, приведенным в соответствующих главах книги.

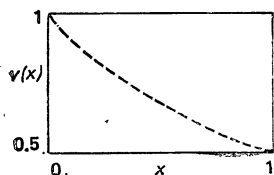
В отличие от пакета MathCAD 2.52 в системе MINIDOS/MathCAD принимаются для предопределенной переменной TOL значение 0.001 и для режима функционирования M-режим (Manual). Остальные значения по умолчанию можно получить по командам пакета Format, Plotformat, Surfaceformat и Sketchformat. После соответствующей установки дискет системы (А-дискета и В-дискета) включается ПК и с устройства «А» начинается загрузка системы MINIDOS. После получения ею ответа на запросы текущих даты и времени происходит автоматическая загрузка пакета MathCAD 2.52 с выходом на начальную заставку пакета. Дальнейшие действия пользователя определяются требованиями его задачи и возможностями пакета, описанными в настоящей книге. Для функционирования системы MINIDOS/MathCAD на ПК ИСКРА 1030 требуются следующие минимальные ресурсы: память 512 К и один дисковод для дискет емкостью не менее 360 К каждая.

По вопросам поставки системы MINIDOS/MathCAD обращаться по адресу: Таллинн 200038, Партизани 13—75, VASCO Ltd.

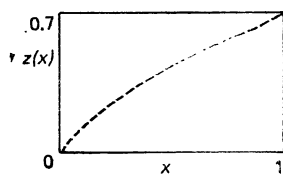
ПРИЛОЖЕНИЕ 14

РИСУНКИ К ПРИМЕРАМ ФРАГМЕНТОВ ДОКУМЕНТОВ

Приводятся рисунки к примерам фрагментов документов, которые по своему характеру весьма тесно связаны со структурой документов, вследствие чего по техническим причинам их целесообразно помещать в сам текст книги. Поэтому в тексте они в соответствующих местах примеров идентифицированы своими номерами, под которыми помещены в данном приложении. Остальные рисунки книги представлены традиционным способом.



Plot of y-function



Plot of z-function

Рис. 9.

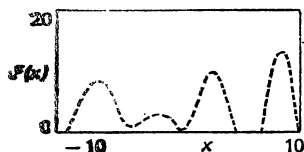


Рис. 10.

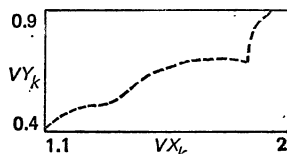


Рис. 11.

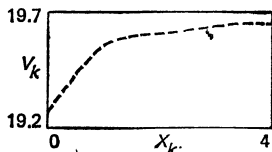
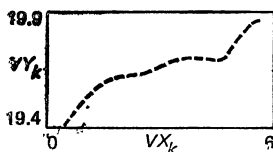
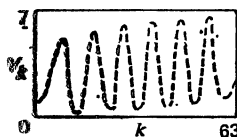
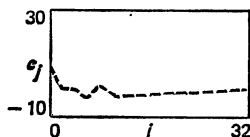


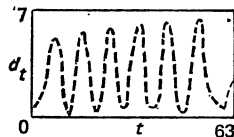
Рис. 12.



Исходный сигнал

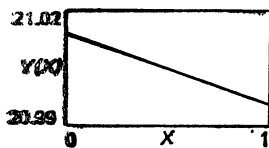


Кодирования

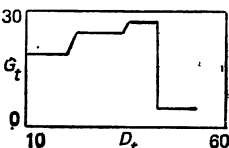


Декодирования

Рис. 13.



Линия регрессии



Гистограмма

Рис. 14.

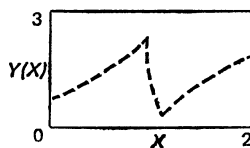


Рис. 15.

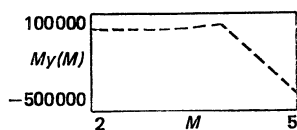
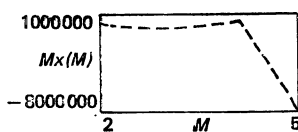


Рис. 16.

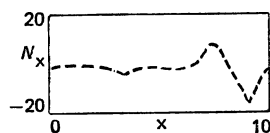
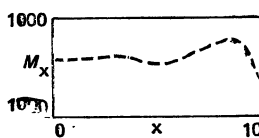


Рис. 17.

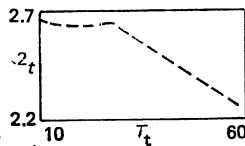
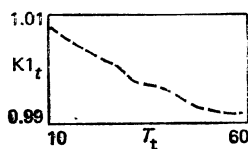


Рис. 18.

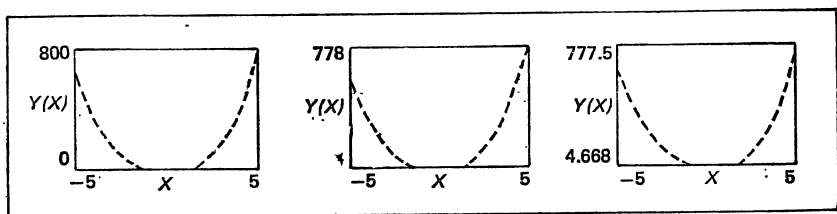


Рис. 20.

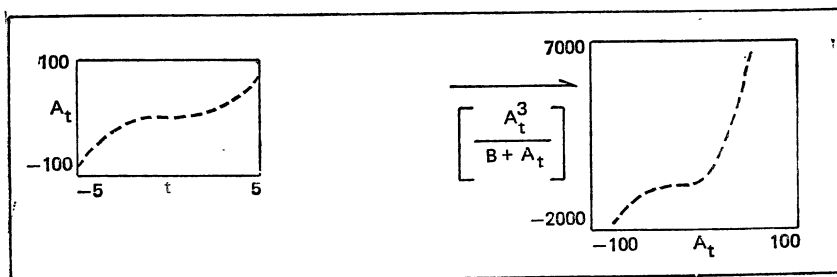


Рис. 21.

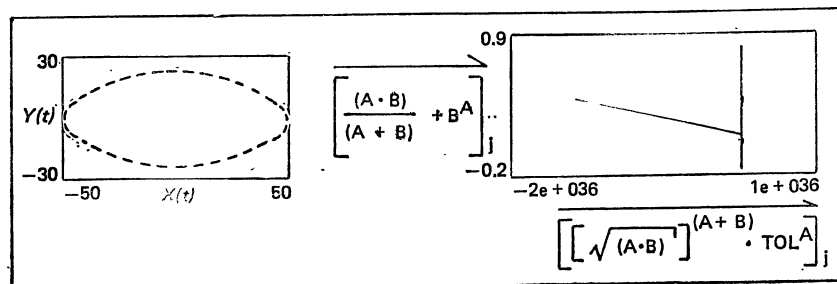


Рис. 22.

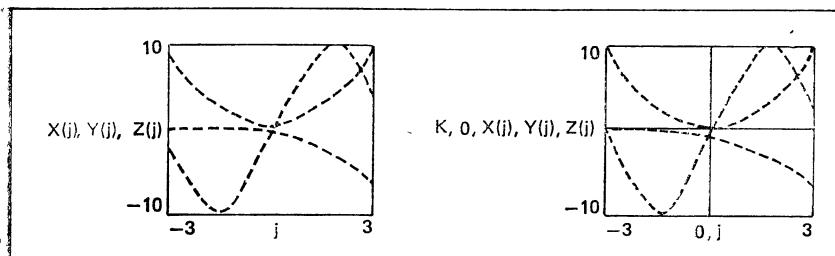


Рис. 23.

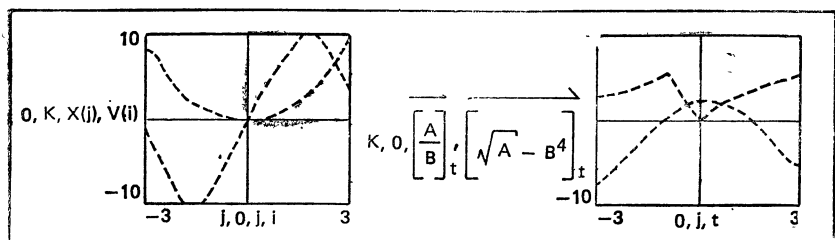


Рис. 24.

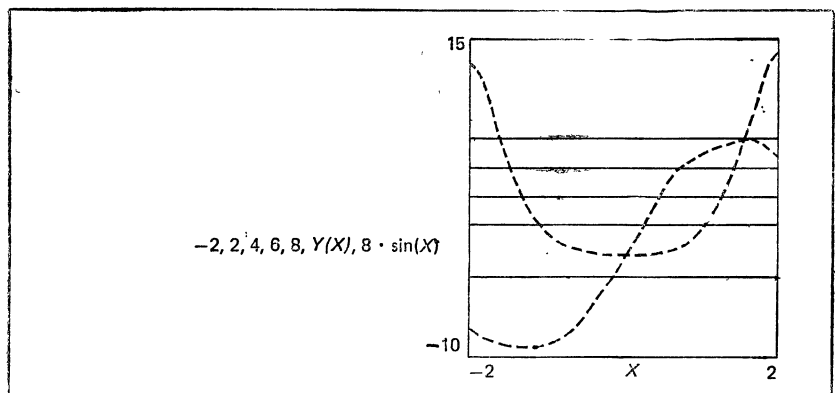


Рис. 25.

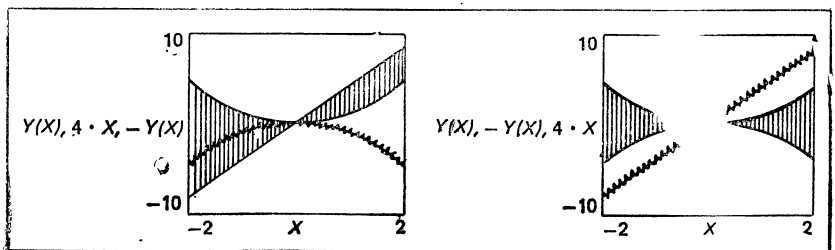


Рис. 26.

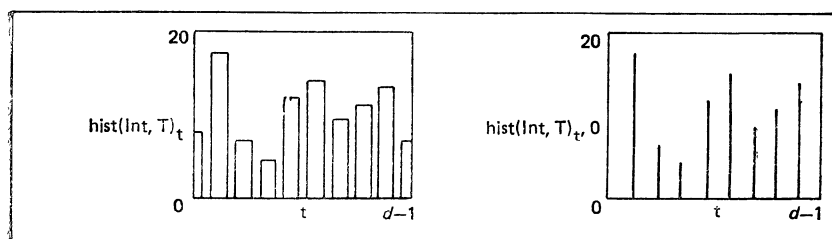


Рис. 27.

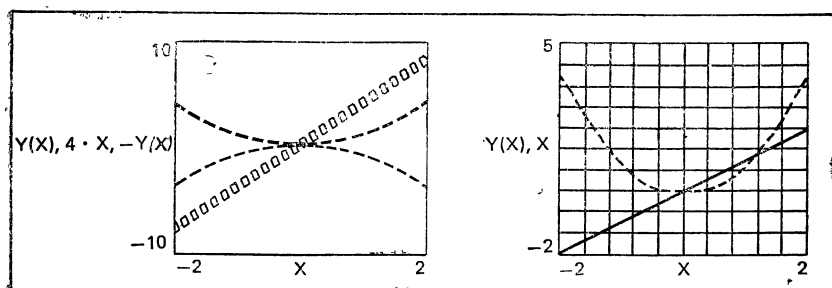


Рис. 28.

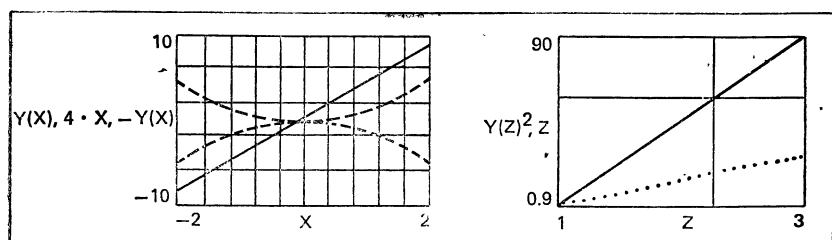


Рис. 29.

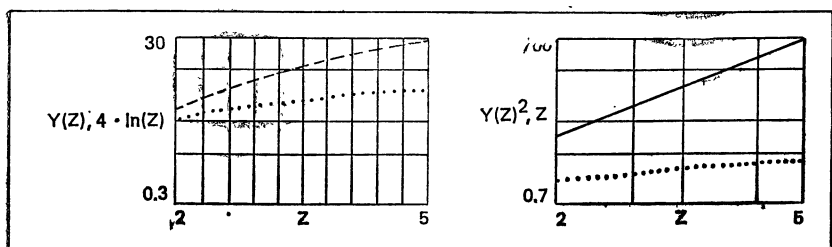


Рис. 30.

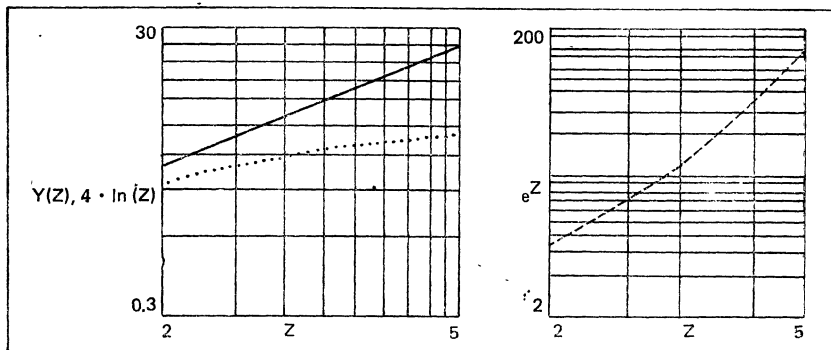


Рис. 31.

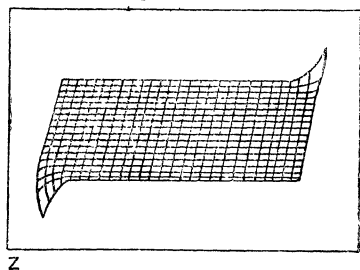
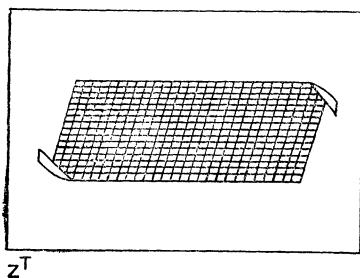


Рис. 33

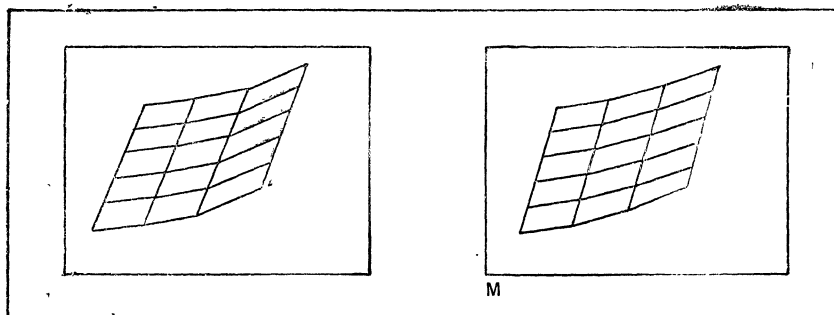
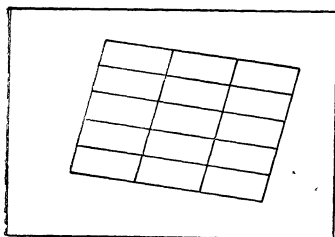
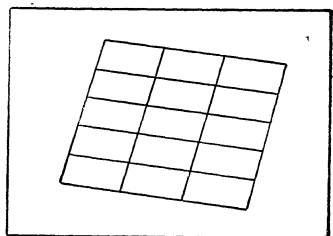


Рис. 34

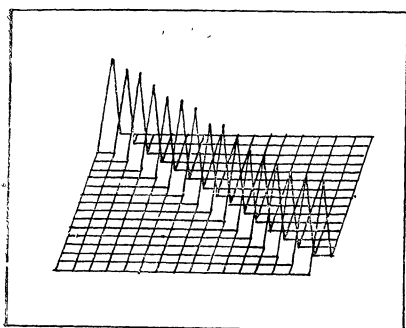


$$\text{Re}(Z - \text{Re}(Z))$$



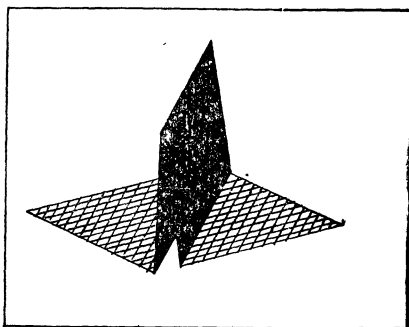
$$Z - \text{Re}(Z)$$

Рис. 35



$$Z + 5 \cdot \text{identity}(2 \cdot |P| + 1)$$

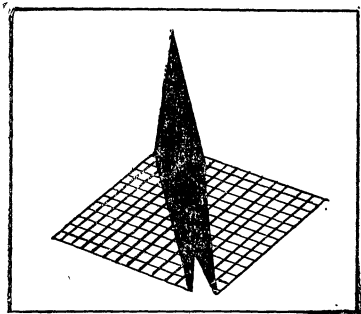
(1)



$$Z + 5i \cdot \text{identity}(2 \cdot |P| + 1)$$

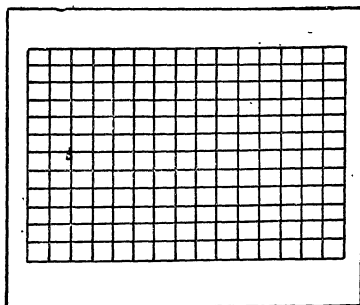
(2)

Рис. 36



$$Z + 5 \cdot \text{identity}(2 \cdot |P| + 1)$$

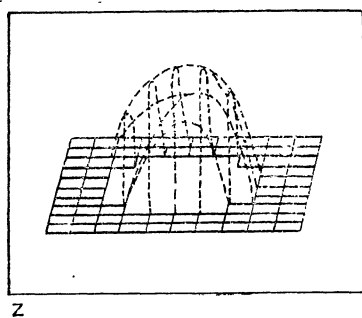
(1)



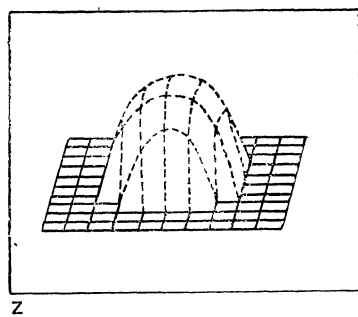
$$Z + 5i \cdot \text{identity}(2 \cdot |P| + 1)$$

(2)

Рис. 37.

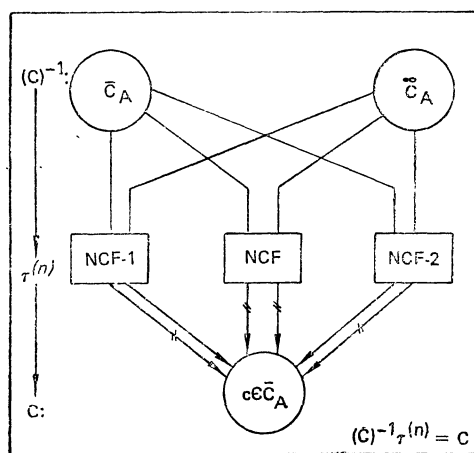


(1)



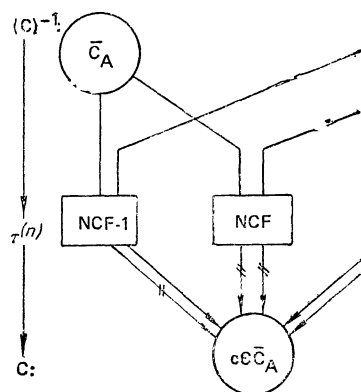
(2)

Рис. 38



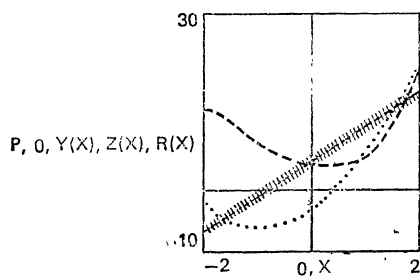
Рис

(1)



(2)

Рис. 39



$P, 0, Y(X), W(X), V(X)$

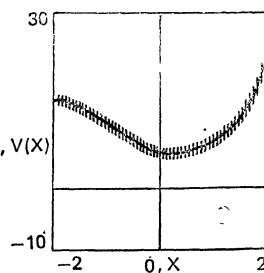


Рис. 40.

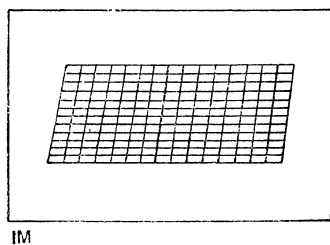
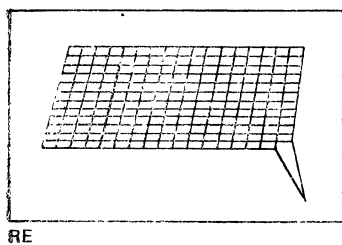
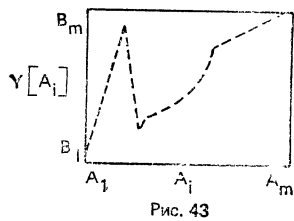
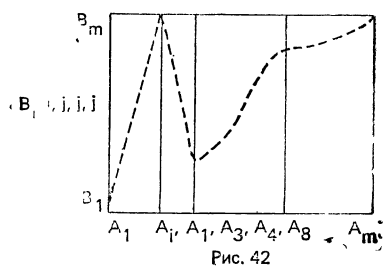
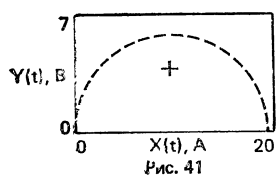


Рис. 44

СПИСОК ЛИТЕРАТУРЫ

1. Аладьев В. З., Осипов О. Б. Введение в операционную систему ЕС ЭВМ.— Таллинн: Валгус, 1975.— 180 с.
2. Аладьев В. З., Осипов О. Б. Введение в архитектуру моделей ЕС ЭВМ.— Таллинн: Валгус, 1976.— 310 с.
3. Математическое обеспечение ЕС ЭВМ и АСУ / Под ред. В. З. Аладьева.— Таллинн: Валгус, 1978.— 190 с.
4. Система управления базами данных на основе операционной системы МИНИОС и СУБД / Под ред. В. З. Аладьева. — Таллинн: Валгус, 1980.— 180 с.
5. Аладьев В. З. Архитектура и программное обеспечение СМ ЭВМ / ИВЦ ЭРК Госбанка СССР.— Таллинн, 1983.— 80 с.
6. Аладьев В. З. Курс лекций по персональному компьютеру ИСКРА-226 / СКБ МПСМ ЭССР.— Таллинн, 1986.— 180 с.
7. Аладьев В. З., Мартыненко Я. Г., Шиленко В. Ф. Персональный компьютер ИСКРА-226.— К.: Гл. ред. УСЭ, 1988.— 150 с.
8. Аладьев В. З., Сироджа И. Б. Программирование и решение инженерных задач в операционной среде БЕЙСИК персональной ЭВМ ИСКРА-226.— Харьков: Изд-во ХАИ, 1988.— 87 с.
9. Аладьев В. З., Краснопрошина А. А., Крижановский В. В., Мартыненко Я. Г. Проектирование программного обеспечения персональной ЭВМ ИСКРА-226.— К.: Техника, 1989.— 255 с.
10. Аладьев В. З., Шиленко В. Ф. Профессиональный персональный компьютер ИСКРА 1030.— К.: Гл. ред. УСЭ, 1990.— 484 с.
11. Аладьев В. З., Сироджа И. Б. Решение инженерных задач на персональном компьютере ИСКРА 1030 в среде языка БЕЙСИК.— М.: Высш. шк., 1991.— 245 с.
12. Аладьев В. З. Однородные структуры. Теоретические и прикладные аспекты.— К.: Техника, 1990.— 272 с.
13. Aberth O. Precise Numerical Analysis.— London: Brown Publishers, 1988.— 225 p.
14. Kohn M. Practical Numerical Methods. Algorithms and Programs // Collier Mac-Millan Ltd.— London, 1987.— 259 p.
15. Kaltofen E., Watt S. M. Computers and Mathematics.— Berlin : Springer-Verlag, 1989.— 326 p.
16. Constantinides A. Applied Numerical Methods with Personal Computers.— N. Y.: McGraw-Hill Book Company, 1987.— 626 p.
17. Shoup T. E. Applied Numerical Methods for Microcomputers.— London : Prentice-Hall, 1984.— 480 p.
18. Charpa S., Canale R. Numerical Methods for Engineers with Personal Computers.— N. Y.: McGraw-Hill Book Company, 1985.— 386 p.
19. MathCAD. Version 2.5. User's Guide / MathSoft Inc.— Cambridge, 1989.— 262 p.
20. MathCAD. Version 2.5. Quick Reference / MathSoft Inc.— Cambridge, 1989.— 12 p.
21. MathCAD. Version 2.5. Reference / MathSoft Inc.— Cambridge, 1989.— 50 p.
22. MathCAD. Version 2.5. Release Notes / MathSoft Inc.— Cambridge, 1989.— 5 p.
23. MathSoft User's Journal / MathSoft Inc.— Cambridge, 1986—1990.
24. Двайт Г. Б. Таблицы интегралов и другие математические формулы.— М.: Наука, 1973.— 228 с.
25. Фриш С. Э., Тиморева А. В. Курс общей физики.— М.: Наука, 1958.— 463 с.

26. Корн Г., Корн Т. Справочник по математике для научных работников и инженеров.— М.: Наука, 1973.— 831 с.
27. Tolimieri R., An M., Lu C. Algorithms for Discrete Fourier Transform and Convolution.— Berlin: Springer-Verlag, 1989.— 350 p.
28. McCarthy T. AutoCAD Express.— Manchester: IFS Publications Ltd., 1990.— 335 p.
29. Encarnacao J., Lindner R., Schlechtendahl E. Computer Aided Design.— Berlin: Springer-Verlag, 1990.— 400 p.
30. EUROCAST—89 // Proc. Intern. Workshop on Comp. Aided Systems Theory.— Las Palmas, 1989.— 256 p.
31. SCOOP // Europe Seminars and Conf. in Object-Oriented Programming.— London: Imperial College, July 16—18, 1990.— 127 p.
32. Printer NX-15 // User's Manual. Star Micronics Co., Ltd.— Tokyo, 1986.— 195 p.
33. Vaughan — Nichols S. J. Saving Space // Byte.— 3.— 1990.— P. 48—52.
34. Vaughan—Nichols S. J. Zippity. PKzip // PC Computing.— February, 1990.— P. 64—67.
35. Held G. Data Compression, Techniques and Applications // Hardware and Software Considerations.— John Walley & Sons, 1987.— 246 p.
36. Storer J. A. Data Compression, Methods and Theory.— London: Computer Science Press, 1988.— 327 p.
37. AutoSketch. User Guide.— London: AutoDesk Inc., 1987.— 115 p.
38. Five Ways to Software Savvy.— N. Y.: SIGS Publications Inc., 1990.— 48 p.
39. Journal of Object-Oriented Programming.— 3, N 1—2.— London.— 1990.
40. Фильчаков П. Ф. Справочник по высшей математике.— К.: Наук. думка, 1973.— 743 с.
41. Яворский Б. М., Детлаф А. А. Справочник по физике.— М.: Наука, 1965.— 847 с.
42. Шум. Методы определения шумовых характеристик источников шума.— М.: Издательство стандартов, 1982.— 30 с.
43. Аладьев В. З. Научно-техническая САПР.— К.: Гл. ред. УСЭ, 1991.
44. Аладьев В. З., Гиссовский В. Б., Пицюра И. С. Компьютерная хрестоматия.— К.: Гл. ред. УСЭ, 1992.
45. Аладьев В. З. Утилиты для персонального компьютера.— К.: Техника, 1992.
46. Kiatte R. PASCAL—XSC.— Berlin: Springer-Verlag, 1991.— 345 p.
47. Brockwell P. J., Davis R. A. ITSM—An Interactive Time Series Modelling Package for the PC.— Fort Collins: Colorado State University, 1991.— 140 p.
48. Dr. DOBB's Journal.— Redwood City: M&T Publishing Inc., 1991.
49. Journal of Object-Oriented Programming.— N. Y.: SIGS Publications Inc., 1991.
50. Inside Turbo-Pascal.— Louisville: The COBB Group Inc., 1991.

СОДЕРЖАНИЕ

Стр.

Предисловие	3
Введение	7
Глава 1. Основные характеристики пакета MathCAD и подготовка к работе с ним	10
1.1. Основные характеристики пакета MathCAD	10
1.2. Состав пакета, требуемые ресурсы и компоновка пакета для конкретных условий эксплуатации	13
1.3. Загрузка пакета на выполнение и параметры загрузки	20
1.4. Общие базовые операции в среде пакета MathCAD	22
Глава 2. Основы работы в среде пакета MathCAD	32
2.1. Основные понятия, определения, обозначения и соглашения	32
2.2. Базовые вычислительные конструкции и форматы результатов вычислений в среде пакета MathCAD	38
2.3. Размерность, единицы измерений и единицы масштабирования	44
2.4. Встроенные и пользовательские функции пакета MathCAD	47
2.5. Вычисление выражений и диагностика ошибок	49
2.6. Ранжированные переменные, таблицы и итеративные вычисления	53
2.7. Редактирование выражений документа	63
2.8. Создание двумерных графиков в среде пакета	70
2.9. Сохранение документа на диске и вывод его на устройство печати	74
Глава 3. Суммы, произведения, производные и определенные интегралы	81
3.1. Сумма и произведение ранжированных выражений	81
3.2. Численное вычисление производных	83
3.3. Вычисление определенных интегралов	86
Глава 4. Вычисления с векторами и матрицами в среде пакета	92
4.1. Создание векторов и матриц в среде пакета	93
4.2. Операции над векторами и матрицами	96
4.3. Параллельные определения и вычисления. Определение нетрадиционных операций над массивами	102
Глава 5. Решение уравнений, систем уравнений и неравенств	104
5.1. Вычисление нулевых корней уравнений	104
5.2. Определение решающих блоков в среде пакета	108
5.3. Использование решающих блоков в документе	112
5.4. Аварийные и особые ситуации при использовании решающих блоков в среде пакета	118
Глава 6. Встроенные функции пакета MathCAD	119
6.1. Встроенные функции интерполяции	120
6.2. Функции быстрого дискретного преобразования Фурье	122
6.3. Встроенные статистические функции	124
6.4. Функции управления вычислениями в документе	127
Глава 7. Методы доступа и работа с файлами данных	129
7.1. Доступ к неструктурированным файлам данных	130
7.2. Доступ к структурированным файлам данных	136
Глава 8. Текстовая информация в среде пакета MathCAD. Редактирование, форматирование и печать документов	146
8.1. Использование текстовых зон в документе	147
8.2. Команды перлюстрации и редактирования текста	150
8.3. Редактирование текущего документа	153
8.4. Форматирование и печать текущего документа	158
Глава 9. Работа с графиками и рисунками в среде пакета	165
9.1. Расширенное использование плоских графиков	166
9.2. Использование графиков функций от двух переменных	173

9.3. Использование в среде пакета изобразительных фрагментов из других САД-систем и систем программирования	178
Глава 10. Примеры использования пакета MathCAD для решения вычислительных задач	183
Заключение	204
Приложения:	
Приложение 1. Типы принтеров и плоттеров, поддерживаемых пакетом MathCAD 2.52	206
Приложение 2. Примеры документов, поставляемых с пакетом MathCAD 2.52	207
Приложение 3. Основные особенности работы в среде пакета MathCAD 2.52, выявленные в процессе его эксплуатации и апробации	209
Приложение 4. Клавиатура ПК, совместимых с IBM PC/XT/AT	214
Приложение 5. Перечень особых и аварийных ситуаций, идентифицируемых пакетом MathCAD 2.52	216
Приложение 6. Клавиши ПК, управляющие движением курсора в пределах текущего документа пакета	220
Приложение 7. Встроенные функции пакета MathCAD 2.52	220
Приложение 8. Основные операции, поддерживаемые пакетом MathCAD 2.52	221
Приложение 9. Основные управляющие и дополнительные клавиши пакета MathCAD 2.52	221
Приложение 10. Сводка команд пакета MathCAD 2.52	225
Приложение 11. Автоматизированные системы для решения конструкторских и технологических задач	228
Приложение 12. Новые эффективные средства сжатия и восстановления файлов данных и программ	231
Приложение 13. Характеристики системы MINIDOS/MathCAD	232
Приложение 14. Рисунки к примерам фрагментов документов	234
Список литературы	242

Производственно-практическое издание

**Аладьев Виктор Захарович
Гершгорн Николай Алексеевич**

**ВЫЧИСЛИТЕЛЬНЫЕ ЗАДАЧИ
НА ПЕРСОНАЛЬНОМ КОМПЬЮТЕРЕ**

**Редактор М. Р. Некрасова
Оформление художника Т. В. Чернышенко
Художественный редактор
В. С. Шапошников
Технический редактор
Н. А. Бондарчук
Корректор В. Н. Руденко**

Сдано в набор 14.05.91. Подписано в печать 13.08.91. Формат 60×90¹/₁₆. Бумага для массовых изд. Гарнитура литературная. Печать высокая. Усл. печ. л. 15,5. Усл. кр.-отт. 15,81. Уч.-изд. л. 18,21. Тираж 5000 экз. Зак. 1—1182. Цена 5 р.

Издательство «Техника». 252601 Киев, 1, ул. Крещатик, 5.

Отпечатано с матриц Харьковской книжной фабрики им. М. В. Фрунзе на Киевской фабрике печатной рекламы им. XXVI съезда КПСС, 252067, г. Киев-67, Выборгская, 84.

ФИРМА VASCO Ltd.

ОСНОВНЫМИ НАПРАВЛЕНИЯМИ ДЕЯТЕЛЬНОСТИ ФИРМЫ ЯВЛЯЮТСЯ:

- * современные и перспективные компьютерные технологии;
- * разработка, апробация и адаптация программных средств (ПС);
- * подготовка и издание литературы по перспективным ПС;
- * математическая теория однородных структур и ее приложения;
- * поставка вычислительной техники и программных средств;
- * посреднические услуги в указанных и других направлениях.

Фирма ищет деловых партнеров как в лице организаций, так и частных лиц, заинтересованных в сотрудничестве по указанным направлениям.

Наш адрес: 200106 Таллинн, Комсомоли 1—221.

Телефоны: (8-0142)—601-937, 453-888. Факс 453-923.

ПРЕДЛАГАЕТСЯ СЛЕДУЮЩАЯ ЛИТЕРАТУРА, ПОДГОТОВЛЕННАЯ СОТРУДНИКАМИ ФИРМЫ VASCO Ltd.:

1. Аладьев В. З., Мартыненко Я. Г., Шиленко В. Ф. Персональный компьютер ИСКРА 226.— К. : Изд-во УСЭ, 1988.— 150 с., ц. 65 к.

Рассматриваются технические и программные средства ПЭВМ ИСКРА 226, описаны основные операторы языка БЕЙСИК и особенности их выполнения. Описан ряд сервисных ПС и пакет A-BASIC, расширяющий выразительные средства среды языка БЕЙСИК.

2. Аладьев В. З., Сироджа И. Б. Программирование и решение инженерных задач в операционной среде БЕЙСИК ПЭВМ ИСКРА 226.— Харьков: Изд-во ХАИ, 1988.— 100 с., ц. 45 к.

Предназначена в качестве учебного пособия для студентов вузов по курсу «Основы информатики и вычислительной техники».

3. Аладьев В. З. и др. Проектирование программного обеспечения персональной ЭВМ ИСКРА 226.— К. : Техника, 1989.— 265 с., ц. 90 к.

Изложены вопросы программирования на ПЭВМ ИСКРА 226, особенности архитектуры и базового программного обеспечения (ПО). Основное внимание уделено методам построения надежных и легко модифицируемых программных комплексов широкого назначения.

4. Аладьев В. З., Шиленко В. Ф. Профессиональный персональный компьютер ИСКРА 1030.— К. : Изд-во УСЭ, 1990.— 484 с., ц. 3 р.

Описаны архитектура и ПО отечественного ПК ИСКРА 1030, совместимого с IBM PC/XT. Приводится характеристика основных компонент ПК и его ПО: операционной системы АДОС (MS DOS 2.0), языка программирования БЕЙСИК, текстового редактора и целого ряда других ПС широкого назначения. Рассчитана на широкий круг специалистов, использующих ПК, совместные с IBM PC/XT/AT.

5. Аладьев В. З. Однородные структуры: Теоретические и прикладные аспекты.— К. : Техника, 1990.— 272 с., ц. 3 р.

Изложены основные вопросы математической теории однородных структур (ТОС), представляющих собой новую перспективную среду моделирования многих дискретных параллельных процессов в различных областях науки и техники. Представленные прикладные аспекты ТОС включают в себя вопросы использования однородных структур (ОС) в качестве основы моделирования. При этом основное внимание уделено применению ОС в стратегически важной области информатики и вычислительной техники, физики, химии, математической биологии, медицине и др.

6. Аладьев В. З., Гершгорн Н. А. Вычислительные задачи на персональном компьютере.— К. : Техника, 1991.— 254 с., ц. 5 р.

7. Аладьев В. З. Научно-техническая САПР.— К. : Изд-во УСЭ,— 18 п. л., ц. 5 р. (4 кв. 1991).

В отличие от предыдущей книги внимание в ней акцентировано на приложениях пакета MathCAD в качестве научно-технической САПР в различных областях науки и техники, включая вычислительную математику, кибернетику, физику, химию, биологию, биофизику, медицину, технику и др. В этом отношении предлагаемый материал хорошо дополняет настоящую книгу.

8. Аладьев В. З., Гиссовский В. Б., Пицюра И. С. Компьютерная хрестоматия.— К. : Изд-во УСЭ,— 20 п. л., ц. 7 р. (2 кв. 1992).

Рассматриваются основы работы с такими известными пакетами для ПК как текстовым процессором SPRINT, табличным процессором QUATTRO, системой автоматизации чертежных работ AUTOSKETCH, экспертной системой EXPERT CHOICE и др., а также вопросы объектно-ориентированной технологии. Рассчитана на широкий круг пользователей ПК, совместимых с IBM PC/XT/AT и PS/2.

9. Аладьев В. З. и др. Утилиты для персонального компьютера.— К. : Техника, 6 п. л., ц. 3 р. (4 кв. 1992).

Утилиты — необходимая компонента инструментария программиста любого уровня. Приводятся описания и основы использования свыше 100 популярных утилит как зарубежной, так и отечественной разработки. Рассматриваются антивирусные, информационные, обслуживания файловых структур, администрирования ПК, управления вычислительным процессом, расширения возможностей внешних устройств ПК и другие группы утилит, часть из которых рядом известных зарубежных компьютерных журналов определялась как лучшие утилиты последних лет. Рассчитана на широкий круг пользователей ПК, совместимых с IBM PC/XT/AT и PS/2.

10. Аладьев В. З. и др. Объектно-ориентированное программирование.— К. : Техника, 25 п. л., ц. 7 р. (1 кв. 1993).

Объектно-ориентированное программирование (ООП) представляет дальнейшее развитие ПС 90-х годов и определяет новую компьютерную технологию. Вопросы ООП рассматриваются на основе системы программирования PASKAL, версий 5.5 и 6.0 фирмы BORLAND, являющейся признанным лидером в этом направлении.

Заказы на книги 1—4, 7, 8 направлять по адресу:
252094 Киев, ул. Сергиенко, 18,
Объединение «Киевкнига»,
а по книгам 5, 6, 9, 10 —
в адрес фирмы VASCO Ltd., указанный выше.

