

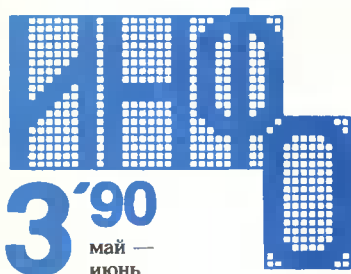
ИНФОРМАТИКА И ОБРАЗОВАНИЕ

3 1990





*Это не экспромт, не фантазия и не игра.
Это реально существующее и активно действующее
разновозрастное ребячье научное общество,
задача которого —
информационное обслуживание школы.
Статью о нем читайте в этом номере.*



ИНФОРМАТИКА И ОБРАЗОВАНИЕ

OldPC.su

7 0 0 3

музей компьютеров

Содержание

Методика обучения

Гольцман М., Дуванов А., Зайдельман Я., Первин Ю. Алгоритмические этюды	3
Манзюра В., Шмелев Д. Моделирование правил орфографии	11
Акимова Г., Деца В. Демонстраторы помогают учиться программированию	16
Сулима-Самойлов К. Опорные конспекты в курсе ОИВТ	19

Кабинет ВТ

КуМир и другие	25
Галицкий М. Коллектив исполнителей	33
Матвеев А. «Микроша»: 48К байт	36
Гинзбург М. Графика БК на уроке	40
Аляев Ю., Овчинников В. Защита НГМД	42
Новости рынка	43
Итоги конкурса	43
Клуб БК	44

Интервью номера

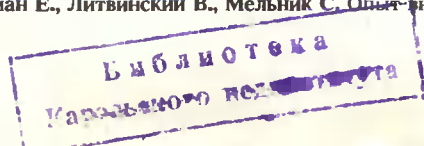
«Альтернатива» существует	55
Положение о кабинете вычислительной техники	60

Педагогический опыт

Житков Г., Прохорова М. Комплексное применение средств ВТ при обучении по курсу ТОР	76
Михеев Ю. Программное сопровождение курса планиметрии	78
Брудман Л., Распопов В. Задачи для ПМК	80
Голицына И., Нарыкова И. Компьютер на уроке физики	82
Елина М. Об организации взаимодействия вуза и школы	84

Внеклассная работа

Кельман Е., Литвинский В., Мельник С. Опыт внеклассной работы на СМ-4	86
---	----



Кузнецов С., Распопов В. Программирование логических игр	93
Переход И., Касаткин В. Моделирование и визуализация в программах учебной ориентации	96
Пахомова Н. Юношеское научное общество школьной информатики	100

ЭВМ в народном хозяйстве

Диагноз ставит компьютер	103
--------------------------	-----

Книги

Методички из США	106
Алгебра на дисплее	107

Нам пишут

Неделя информатики	109
Почтовый ящик	110

Информация

Пленум УМО по информатике	114
Быть ли единой системе?	115
Третья Ленинградская научно-методическая конференция	116
Всесоюзная конференция «Компьютер в школе и педвузе»	118
Международный конгресс «Образование и информатика»	119
Конференция в г. Ульяновске	120
Конкурс школьных химических программ на ЭВМ	122
Семинар в Узбекистане	123

Веселый урок

Жаворонков В. Барскиана, или Новые операторы языков программирования	125
--	-----

Главный редактор
 академик
В. А. МЕЛЬНИКОВ
 Редакционная
 коллегия
И. Н. АНТИПОВ
В. Н. АФНАСЬЕВ
И. М. БОБКО
Г. В. ГОДЖЕЛЛО
С. А. ЖДАНОВ
Б. В. ЛОМОВ
Ю. В. ЛУИЗО
 (зам. главного
 редактора)
Н. Г. МЕЛЬДИАНОВ
И. С. ОРЕШКОВ
О. К. ПАВЛОВА
А. Ю. УВАРОВ
А. И. ФУРСЕНКО
В. О. ХОРОШИЛОВ
К. В. ШЕХОВЦЕВ
 (редактор отдела)

Обложка Э. Бажиллина

Редактор отдела *А. Кравцова*
 Научный редактор *Т. Драгныш*
 Зав. редакцией *Н. Игнатова*
 Художественный редактор *Л. Розанова*
 Корректоры *О. Пурлова, М. Суворова*

Сдано в набор 26.03.90. Подписано в печать 26.04.90. А 06374.
 Формат 70×100¹/₁₆. Бумага офсетная № 1. Печать офсетная.
 Усл. печ. л. 10,40. Усл.-кр. отт. 42,88. Уч.-изд. л. 13,36.
 Тираж 72 635 экз. Заказ 599. Цена 60 коп.

Издательство «Педагогика» Академии педагогических наук СССР
 и Государственного комитета СССР по печати.
 Адрес для переписки: 107005, Москва, Лефортовский пер., 8.
 Адрес редакции: Студенческая ул., 19, корп. 1, кв. 17.
 Телефон редакции: 249-97-77

Ордена Трудового Красного Знамени Чеховский полиграфический комбинат
 Государственного комитета СССР по печати.
 142300, г. Чехов Московской обл.

© «Педагогика», «Информатика и образование», 1990.

М. ГОЛЬЦМАН, А. ДУВАНОВ, Я. ЗАЙДЕЛЬМАН, Ю. ПЕРВИН

Алгоритмические этюды

С третьего урока наряду с обсуждением темы «Информация» в Роботландии начинает развиваться алгоритмическая линия. Здесь также нет тяготения к строгим определениям: школьники осваивают новые понятия, обсуждая серии примеров. Набор разнообразных задач постепенно вводит детей в мир алгоритмов. В методических комментариях введение в алгоритмику называется «Алгоритмические этюды».

Урок 3. Тема: «Работа с информацией».

П л а н у р о к а:

Обсуждение домашнего задания — примеров информации.

Что можно делать с информацией. Примеры учителя.

Обсуждение примеров учащихся.

Выделение видов информационных процессов.

Обсуждение рецепта приготовления манной каши.

Рецепт манной каши — план решения задачи.

Практическая работа: игра-программа «Мудрый Кролик».

Домашнее задание: план приготовления манной каши.

На уроке учитель задает вопрос, умеют ли дети сами приготовить манную кашу. Трудно рассчитывать на единодушный утвердительный ответ. Тогда учитель может предложить рецепт. Школьники воспринимают план приготовления каши не только со слов учителя. Текст рецепта выводится на демонстрационный экран (записывается на доске). Хотя рецепт станет сегодняшним домашним

заданием, не беда, если дети его увидят; **3** буквально они его не запомнят, а форма представления подскажет способ выполнения задания.

Итак, чтобы решить задачу, надо составить план, который можно записать по-разному, лишь бы он оказался понятным исполнителю. Например:

1. Налить в кастрюлю 1 л молока.
2. Поставить кастрюлю на плиту.
3. Включить горелку, довести молоко до кипения.
4. Засыпать чайную чашку манной крупы.
5. Положить 2 чайных ложки сахарного песка.
6. Положить щепотку соли.
7. Равномерно помешивать 5 мин.
8. Выключить горелку.
9. Положить одну столовую ложку сливочного масла.

Урок 4. Тема: «План решения задачи — алгоритм».

Обсуждение домашнего задания — плана приготовления манной каши.

Составление плана решения. План — информация.

Постановка задачи о перевозке волка, козы и капусты.

Способ записи команд алгоритма.

Цифры. Русские буквы. Заглавные и строчные буквы. Клавиша стирания BS. Пробел.

Лабораторная работа: набор на клавиатуре слов заглавными и строчными буквами; набор предложений.

Домашнее задание: описать алгоритм перевозки.

Учитель просит детей зачитать рецепты, настойчиво называя каждый из них «планом решения задачи». Затем он спрашивает:

— Как можно назвать сообщения, которые были пунктами ваших планов?

Среди ответов, вероятно, будут фигурировать слова «информация» и «план решения» (или «план приготовления манной каши»). Во всяком случае, учитель может вывести класс на комбинацию этих терминов.

— Верно! План решения задачи — тоже информация. Ведь план — это очень важное сообщение о том, как надо решать задачу. Оно состоит из описаний отдельных действий повара, который варит манную кашу. Такие планы, состоящие из отдельных действий, мы будем называть *алгоритмами*.

Следующий важный момент урока начинается вопросом учителя:

4 — Как вы думаете, почему среди ваших решений задачи о манной каше нет в точности одинаковых?

Из многочисленных ответов на этот вопрос надо отобрать примерно такой: «Одинаковые действия могут быть выражены разными словами».

Если ввести точные обозначения для действий, то можно избавиться от такого разнобоя. Например, можно написать плакат: «Переход улицы разрешен только здесь». Можно заменить его другим: «Пожалуйста, переходите улицу в этом месте». Однако и тот, и другой выражают словами тот же смысл, который скрыт в понятном каждому школьнику знаке дорожного движения.

Все следующие задачи, которые встретятся на уроках информатики, мы будем решать так: обсудим задачу, договоримся о единообразном обозначении действий и только потом приступим к решению. Тогда алгоритм будет всегда иметь точное формальное описание — как формула в математике.

Первая из таких задач на составление алгоритма — это общеизвестная задача-сказка про перевозчика, перед которым стоит проблема перевезти через реку волка, козу и капусту в маленькой лодке. Дома ребят ждет задание: написать алгоритм перевозки. Чтобы подготовить их к этой работе, надо подробно рассмотреть задачу в классе.

Прежде всего надо убедиться в том, что правила перевозки понятны всем. Здесь важно проигнорировать отдельные восторженные выкрики «А я знаю!». Лучше дать высказать свои предложения по первым шагам (но не

далее второго шага!) тем, кто впервые познакомился с этой задачей. Работа по толкованию условий задачи оказывается полезной, поскольку дети, столкнувшись с логическими трудностями составления плана перевозки, начинают изобретать самые непредсказуемые решения. Вот примеры, взятые из реальных уроков: «...надо сначала накормить волка так, чтобы он на козу и смотреть даже не захотел, а потом уж перевозить», «...надо научить козу грести и отправить в лодке одну», «...воткнуть капусту на рога козе, тогда она ее съесть не сможет».

Распоряжаясь этими «идеями» в соответствии с чувством юмора тех, кто предлагает подобные решения, учитель постоянно требует строгого соблюдения правил.

Добившись понимания условия задачи, надо вернуться к обозначениям, которые можно применять в алгоритме перевоза. Формализация записи алгоритма проводится впервые, поэтому слово — учителю.

В команде для перевозчика надо указать направление перевоза:

когда перевозчик едет с левого берега на правый, это можно отметить стрелкой «→»; когда надо отправиться в путь с правого берега на левый — стрелкой «←».

Кроме того, в команде, т. е. рядом со стрелкой, надо указать, кто поедет в лодке: если в лодку садится волк, то пишем слово ВОЛК; когда в лодку села коза, около стрелки отметим — КОЗА; если старик взял с собой капусту, надо написать слово КАПУСТА; наконец, если старик едет в лодке один, то около стрелки ничего писать не надо.

В таких обозначениях требуется описать весь алгоритм перевоза. Попробуйте записать, например, самую первую команду алгоритма. Первый ход в этой задаче, допускающей, вообще говоря, два решения, однозначен. Разветвление начинается с третьего хода.

Итак, первый ход: → КОЗА

Учитель делает эту запись и прерывает творческий порыв класса: «Остальную часть алгоритма описать в домашних тетрадях».

Урок 5. Тема: «Работа с информацией, хранение».

П л а н у р о к а:

Обсуждение домашнего задания — формального описания алгоритма перевозки волка, козы и капусты через реку.

Игра в классе.

Лабораторная работа: Перевозчик.

Примеры хранения информации.

Информационные носители: от древнего мира до наших дней.

Способы хранения. Примеры.

Практическая работа: диалог с машиной по программе Привет.

Домашнее задание: придумать примеры хранения информации.

Проверка домашнего задания — это обсуждение алгоритма перевозки волка, козы и капусты с левого берега на правый. Проверку целесообразно осуществить в два этапа. Хотя формальная запись команд школьникам уже знакома, очень важно добиться неформального понимания алгоритма решения.

Первый этап проверки — игра, которую легко организовать в классе. Учитель может предложить детям разыграть спектакль «Перевоз». На полу условно отмечаются берега речки. На одном берегу сосредоточены все персонажи. Выбор исполнителей на роли старика-перевозчика, волка, козы и капусты не составит труда. Правда, нельзя быть уверенным, что юные исполнители смогут ярко воплотить образы своих героев, поэтому учитель должен заготовить головные уборы-ленточки с надписями «Перевозчик», «Волк», «Капуста», «Коза» и раздать их артистам. Они надевают ленточки так, чтобы весь класс мог видеть, как будут перемещаться персонажи сказки.

В спектакле есть еще один участник — автор предлагаемого решения, который должен подавать команды артистам. На эту роль вызывается ученик, который в предварительных устных обсуждениях успел показать правильный подход к построению алгоритма. Автор-резонер, встав немного в стороне от группы персонажей, произносит команды одну за другой. Следуя этим командам, участники «переплывают» (переходят) с одного берега на другой. После выполнения каждой команды учитель высписывает ее для демонстрации всему классу.

Если спектакль достигнет цели и вызовет энтузиазм зрителей, можно, сменив состав исполнителей, повторить сценку еще раз.

Не менее важным, чем настроение, представляется результат игры — закрепление понятий алгоритма и его выполнения. Если позволяет время, можно задать ряд дополнительных вопросов: «Кто во время игры подавал команды?», «Кто исполнял их?», «Мог ли исполнитель не знать алгоритма пере-

воза?». Эти вопросы направлены на опережение, они готовят учеников к одной из последующих тем — «Исполнители».

После этого можно переходить к проверке алгоритма в его формальной записи.

Игра и анализ алгоритма подготовили учеников к практической работе. Теперь тот же спектакль каждый из них сыграет на машине.

С фрагмента урока, посвященного перевозу через реку, дети уносят очень важное понятие алгоритма. Они составили алгоритм из отдельных команд и увидели процесс выполнения алгоритма, выполняя на компьютере одну за другой его команды. Важную информацию должен извлечь и учитель. Его ученики познакомились (пока еще не осознавая этого) с понятием исполнителя: Перевозчик — типичный исполнитель. Такой термин учащиеся будут использовать позднее для определения роботов (или их программных моделей), умеющих понимать и выполнять фиксированный набор команд. На текущем уроке учитель не пользуется новым термином.

Характерная черта любого исполнителя: реакция на ошибки человека, управляющего исполнителем. Различаются ошибки трех типов:

1) синтаксические — ошибки формальной записи отдельных команд (пример: ученик вводит имя пассажира, забыв задать направление перевоза; естественная реакция на синтаксическую ошибку — сообщение исполнителя НЕ ПОНИМАЮ);

2) семантические — ошибки, заставляющие исполнителя выполнять действия, которые выходят за пределы его возможностей (пример: находясь на правом берегу, Перевозчик получает команду «Переехать на правый берег»); семантические ошибки чаще всего свидетельствуют о том, что школьник не усвоил сущность алгоритма, по которому работает исполнитель (хотя исключать возможность случайной описки или нечаянного нажатия посторонней клавиши нельзя и в этом случае); на семантические ошибки исполнитель реагирует сообщением НЕ МОГУ;

3) логические — формально правильные (с точки зрения исполнителя) действия, не приводящие к решению задачи; причина ошибки — неверный алгоритм (пример: давая в начальном положении команду Перевозчику «→КАПУСТА», школьник может правильно набрать команду и не нарушить при этом ни одно из правил игры, но, тем не менее, получит на экране сообщение

«Съели!» и даже увидит динамичную, хотя и довольно грустную, картину пожирания козы волком).

Сообщения НЕ ПОНИМАЮ и НЕ МОГУ учащиеся будут наблюдать на экранах компьютеров, работая не только с Перевозчиком, но и со многими другими исполнителями. Сообщения о синтаксических и семантических ошибках унифицированы для всех исполнителей. Не акцентируя внимание на общности этих реакций (возможно, дети сами смогут прийти к обобщениям еще до изучения темы «Исполнители»), учителю полезно каждый раз при возникновении синтаксических и семантических ошибок в работах с исполнителями спрашивать детей: «Почему машина не может выполнить эту команду? Почему компьютер не понимает вашу команду?» Такой постоянный контроль представляет собой не только форму работы над ошибками, но и опережение (пропедевтику) фундаментального понятия исполнителя.

Урок 7. Тема урока: «Передача информации. Ханойская башня».

План урока:

Обсуждение домашнего задания: порядок в домашней библиотеке.

Обсуждение примеров передачи информации.

Игры в передачу информации.

Постановка задачи о Ханойской башне.

Решение задачи с двумя кольцами на детской пирамидке.

Способ формальной записи решения.

Знаки препинания на клавиатуре.

Лабораторная работа: знаки препинания и специальные символы.

Домашнее задание: составить и формально описать алгоритм переноса трех колец Ханойской башни.

Важная тема урока — Ханойская башня. Учитель рассказывает задачу, иллюстрируя ее обычной детской пирамидкой с цветными разнокалиберными колечками и двумя заранее подготовленными стержнями-карандашами. Сначала задача формулируется в общем виде и подробно объясняются правила переноса, которые описаны в § 2.2.

§ 2.2. Ханойская башня

Одна из самых любимых игрушек в младшей группе детского сада — пирамидка с цветными кольцами. Кольца нанизаны на стерженек так, что для

любой пары колец нижнее больше верхнего.

Есть страны, где такой игрушкой любят играть взрослые. В давние-предавние времена, в тех далеких краях, где сейчас расположена страна Вьетнам, в древнем городе Ханое жили монахи, которые придумали игру, очень похожую на детскую пирамидку. У них была одна полная пирамидка, а от двух других только пустые стержни.

По имени города игра названа Ханойской башней. В игре требуется перенести кольца с одного стержня (например, со стержня 1) на другой (например, на стержень 3), используя еще один стержень (например, стержень 2) в качестве промежуточного.

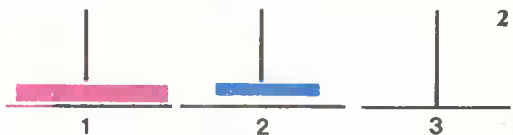
Правила игры в Ханойскую башню просты:

за одно действие можно переносить только одно кольцо;

любое кольцо можно укладывать либо на большее кольцо, либо на свободный стержень.

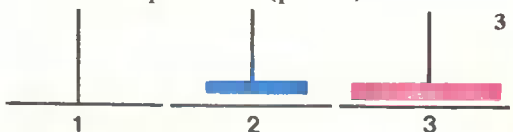
Нетрудно догадаться, что чем больше колец на пирамидке, тем труднее игра. В Ханойскую башню с двумя кольцами сыграть совсем легко. Вот алгоритм переноса двух колец.

1. Исходное положение — рис. 1. Перенести малое кольцо со стержня 1 на стержень 2 — промежуточный (рис. 2).



2. Перенести большое кольцо со стержня 1 на стержень 3 — окончательный (рис. 3).

3. Перенести малое кольцо со стержня 2 на стержень 3 (рис. 4).





Чтобы запись алгоритма была короче, обозначим ходы двумя числами: первое из них означает номер стержня, с которого кольцо снимается, а второе — номер стержня, на который кольцо надевается. В таких обозначениях алгоритм для двух колец можно переписать так:

1 — 2
1 — 3
2 — 3

Если есть алгоритм переноса со стержня 1 на стержень 3, то нетрудно написать алгоритм, переносящий кольца со стержня 1 на стержень 2:

1 — 3
1 — 2
3 — 2

Здесь стержень 2 играет роль окончательного, а стержень 3 — промежуточного. Точно так же, если потребуется перенести кольца со стержня 2 на стержень 3, то алгоритм переноса окажется очень похожим на два предыдущих:

2 — 1
2 — 3
1 — 3

(2 — исходный стержень, 3 — окончательный, 1 — промежуточный).

Можно о всех трех алгоритмах сказать одними и теми же словами:

- 1) перенести малое кольцо с исходного стержня на промежуточный;
- 2) перенести большое кольцо с исходного стержня на окончательный;
- 3) перенести малое кольцо с промежуточного стержня на окончательный.

В этом описании достаточно представлять разные номера стержней вместо слов «исходный», «окончательный» и «промежуточный» и получать описания любого алгоритма переноса двух колец.

Вопросы и задания:

1. Опишите алгоритм переноса трех колец Ханойской башни.
2. Опишите алгоритм переноса четырех колец Ханойской башни.

Когда правила освоены, начинается созда-

ние, а затем выполнение алгоритма переноса. Сначала работа ведется с двумя кольцами. Сначала работа ведется с двумя кольцами. Один из вызванных к доске учеников произносит команды алгоритма. Другой, исполнитель, эти команды выполняет, перекладывая кольца со стержня на стержень. Учитель должен с самого начала отметить стержни номерами 1, 2 и 3 и строго следить за тем, чтобы в произносимых учениками командах фигурировали номера стержней: это база для последующего перехода к формальной записи алгоритма.

Игру можно повторить еще один раз, а затем надо подвести детей к необходимости формализовать команды алгоритма. С этой целью можно напомнить, как изучался алгоритм перевоза через реку: сначала выраженные словами команды, потом — переход к их строгим обозначениям.

Можно предложить детям самостоятельно выбрать обозначения, но в этом случае надо быть готовым к всевозможным излишествам: дети могут указывать в команде цвет кольца, задавать его размер. Поэтому, не слишком затягивая этот фрагмент урока, учитель может предложить (если такое предложение еще не прозвучало) каждую команду задавать двумя цифрами: первая из них — номер стержня, с которого снимается кольцо; вторая — номер стержня, на которое кольцо надевается. В обоих случаях речь идет только об одном, самом верхнем кольце указываемого стержня. Даже стрелка, показывающая направление переноса, избыточна, поскольку направление однозначно определяется порядком следования номеров стержней в команде.

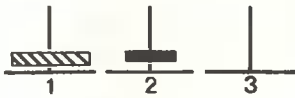
В таких обозначениях алгоритм переноса двух колец (уже обыгранный на пирамидке) со стержня 1 на стержень 3 записывается учителем так, как в § 2.2. Столбец с формальными командами алгоритма занимает левую половину демонстрационного экрана (доски). На правой половине изображены соответствующие ситуации алгоритма переноса (рис. 5). И правая, и левая части изображения не заготовлены, а создаются в ходе урока. Рисунки пирамидок могут быть стилизованными, но они необходимы: столбец из цифр, не сопровождаемый рисунком, плохо воспринимается, даже если произнесен устный комментарий.

Перед школьниками ставится новая, очень похожая задача: перенести два кольца с первого стержня, но не на третий, а на второй. Стержень 3 используется в этой задаче

5

исходное
состояние

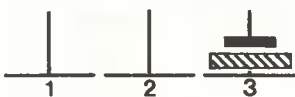
1 - 2



1 - 3



2 - 3



8 в качестве промежуточного. Технология решения остается той же. В выполнении алгоритма по-прежнему участвуют дети, а учитель фиксирует команды.

Школьники должны быть убеждены, что они умеют выполнять любой перенос двух колец.

Когда постановка задачи ясна, наступает очередь домашнего задания. Оно состоит в описании алгоритма переноса трех колец с соблюдением принятых на уроке обозначений.

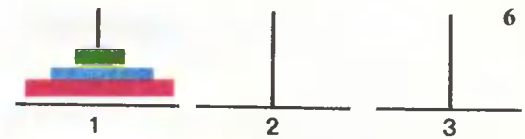
Хотя компьютеры еще не включались, рассуждения о переносе колец потребовали умственного напряжения учащихся. Небольшая разрядка необходима. Именно здесь уместно рассказать об истории игры «Ханойская башня», о происхождении ее названия, о легенде про конец света.

Восьмой и девятый урок, вообще говоря, непросты. Тем не менее в кратком описании уроков Роботландии мы сознательно опускаем методические комментарии к ним, поскольку в этих уроках в еще большей мере, чем в остальных, учителю рекомендуется проявить самостоятельность и учесть уровень подготовки детей.

В простейших ситуациях достаточно ограничиться рассмотрением алгоритма переноса трех колец (разумеется, с его машинной реализацией). Другой полюс этой части курса — обсуждение понятия рекурсии, введение формальных параметров и имен алгоритмов.

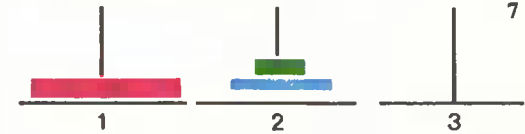
§ 2.3. От сложного к простому

Попробуем переложить Ханойскую башню из 3 колец (рис. 6).

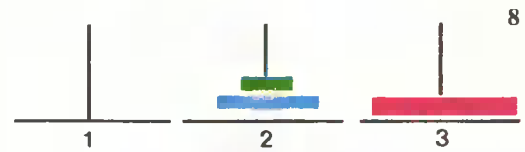


Давайте порассуждаем. Самое трудное — это переложить нижнее, большое кольцо. Его нельзя класть ни на какое другое кольцо, а можно только на пустой стержень.

Чтобы переложить большое кольцо с первого стержня на третий, надо куда-то убрать два кольца поменьше. Куда? Конечно же, на промежуточный стержень (рис. 7)! А переносить башню из 2 колец мы уже умеем!



После этого можно перенести большое кольцо (рис. 8).



Осталось перенести башню из двух колец со второго стержня на третий, а этот алгоритм нам тоже известен. Задача решена (рис. 9)!



Запишем алгоритм решения:

1) перенести два верхних кольца со стержня 1 на стержень 2 по известному алгоритму;

2) перенести нижнее кольцо на стержень 3;

3) перенести два кольца со стержня 2 на стержень 3 по известному алгоритму.

Теперь нужно вместо слов «известный алгоритм» переписать действия алгоритма переноса двух колец:

- 1) 1 — 3
 1 — 2
 3 — 2
 2) 1 — 3
 3) 2 — 1
 2 — 3
 1 — 3

Если вместо номеров стержней подставить в этот алгоритм слова «исходный», «промежуточный» и «окончательный», получится общий алгоритм переноса трех колец. Пользуясь этим алгоритмом, можно перенести три кольца с любого стержня на любой.

Умея переносить 2 кольца, мы легко перешли к задаче с 3 кольцами. Теперь, когда у нас есть алгоритм переноса 3 колец, попробуем переложить 4 кольца:

1) сначала освободим нижнее кольцо. Для этого перенесем верхние 3 кольца на промежуточный стержень по известному алгоритму;

2) перенесем самое большое кольцо с исходного стержня на окончательный;

3) перенесем 3 кольца с промежуточного стержня на окончательный по известному алгоритму.

Зная алгоритм переноса 4 колец, можно решить задачу с 5 кольцами, затем с 6 и т. д.

Монахи древнего Ханоя верили, что конец света наступит тогда, когда будут перенесены 64 кольца. Это, конечно, сказка, но если говорить серьезно, то на перекалывание башни из 64 колец не хватит долгой человеческой жизни, даже если не пить, не есть, не спать, а только переносить кольца.

Вопросы и задания:

1. Запишите общий алгоритм переноса 3 колец. Пользуясь общим алгоритмом, запишите алгоритм переноса 3 колец со стержня 2 на стержень 3.

2. Составьте алгоритм переноса 4 колец Ханойской башни.

3. Известно, что 5 колец можно перенести за 31 перекалывание. Сколько перекалываний понадобится для переноса 6 колец?

4. Рядом с каждым действием алгоритма переноса 3 или 4 колец напишите номер стержня, который не участвует в перекалывании.

5. Рядом с каждым действием алго-

ритма переноса 3 или 4 колец напишите номер кольца, которое переносится при этом перекалывании.

На следующих уроках разбирается еще несколько типовых алгоритмических этюдов, на которых мы здесь не останавливаемся. Последний этюд более сложен, он развивает уже сложившееся у детей представление об алгоритмах, готовит их к восприятию нелинейных алгоритмов.

§ 2.9. Угадайка

В эту игру играют двое. Один задумывает число, второй его угадывает. Потом наоборот. Выигрывает тот, кто угадает за меньшее число ходов.

Как происходит игра? Угадывающий предлагает товарищу на пробу какое-нибудь число. Тот отвечает:

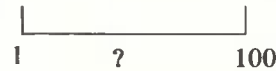
мало, если задуманное число больше;
много, если задуманное число меньше;
угадано, если числа совпали.

Такой вопрос-ответ считается одним ходом.

Перед началом игры надо договориться, какой величины может быть задуманное число.

Вот как Вася Ляпушкин отгадывал число Пети Кука.

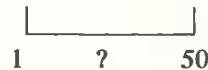
Петя. Я задумал число от 1 до 100.



Вася. Я не знаю, какое число ты задумал, но знаю, что оно между 1 и 100. Возьму для пробы число посередине: 50.

Петя. Много.

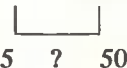
Вася. 50 — много. Значит, ты задумал число между 1 и 50.



Возьму для пробы снова число посередине: 25.

Петя. Мало.

Вася. Значит, ты задумал число между 25 и 50.



Теперь примерная середина — число 40.

Петя. Мало.

Вася. Ну, тогда твое число лежит между 25 и 40. 32?

Петя. Мало.

Вася. Надо искать между 32 и 40. 36?

Петя. Много.

Вася. 34?

Петя. Много.

Вася. Ты задумал число 33.

Петя. Да, ты угадал.

Вы заметили, как Вася угадывал? Он всегда старался взять сначала число по середине между известными границами. Таким образом, он каждый раз сужал отрезок поиска в 2 раза.

Давайте посчитаем, сколько ходов нужно при этом сделать. Сначала отрезок поиска был длиной 100. После первого вопроса он стал равен 50, после второго — 25 и т. д. (см. табл.)

Номер хода	Длина отрезка поиска
в начале	100
1	50
2	25
3	13
4	7
5	4
6	2
7	1

Итак, этим способом число между 1 и 100 можно отгадать самое большее за 7 ходов.

А если бы Вася предлагал числа как попало или по порядку: 1, 2, 3, 4, ...?

Ясно, что ему могло потребоваться очень много ходов, почти 100, если бы сильно не везло.

Способ, которым Вася отгадывал число Пети, называется методом деления пополам.

§ 2.10. Шаги алгоритма — команды исполнителю

Котенку Ваське надоели мыши. Эти шkodливые создания дергали его за усы или, еще хуже, привязывали к хвосту какую-нибудь дрянь, стоило ему только заснуть после обеда.

Васька стал составлять план хитрой мышиной охоты. Сначала, подумал Васька, я пройдусь перед норкой и как бы между прочим скажу, что в кладовке появился *очень* вкусный сыр. Потом съем кусочек сыра и сделаю вид, что ложусь спать. Мыши будут очень злы и непременно захотят отомстить. Они подойдут ко мне, и я их схвачу!

Чтобы ничего не перепутать, Васька записал свой алгоритм на бумаге. Вот что у него получилось:

Алгоритм хитрой мышиной охоты

1. Сказать про сыр.
2. Съесть кусочек сыра.
3. Сделать вид, что уснул.
4. Ждать, пока придут мыши.
5. Хватать!

Как вы думаете, кто будет исполнять составленный Васькой план? Неизвестно, что получится из его затей, но каждый шаг алгоритма, конечно же, будет выполнять он сам.

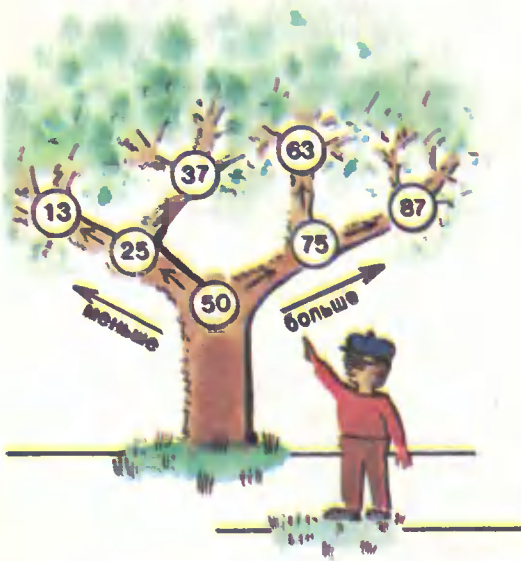
Значит, каждый шаг алгоритма хитрой мышиной охоты — это команда Ваське выполнить действие.

Всегда ли алгоритм исполняет тот, кто его составил?

Вы составляли алгоритм перевоза через реку. Шаги алгоритма в этом случае были командами для Перевозчика. Исполняя эти команды, Перевозчик трудился на своей лодке.

Исполнителем шагов-команд алгоритма переливания воды был робот Переливашка.

Можно сделать важный вывод: каждый шаг алгоритма — это команда исполнителю.



Моделирование правил орфографии

...мы, учителя русского языка, вправе ждать публикаций, раскрывающих приемы, методы алгоритмизации по частным вопросам грамматики.

Ю. Львова
(ИНФО. 1988. № 5. С. 121)

В статье пойдет речь о компьютерном моделировании орфографических правил в основном применительно к преподаванию русского языка в средней школе. Ее материалы могут быть также использованы на уроках информатики и при изучении орфографических тем в вузах и на курсах повышения квалификации учителей-словесников.

Модели (алгоритмы), которые будут рассмотрены, предназначены для работы с компьютерной моделирующей системой СИМВОЛ, описанной ранее (см.: Манзюра В., Шмелев Д. Система для учебного моделирования языковых явлений // ИНФО. 1989. № 1. С. 66—70). Таблицы, отражающие обработку языкового материала компьютером, строятся по тем же принципам, что и в упомянутой статье. Все модели пронумерованы, озаглавлены и приводятся компактно в конце статьи, а в основном тексте даны отсылки к моделям. Характер представления моделей позволяет использовать их и как средства наглядности при безмашинном обучении.

Компьютерное моделирование с помощью системы СИМВОЛ очень эффективно при изучении и обобщающем повторении правил русской орфографии, причем во многих случаях учащиеся способны сами составить алгоритмы изучаемых явлений и подобрать материал для их проверки, т. е. осуществить по отношению к компьютеру роль учителя, который сначала объясняет материал, а затем проверяет его усвоение. Практика показывает, что такая работа значительно повышает интерес к языку и сознательность изучения правил. Как следствие повышается внимание к написанному слову и улучшается грамотность.

Моделировать на ЭВМ можно очень многие правила русской орфографии. Орфографические модели допускают

достаточное разнообразие типов, что дает возможность не только закрепить навыки правописания, но и показать системные связи орфографии с фонетикой, словообразованием, морфологией.

Многие правила орфографии могут быть представлены моделями, описывающими выбор необходимой буквы или сочетания букв в зависимости от буквенного окружения, определяемого правилами («перед такими-то буквами или после таких-то букв пиши то-то»). При работе с такими моделями на вход ЭВМ подаются слова со звездочкой вместо «сомнительной» буквы, а машина эту букву «вставляет».

Учащимся вполне доступно составление модели, позволяющей правильно выбрать букву на конце приставок на -З (БЕЗ-, ВЗ-, РАЗ- и т. д.). Существуют разные варианты этой модели. Мы приведем вначале один из них, позволяющий помимо правильной записи приставок на -З отличить от них не имеющую орфографических вариантов приставку С- (см. модель М1).

Изучение приставок на -З, и приставки С- уместно связать с повторением правописания Ъ после приставок на согласную. Для этого модель М1 надо несколько усложнить (см. М2). Возможно, шестиклассникам понадобится в этом помощь учителя. Как пишет известный педагог и методист Ю. Л. Львова, «алгоритмы, составленные совместно учителем и учащимся, наиболее эффективны» (ИНФО. 1988. № 5. С. 121).

Дидактическая ценность М1 и М2 состоит, в частности, в том, что, составляя модель, дети скрупулезно повторяют условия, в которых применяется то или иное правило: для твердого знака это буква Е, Ё, Ю, Я, перед которыми он употребляется (см. обозначение 1 в М2), для выбора буквы С в приставках на -З — глухие согласные, список которых приводится полностью (обозначение 2 в М2).

Модель М3 отражает правила выбора букв Е—И в корнях с чередованием (-БЕР- — -БИР-, -ПЕР- — -ПИР-

и т. п.) Даже учащиеся V кл. в силах построить ее самостоятельно. Ценность таких несложных моделей состоит в том, что на них легко учиться основным принципам моделирования. Отметим, что именно наличие моделирующей системы позволяет в подобных случаях приобщить к активному управлению компьютером учащихся младших классов.

Интересно промоделировать правило о правописании Н и НН в суффиксах отыменных прилагательных (М4). В модели необходимо ввести исключения (см. формулы 1—4). При пользовании моделью нельзя забывать, что она рассчитана только на имена прилагательные, образованные от существительных с помощью суффиксов -ЕНН-, -ОНН-, -ЯН- (после шипящих -АН-), -ИН-. Исключения расположены раньше правил, так как в противном случае при анализе слов-исключений компьютер применит к ним основное правило и на выходе получатся неверно написанные слова.

По принципу «вставки букв» нетрудно построить модели для правил о разделительных знаках Ъ и Ь, буквы Ы—И после приставок, буквах Щ—Ч в суффиксе -ЩИК- (-ЧИК-) имен существительных.

Интересны модели, описывающие вставку не отдельной буквы, а значимой единицы языка — например, суффикса. Модель такого типа (М5) мы предлагаем для внеклассной работы учащихся, проявляющих интерес к компьютерному исследованию русского языка.

Дети нередко делают ошибки при выборе суффиксов -ЕНН- — -ЯН-, которым посвящена М5: пишут «серебряный», «соломьяный» и т. п. Правило различения этих суффиксов в школе не изучается. Предлагаем следующую его формулировку:

«Безударный суффикс -ЯН- (после шипящих -АН-) пишется перед ударными окончаниями прилагательных (ДОЩАНОЙ, ДРОВАНОЙ, НЕФТЯНОЙ и т. п.), а также в словах ГЛІНЯНЫЙ, КՔЖАНЫЙ, МАСЛЯНЫЙ (пятно, краски, насос и т. п.), СЕРЕБРЯНЫЙ и некоторых других, очень редких (приводим их для учителя: НИТЯНЫЙ, ПАКЛЯНЫЙ, СЕЛИТ-

РЯНЫЙ, СՔСЛЯНЫЙ, ФИНИФТЯНЫЙ). В прочих случаях пишется безударный суффикс -ЕНН- (МЫСЛЕННЫЙ, ТОРЖЕСТВЕННЫЙ, ФИРМЕННЫЙ и т. п.)».

Строя модель, необходимо ввести обозначение для шипящих Ж, Ш, Щ, после которых употребляется вариант -АН- (шипящая Ч перед безударным суффиксом -АН- не встречается). Ограничив модель обработкой только начальных форм прилагательных, можно отказаться от обозначения ударения (оно автоматически передается вариантами окончаний -ЫЙ- — -ОЙ). Не забудем также, что в прилагательном ВЕТРЕННЫЙ суффикс -ЕН- передается с одной буквой Н. Введем это слово в состав исключений. Слова для обработки будем задавать со звездочкой вместо суффикса.

Следующий тип орфографической модели позволяет определить наличие или отсутствие «сомнительной» буквы. Такова модель М6, отражающая правило о правописании Ь в форме родительного падежа множественного числа существительных на -НЯ (НЯНЯ — НЯНЬ, ВИШНЯ — ВИШЕН). Оно также не изучается по школьной программе, и его моделирование может быть дано как индивидуальное задание интересующимся ученикам. Правило известно в двух формулировках.

Первая из них ориентирована непосредственно на форму родительного падежа множественного числа: если она оканчивается на -ЕН, то Ь не пишется (БАШЕН, ВИШЕН и т. п.). Исключение: БАРЫШЕНЬ, БОЯРЫШЕНЬ, ДЕРЕВЕНЬ. В прочих случаях Ь необходим (ДЫНЬ, НЯНЬ и т. п.). При моделировании правила в этой формулировке удобно вводить исходные формы с вопросительным знаком на конце, а в модели отразить исключение этого знака или замену его на Ь.

Переходя ко второй формулировке правила, мы одновременно переходим и еще к одному типу орфографической модели, которая на этот раз тесно связана с морфологией.

Вторая формулировка учитывает очевидность конечного сочетания -ЕН в безударном положении и исходит из начальной формы слова: если в ней перед

-НЯ есть согласная, то Ъ в родительном падеже множественного числа не пишется (БАШНЯ — БАШЕН, ПЕСНЯ — ПЕСЕН и т. п.). К исключениям добавляется слово КУХНЯ — КУХОНЬ. Модель, соответствующая этой формулировке, интересна тем, что строит косвенную форму слова по начальной (см. М7). Если к этой модели добавить еще 2-формулы, то можно образовать орфографически правильные уменьшительные формы от исходных слов на -НЯ. Формулы эти таковы:

9. Н# — → НКА#

10. Ъ# — → ЪКА#

Компьютер образует формы: ДЕРЕВЕНЬКА, КУХОНЬКА, КОФЕЕНКА, СПАЛЕНКА, ВИШЕНКА, НЯНЬКА.

При изучении правописания Ъ после шипящих на конце начальных форм существительных (МЯЧ — РЕЧЬ) необходимо опираться на род, т. е. проводить неформальный анализ слов, на который компьютер не способен. Однако можно заставить машину выводить начальные формы из косвенных. Удобно в этом случае исходить из формы творительного падежа единственного числа, где существительные женского рода также содержат Ъ — правда, в «разделительной» роли (НОЧЬЮ — НОЧЬ). Обращение к моделированию упомянутого правила уместно при изучении правописания О—Е после шипящих в окончаниях существительных (V класс по теперешней нумерации), когда учащиеся уже умеют правильно записать формы типа ЛУЧОМ, ТОВАРИЩЕМ. Таким образом, закрепляется новое правило и попутно повторяется правило о Ъ в начальной форме (см. модель М8). Модель будет особенно эффективной, если компьютер рядом с начальными формами слов проставит их род: ПОМОЩЬ — Ж., МЯЧ — М., МАТЧ — М. Сделать это нетрудно: например, формула 2 в М8 примет вид: БЮ# — → Ъ — Ж.#.

Правило об отсутствии Ъ после шипящих в формах родительного падежа множественного числа имен существительных моделируется элементарно (см. М9), но, кто знает, может быть, эта простота позволит детям не только

самостоятельно построить модель, но и твердо запомнить правило?

В следующей модели (М10) сочетаются принципы вставки букв и образования одной формы от другой. При моделировании правила о гласных в суффиксах существительных -ЕК и -ИК удобно, как и в М8, обратиться к «выведению» компьютером начальной формы слова из косвенной. За исходную можно принять форму родительного падежа единственного числа, в которой будем заменять звездочкой «сомнительную» гласную.

Компьютерные модели могут использоваться и для того, чтобы дать своеобразное графическое объяснение тех или иных написаний. Таким объяснением может быть, к примеру, частичное обозначение морфемного состава слова. Этот прием позволяет уяснить правописание слов с непроизносимыми согласными. Слова вводятся в обычный орфографическом облике, а на выходе представлены расчлененными по составу. Модель М11 показывает, как можно этого добиться.

Модель такого типа нетрудно построить и для анализа глаголов на -ТСЯ — -ТЬСЯ, причем относительная простота модели достигается за счет того, что основа инфинитива или настоящего (простого будущего) времени не членится на морфемы (в данном случае это оправдано и методически, так как различие основ здесь важнее, чем особенности их строения). На выходе получаются глагольные формы в следующем виде: САДИ-ТЬ-СЯ — САД-ИТ-СЯ, РАССМЕЯ-ТЬ-СЯ — РАССМЕ-ЮТ-СЯ, ПОСЛЫША-ТЬ-СЯ — ПОСЛЫШ-АТ-СЯ и т. п.

Итак, компьютерные орфографические модели достаточно разнообразны и охватывают значительное количество правил русского правописания. Не менее разнообразна и методика работы с моделями, которая заслуживает особого разговора. Широкий диапазон применения моделирования при изучении орфографии позволяет осуществлять индивидуальный подход к учащимся. Модели, созданные учениками на основе изучения правил, не входящих в школьную программу, и соответствующего

языкового материала, могут быть защищены на научно-практических конференциях школьников. Все удачные, хоро-

шо работающие модели могут использоваться в качестве компьютерных справочников.

Модели

М1. Конечная согласная в приставках на -З и приставках на -С.

Обозначение

+ : КПСТФХЦЧЩ (т. е. все глухие согласные)

ПРАВИЛА	*ДЕЛАТЬ	*ЩЕПИТЬ	ВО*ДЕЛАТЬ	РА*ЩЕПИТЬ
1. #* → #С	СДЕЛАТЬ	ЩЕПИТЬ	—	—
2. *+ → С	—	—	—	РАСЩЕПИТЬ
3. * → З	—	—	ВОЗДЕЛАТЬ	—
	СДЕЛАТЬ	ЩЕПИТЬ	ВОЗДЕЛАТЬ	РАСЩЕПИТЬ

М2. Конечная согласная в приставках на -З и приставках на С- и употребление Ъ после этих приставок.

Обозначения

J : ЕЁЮЯ

+ : КПСТФХЦЧЩ (т. е. все глухие согласные)

ПРАВИЛА	*ЕЗД	*ДАТЬ	БЕ*ЩЕННЫЙ	БЕ*ЯДЕРНЫЙ	БЕ*МЕРНЫЙ
1. #*J → #СЬ	СЬЕЗД	—	—	—	—
2. #* → #С	—	СДАТЬ	—	—	—
3. *+ → С	—	—	БЕСЩЕННЫЙ	—	—
4. *J → ЗЬ	—	—	—	БЕЗЪЯДЕРНЫЙ	—
5. * → З	—	—	—	—	БЕЗМЕРНЫЙ
	СЬЕЗД	СДАТЬ	БЕСЩЕННЫЙ	БЕЗЪЯДЕРНЫЙ	БЕЗМЕРНЫЙ

М3. Буквы Е — И в корнях с чередованием.

ПРАВИЛА

ПРАВИЛА	ЗАП*РАТЬ	РАССТ*ЛАЕТСЯ	ЗАП*Р	РАССТ*ЛИЛСЯ
1. *РА → ИРА	ЗАПИРАТЬ	—	—	—
2. *ЛА → ИЛА	—	РАССТИЛАЕТСЯ	—	—
3. * → Е	—	—	ЗАПЕР	РАССТЕЛИЛСЯ
	ЗАПИРАТЬ	РАССТИЛАЕТСЯ	ЗАПЕР	РАССТЕЛИЛСЯ

М4. Буквы Н и НН в суффиксах отыменных прилагательных.

Обозначения

I : ИЯА

⊘ : ЕО

ПРАВИЛА

ПРАВИЛА	КОЖА*ЫЙ	ДЕРЕВЯ*ЫЙ	ВЕТРЕ*ЫЙ	БЕЗВЕТРЕ*ЫЙ	ЛЕБЕДИ*ЫЙ
1. СТЕКЛЯ* → СТЕКЛЯНН	—	—	—	—	—
2. ОЛОВЯ* → ОЛОВЯНН	—	—	—	—	—
3. ДЕРЕВЯ* → ДЕРЕВЯНН	—	ДЕРЕВЯННЫЙ	—	—	—
4. #ВЕТРЕ* → # ВЕТРЕН	—	—	ВЕТРЕННЫЙ	—	—
5. I* → Н	КОЖАНЫЙ	—	—	—	ЛЕБЕДИНЫЙ
6. ⊘* → НН	—	—	—	БЕЗВЕТРЕННЫЙ	—
	КОЖАНЫЙ	ДЕРЕВЯННЫЙ	ВЕТРЕННЫЙ	БЕЗВЕТРЕННЫЙ	ЛЕБЕДИНЫЙ

М5. Выбор безударного суффикса -ЯН-(-АН-) — -ЕНН- имен прилагательных

Обозначения

W : ЖШЦ

ПРАВИЛА:

ПРАВИЛА:	ЛАПШ*ОЙ	ГЛИН*ЫЙ	ВЕТР*ОЙ	ВЕТР*ЫЙ	УТР*ЫЙ
1. W*ой → АНОЙ	ЛАПШАНОЙ	—	—	—	—
2. *ой → ЯНОЙ	—	—	ВЕТРЯНОЙ	—	—
3. ГЛИН* → ГЛИНЯН	—	ГЛИНЯНЫЙ	—	—	—
4. КОЖ* → КОЖАН	—	—	—	—	—
5. МАСЛ* → МАСЛЯН	—	—	—	—	—
6. СЕРЕБР* → СЕРЕБРЯН	—	—	—	—	—
7. #ВЕТР* → # ВЕТРЕН	—	—	—	ВЕТРЕННЫЙ	—
8. * → ЕНН	—	—	—	—	УТРЕННИЙ
	ЛАПШАНОЙ	ГЛИНЯНЫЙ	ВЕТРЯНОЙ	ВЕТРЕННЫЙ	УТРЕННИЙ

М6. Употребление Ъ в формах родительного падежа множ. числа существительных на -НЯ.

ПРАВИЛА	ДЕРЕВЕН?	ВИШЕН?	КУХОН?	НЯН?
1. БАРЫШЕН? --> БАРЫШЕНЬ	—	—	—	—
2. БОЯРЫШЕН? --> БОЯРЫШЕНЬ	—	—	—	—
3. ДЕРЕВЕН? --> ДЕРЕВЕНЬ	ДЕРЕВЕНЬ	—	—	—
4. ЕН? --> ЕН	—	ВИШЕН	—	—
5. ? --> Ъ	—	—	КУХОНЬ	НЯНЬ
	ДЕРЕВЕНЬ	ВИШЕН	КУХОНЬ	НЯНЬ

М7. Образование формы родительного падежа множ. числа существительных на безударное -НЯ.

Обозначение
S : БВГДЖЗПРСТЧШ

ПРАВИЛА	ДЕРЕВНЯ	КУХНЯ	КОФЕЙНЯ	СПАЛЬНЯ	ВИШНЯ	НЯНЯ
1. БАРЫШНЯ --> БАРЫШЕНЬ	—	—	—	—	—	—
2. БОЯРЫШНЯ --> БОЯРЫШЕНЬ	—	—	—	—	—	—
3. ДЕРЕВНЯ --> ДЕРЕВЕНЬ	ДЕРЕВЕНЬ	—	—	—	—	—
4. КУХНЯ --> КУХОНЬ	—	КУХОНЬ	—	—	—	—
5. ИНЯ --> ЕН	—	—	КОФЕЕН	—	—	—
6. ЪНЯ --> ЕН	—	—	—	СПАЛЕН	—	—
7. ШНЯ --> ЕН	—	—	—	—	ВИШЕН	—
8. НЯ# --> НЬ#	—	—	—	—	—	НЯНЬ
	ДЕРЕВЕНЬ	КУХОНЬ	КОФЕЕН	СПАЛЕН	ВИШЕН	НЯНЬ

15

М8. Образование начальной формы существительных мужского и женского рода с основой на шипящую от формы творительного падежа ед. числа

ПРАВИЛА	ДОЧЕРЬЮ	ПОМОЩЬЮ	МЯЧОМ	МАТЧЕМ
1. ДОЧЕРЬЮ --> ДОЧЬ	ДОЧЬ	—	—	—
2. БЮ# --> Ъ#	—	ПОМОЩЬ	—	—
3. ОМ# --> #	—	—	МЯЧ	—
4. ЕМ# --> #	—	—	—	МАТЧ
	ДОЧЬ	ПОМОЩЬ	МЯЧ	МАТЧ

М9. Образование формы родительного падежа множ. числа существительных с основой на шипящую от начальной формы.

ПРАВИЛА	ЗАДАЧА	УЧИЛИЩЕ	ПЛЕЧО
1. А# --> #	ЗАДАЧ	—	—
2. Е# --> #	—	УЧИЛИЩ	—
3. О# --> #	—	—	ПЛЕЧ
	ЗАДАЧ	УЧИЛИЩ	ПЛЕЧ

М10. Гласные в суффиксах существительных -ЕК, -ИК (образование начальной формы от формы родительного падежа ед. числа)

ПРАВИЛА	КЛЮЧ*КА	ПЕРЕШЕЙКА	ЗАМОЧКА
1. *КА# --> ИК#	КЛЮЧИК	—	—
2. ЯКА# --> ЕК#	—	ПЕРЕШЕЕК	—
3. КА# --> ЕК#	—	—	ЗАМОЧЕК
	КЛЮЧИК	ПЕРЕШЕЕК	ЗАМОЧЕК

М11. Состав слов с непроизносимыми согласными и отсутствием их.

Обозначения

У : ИОЫ

З : ЙК

ПРАВИЛА	АГЕНТСТВО	КРИТИКАНСТВО	ГИГАНТСКИЙ	КОЛХОЗНИК	СЛОВЕСНЫЙ
1. СТВО# --> -СТВ-О#	АГЕНТ-СТВ-О	КРИТИКАН-СТВ-О	—	—	—
2. СКЙ# --> СК-ИЙ#	—	—	ГИГАНТ-СК-ИЙ	—	—
3. СКОЙ# --> СК-ОЙ#	—	—	—	—	—
4. ЗДН --> ЗД-Н-	—	—	—	—	—
5. СТН --> СТ-Н-	—	—	—	—	—
6. ЗНУЗ --> З-Н-	—	—	—	КОЛХОЗ-Н-ИК	—
7. СНУЗ --> С-Н-	—	—	—	—	СЛОВЕС-Н-ЫЙ
	АГЕНТ-СТВ-О	КРИТИКАН-СТВ-О	ГИГАНТ-СК-ИЙ	КОЛХОЗ-Н-ИК	СЛОВЕС-Н-ЫЙ

Демонстраторы помогают учиться программированию

У учащихся младших классов средней школы слабо развито абстрактное математическое мышление. В основном дети этого возраста более склонны играть в игры, нежели решать задачи, и это естественно. Поэтому содержание урока с использованием игровых элементов усваивается и запоминается более эффективно, чем традиционного.

Мы убеждены, что введение формальных определений, использование огромных формул и т. п. в младших классах может вызвать у учащихся нежелательное ощущение непонимания предмета и как следствие породить негативное отношение к вычислительной технике и программированию.

Экспериментальная работа, которую проводят авторы, основывается на том, что игровая мотивация рассматривается как средство обучения младших школьников основам программирования. Это не означает, что учащиеся играют в компьютерные игры, получившие распространение в нашей стране и за рубежом. Речь идет о качественно новом методе преподавания программирования. Его сущность сводится к формированию навыков анализа решаемых задач, свойственных профессиональным программистам.

С этой целью авторами разработано программно-методическое обеспечение курса обучения началам программирования. Курс адаптирован к операционным системам RSX 11 ЭВМ СМ-4 и ФОДОС ЭВМ УКНЦ для черно-белых алфавитно-цифровых дисплеев и ПЭВМ РС/ХТ в цветном варианте. Все программы написаны на языке Бейсик.

Структурно курс состоит из трех частей, каждая из которых поддержана комплексом программ, методикой и задачиком с набором постепенно усложняющихся задач. Каждая часть называется Демонстратором и реализована как совокупность исполнителей, объединенных общим интерфейсом и направленностью на поддержку опре-

деленной педагогической методики. Таким образом, весь курс состоит из трех Демонстраторов, условно названных «Художник», «Танк» и «Лабиринт».

Демонстратор «Художник»

В статье описывается первая часть — Демонстратор «Художник». В его основу, как и в языке ЛОГО, положен принцип визуализации действий абонента за дисплеем с помощью «следов», оставляемых Черепашкой, ползущей по экрану. «Художник» рассматривается как единство набора средств для создания на экране дисплея рисунков различной формы, качества, сложности; методик обучения и совокупности решаемых задач как необходимой компоненты познания. Основной идеей функционирования программы-Демонстратора является реализация возможности для учащихся «заглянуть внутрь» акта создания рисунков.

Сравнивая методики обучения школьников с помощью Демонстратора «Художник» с методиками, используемыми при изучении языка ЛОГО, можно выделить следующие отличительные черты.

1) Предлагаемая авторами методика может быть реализована на алфавитно-цифровых дисплеях.

2) Демонстратор «Художник» использует язык, приближенный к естественному русскому языку.

3) Наличие в Демонстраторе понятия «след» и возможности его изменения позволяет строить картинки негеометрического графического представления.

4) Программно поддерживаются необходимые интерактивный режим и режим интерпретации.

5) Реализована возможность мультиплективного, т. е. посимвольного, высвечивания картинки на экране дисплея.

Изображения, которые получаются при работе с Демонстратором «Художник», сильно отличаются от изображений, строящихся с помощью языка ЛОГО. Они менее «геометричны», более

эмоционально выразительны, поскольку «следом», оставляемым Черепашкой, может быть не только черточка, но и крестик, звездочка и вообще любой символ клавиатуры, имеющий графическое представление.

В языке Демонстратора использованы группы команд: а) группа управления перемещением курсора дисплея вправо, влево, вверх, вниз; б) команды изменения состояния — символа «следа», оставляемого на экране дисплея при перемещении «кончика кисти «Художника»; в) вспомогательные команды — высвечивание написанной программы, восстановление рисунка по этой программе, гашение экрана, вызов системы коррекции и возврат из нее, установка начального состояния (гашение экрана и уничтожение программы); г) команды цикла, организации подпрограмм и обращений к ним (блоки); д) команды системы посимвольной коррекции.

Функция объекта конструктивной деятельности сводится не только к созидательным действиям по построению различных картинок, но и к строгому протоколированию всех проводимых учащимся действий в их реальной последовательности (т. е., рисуя «картинку», учащийся составляет, сам того не осознавая, программу построения изображения, которая запоминается в памяти ЭВМ). Это позволяет установить взаимно однозначное соответствие между действиями учащихся и возникающих в результате рисунков и их деталей. По заявке учащегося «Художник», снабженный специальной подпрограммой, может продемонстрировать всю последовательность действий, проведенных учащимся в процессе решения задачи, и тем самым «показать» алгоритм составления рисунка, а также программу его реализации. Такая же программа может быть построена и на других дисплеях и, конечно, породит на экране такой же рисунок. Так осуществляется привязка алгоритмизации и программы к объектам, создающимся буквально на глазах учащегося. Эти объекты постепенно ассоциируются со способами их создания, т. е. с построением программ.

«Художник» снабжен подпрограммой, способной по заданной программе

восстановить рисунок. Само восстановление осуществляется двумя способами: в режиме быстрого вывода на дисплей рисунка построчно и в режиме медленного вывода — побайтно. Второй способ особенно нравится детям, поскольку напоминает мультипликационные построения.

Методически обучение осуществляется таким образом, что степень сложности конструируемых рисунков повышается в зависимости от усвоения изучаемого материала. В начале учащиеся обучаются управлению «кистью «Художника» на простейших рисунках — построение изображений различных геометрических линий (рис. 1). Затем сложность картин повышается. Изучаются понятия алгоритма, простейшей коррекции (удаление последнего оператора). На рис. 2 дан набор задач, которые должны быть решены учащимися с использованием техники формирования циклов.

17

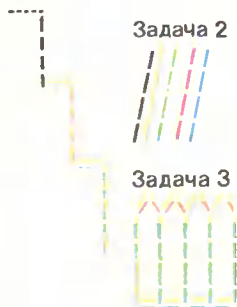


От геометрических построений дети переходят к построению реально существующих предметов (деревья, дома, цветы). Здесь приходится искусно изменять оставляемый кистью «след». При этом алфавитно-цифровые дисплеи оказываются в некотором смысле выразительнее графических. Поскольку эта работа проводится в интерактивном режиме, учащиеся поставлены в условия, при которых они вынуждены добиваться результата с помощью ряда последовательных операций, т. е. самостоятельно проводить алгоритмизацию задачи.

На следующем этапе решается задача обучения синтезу сложных картинок из набора отдельных деталей. Важнейшим аппаратом для решения этих задач оказывается структурное программирование и, в частности, формирование подпрограмм. Представленные на рис. 3 задачи поясняют, как учащиеся постепенно овладевают аппаратом подпрограмм (блоков) сначала для оформления повторяющихся деталей, а затем и для разбиения сложной задачи на простые части.

3

Задача 1



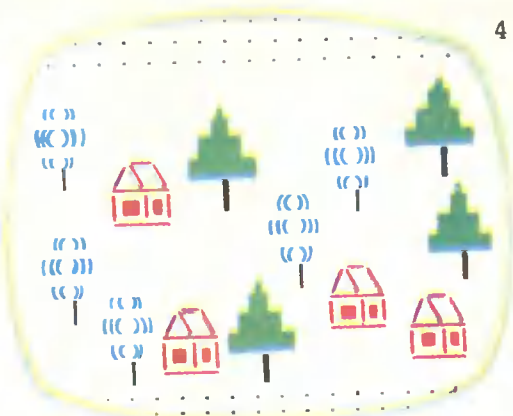
Задача 4



Задача 5



Картинками, реализуемыми с использованием аппарата подпрограмм, могут стать: лес — как совокупность деревьев, деревня — дома и деревья, цветочный луг и т. п. При этом нет необходимости точно определять положение отдельных деталей, достаточно фиксировать их общее количество. Естественно, что при такой методике на разных дисплеях получатся разные рисунки. Это различие достигается за счет варьирования межблочных связей (поскольку используются одни и те же блоки).



Картинка на рис. 4 состоит из компонент: домик, лиственное дерево и елочка. Каждая из них оформлена как самостоятельная подзадача и реализуется в виде подпрограммы, что позволяет любую из компонент изобразить в произвольном месте экрана. На этом этапе приходится отказаться от интерактивного режима и перейти в режим интерпретации.

Работая с Черепашкой, учащиеся не сразу понимают, что на самом деле происходит сложная работа по отображению алгоритма создания рисунка на язык «Художника», а только затем интерпретатор полученной программы выдает на экран дисплея изображение. В таком познании им, конечно, поможет преподаватель, который на определенном этапе изучения программирования расскажет учащимся об «исполнении» и «интерпретации».

Далее учащихся обучают основам разбиения крупных задач на мелкие подзадачи и программному оформлению их в виде независимых блоков. Содержательно такие картинки существенно отличаются от строящихся на предыдущих этапах. Это может быть многорукый робот, бригантина, ковер со сложным орнаментом и т. д., т. е. законченная картина, занимающая почти весь экран дисплея. Их особенностью является отсутствие или очень малое количество повторяющихся деталей, что осложняет разбиение на блоки и тем самым служит развитию логического мышления.

Для этой цели Демонстратор «Художник» снабжен серией программ, по-

звolyающих дробить на части задачу и рисовать эти части — блоки в произвольной последовательности, чтобы затем соединять их, добываясь нужного результата. Одновременно Демонстратор позволяет вводить теоретические понятия блока, подпрограммы, межблочных связей естественным способом, как нечто само собой разумеющееся. Демонстратор вырабатывает навыки к анализу больших и трудных рисунков, продвижению к решению постепенно, поэтапно.

Такой подход позволяет вводить понятия «оператор», «алгоритм», «программа» — трудных, почти философских понятий — без особых затрат времени на их специальное изучение. Эти понятия вводятся естественно, поскольку они «видимы».

При этом учащиеся обучаются с желанием выслушать учителя, который подскажет, как рисовать все более и более сложные рисунки.

В процессе преподавания тщательно поддерживается соревновательный дух. Оценки ставит «Художник» по количеству операторов, затрачиваемых на реализацию программы, и каждый ученик стремится минимизировать эти затраты. Мы, конечно, понимаем, что такая оценка не всегда объективна. Она может

не оценить красивую задумку учащегося при создании некоторого фиксированного рисунка, но учитель, заметив возможную недооценку труда, может исправить это положение. Учащимся кажется, что «Художник», сам выставляющий оценки, весьма объективен. Тот, кто нашел действительно красивое программистское решение, обязательно будет похвален и расскажет о своей находке всему классу. Похвалы будет удостоено и то, что может стать трамплином к рассказу о некоторых новых понятиях и особенностях программирования. Вот почему необходимо поддержать ученика, который продемонстрировал некоторый нестандартный подход к реализации, понимание каких-то глубинных свойств структуры и особенности рисунка, предложенного в качестве задачи.

На этом завершается первая, вводная часть курса. Учащиеся получили необходимые знания и умения для перехода на новую, более высокую ступень, поддерживаемую другими Демонстраторами*.

* От авторов. Заинтересованные организации могут обратиться за справками по адресу: 117312, Москва, проспект 60-летия Октября, дом 9, ВНИИСИ АН СССР, комната 618, Акимовой Г. П., Деза В. Н.

К. СУЛИМА-САМОЙЛОВ

Киевское суворовское военное училище

Опорные конспекты в курсе ОИВТ

Изучение начального уровня основных понятий основ информатики и вычислительной техники в нашем училище проводилось с применением листов опорных сигналов (ЛОС) Т. Н. Поддубной*. Эффективность использования ЛОС привела к разработке комплекта опорных конспектов (ОК) на базе ЛОС, которые были дополнены изображениями основных устройств ЭВМ, установленных в кабинетах училища, а также синтаксическими диаграммами из серии уроков заочной школы программирования Г. А. Звенигородского и его коллег

(«Квант», 1979—1982 гг.). ОК охватывают все разделы курса ОИВТ.

ОК не заменяют учебное пособие, а дополняют его, позволяя наглядно выделить основные понятия, приемы построения алгоритмов простейших задач, помогают учащимся в самостоятельной работе с учебными пособиями и при работе на ЭВМ. При подготовке к работе учащихся с ОК необходимо, исходя из содержания ОК, сформулировать требования к знаниям и умениям, подобрать вопросы и упражнения.

ОК способствуют активному усвоению учащимися основ информатики и вычислительной техники, развитию у них логи-

* См.: Информатика и образование. 1987. № 3.

ческого мышления, компьютерной грамотности и, что очень важно, побуждают интерес к информатике. Каждый учащийся имеет комплект ОК.

Каждый ОК снабжен вопросами и упражнениями. Упражнения сочетаются с выполнением программ, записанных на дискетах.

На практических занятиях учащиеся выполняют весь комплекс работ, необходимых при работе на ЭВМ: загружают операционную систему, просматривают каталоги дискет, инициализируют работу интерпретатора Бейсика, загружают и просматривают на экране дисплея программу из файла по указанию преподавателя и выполняют заданные упражнения.

Для изучения основ языка Бейсик составлен комплект из шести ОК:

1. Ввод-вывод. Присваивание.
2. Управление в программе.
3. Величины.
4. Функции. Подпрограммы.
5. Управление в программе (продолжение).
6. Литерные величины.

В статье подробно описывается методика использования ОК-1. ОК-4 приведен с вопросами и упражнениями.

Для ОК-1 были сформулированы следующие требования к знаниям и умениям.

Учащиеся должны знать:

назначение и основные технические данные всех основных устройств ЭВМ; правила техники безопасности при работе на ЭВМ;

физический смысл операции присваивания;

операторы ввода-вывода, присваивания.

Учащиеся должны уметь:

начинать и завершать работу на ЭВМ; пользоваться клавиатурой и курсором;

использовать в режиме диалога простую программу, содержащую операторы ввода и вывода в непосредственном (режиме калькулятора) и программном режиме.

К первому практическому занятию с ОК-1 необходимо ознакомить учащихся с историей языка Бейсик, содержанием вопросов 4—8. Нужно задать им по-

вторить по учебному пособию и конспектам назначение и основные технические характеристики ЭВМ и ее центральных устройств (вопросы 1—3); предложить записать несколько арифметических выражений из учебника математики.

Урок следует начать с объяснения физического смысла операции присваивания (помещения значения переменной в ячейки памяти, которым присвоено имя данной переменной, т. е. имя переменной — это адрес, по которому хранится текущее значение переменной) и по ОК-1 показать как в языке Бейсик осуществляются операции присваивания, ввода и вывода. Во время рассказа преподавателя учащиеся «поднимают» красным цветом в заголовке ОК-1 слово «присваивание» и стрелки, идущие в ОЗУ от клавиатуры, магнитных дисков и процессора.

Преподаватель показывает на ЭВМ и объясняет порядок инициализации и режимы работы интерпретатора Бейсика.

На первом занятии учащиеся в режиме непосредственного исполнения (режим калькулятора) и программном выполняют упражнения а, б, в, г.

Завершается первое занятие просмотром каталога дискеты, загрузкой и распечаткой на экране дисплея содержимого файлов с программами.

Задание на самоподготовку

Работа с ОК-1, вопросы 8—12, упражнения II (1, 2, 3).

На втором занятии особое внимание следует уделить оператору PRINT, его возможностям при выводе информации на экран дисплея, дискету и принтер (АЦПУ). Здесь необходимо подробно рассмотреть с показом на ЭВМ назначение и возможности всех параметров оператора: пустой оператор, разделители параметров и их влияние на формат выводимой информации, а также возможности оператора подавить перевод маркера на следующую строку.

Учащиеся на втором занятии работают с упражнениями I (1, 2) и II (1, 2, 3, 4).

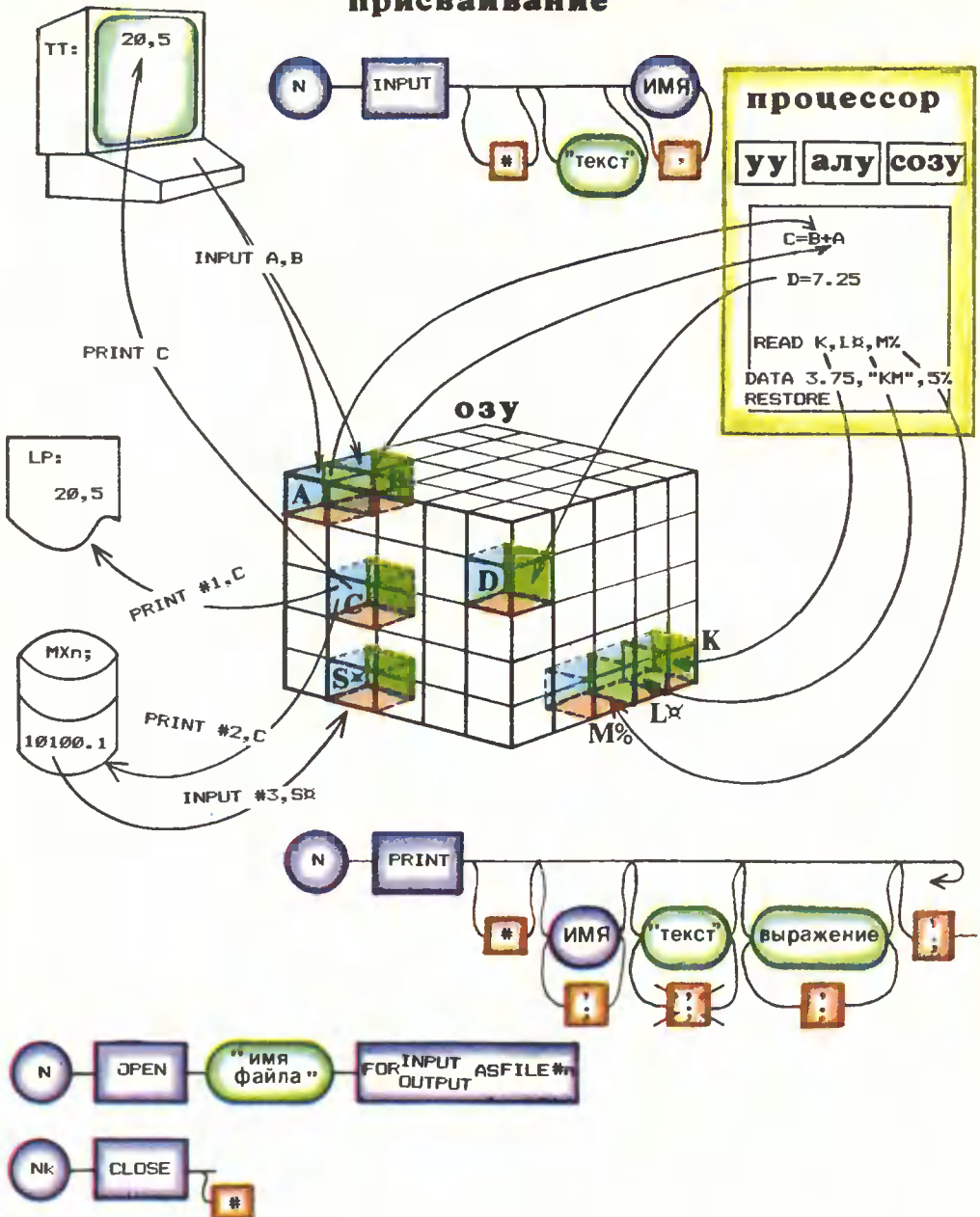
Задание на самоподготовку

Работа с ОК-1, вопросы 8—12, составление программы диалога с ЭВМ из четырех-пяти вопросов.

Вопросы к Бейсик ОК-1

1. Назначение и общее устройство ЭВМ.

присваивание



2. Назначение и общее устройство процессора.

3. Назначение и структура памяти ЭВМ.

4. Арифметическое выражение. Знаки и приоритет арифметических операций.

5. Особенности режимов непосредственного и программного исполнения.

6. Формат записи имени переменной и файла. Что такое файл?

7. Структура программы.

8. Физический смысл операции присваивания. Операторы присваивания.

9. Назначение и основные данные устройств ввода. Операторы ввода.

10. Назначение и основные данные устройств вывода. Операторы вывода.

11. Что произойдет, если в строке не хватит места для вывода очередного числа, константы?

12. Назначение и работа директив: LOAD, LIST, RUN, SYS.

В режиме непосредственного исполнения (калькулятора) вычислить: а) площадь круга и длину окружности при $R=8,25$; б) объем и поверхность правильного тетраэдра с ребром $L=5,97$; в) $\log_2 7$; г) $f(37,5)$, если $f(x) = \sin^2 x + x^2 + 10x - \text{ctg } x$.

Упражнения

I. На какое устройство и в каком виде будет получен результат выполнения следующих фрагментов программ:

```
1) 10 READ A,B,C
    20 READ D,E,F
    30 ? ,C,B,D
    40 PRINT F,E,A
    50 DATA 5,"БОЛЬШЕ"
    60 DATA -3,-9
    70 DATA "МЕНЬШЕ",4
```

```
2) 10 DATA 5,7,-9,4,8,-2
    20 READ A,B,C
    30 RESTORE
    40 READ X,Y,Z
    50 RESTORE
    60 READ K,K,K
    70 PRINT A,B,C,X+Y-Z,K
```

II. Написать программы. 1. Ввести с клавиатуры два вещественных числа и вывести на экран дисплея эти числа и их полусумму.

2. Ввести с клавиатуры три вещественных числа и вывести на экран дисплея эти числа в обратном порядке и их среднее арифметическое.

3. В программе четырем переменным присвоить значения: 4, 8, 16, 32 и вывести на экран дисплея по зонам эти числа и их среднее геометрическое.

4. Программу следующего диалога с ЭВМ:

ЭВМ	Учащийся
— «КАКОЕ	— вводит день,
СЕГОДНЯ ЧИСЛО	месяц и год;
— «ВАШЕ ИМЯ»	— вводит свое имя.

— «ЗДРАВСТВУЙТЕ (имя ученика) СЕГОДНЯ (день, месяц и год) МЫ ПРОДОЛЖАЕМ ИЗУЧЕНИЕ ЯЗЫКА БЕЙСИК».

Вопросы к Бейсик ОК-4

1. Какие стандартные функции имеет язык?

2. Что такое функции пользователя, формальные и фактические параметры?

3. Как именуются функции пользователя, каким оператором и когда они определяются?

4. Можно ли при определении функции пользователя в качестве формальных параметров использовать числовые константы?

5. Что такое подпрограмма и ее структура?

6. Структура и работа оператора вызова подпрограмм.

7. Допустимо ли в одной подпрограмме использовать несколько операторов?

8. Можно ли обратиться к подпрограмме оператором GOTO?

9. Сколько уровней вложения подпрограмм допускается?

10. Назначение и работа директив CONT, DEL, NEW.

Упражнения

I. Составьте блок-схему алгоритма и исполните следующие фрагменты программ:

```
1) 10 INPUT "ВВЕДИТЕ СЛОВО":A$
    20 C$=""
    30 B$=""
    40 FOR K=1 TO LEN(A$)
    50 B$=MID$(A$,K,1)+B$
    60 C$=C$+MID$(A$,K,1)
    70 NEXT K
    80 ? A$,B$,C$
    90 END
```

Исполните алгоритм для слова ИНФОРМАТИКА.

2)

```
10 DEF FNF(A,W,X,F,C)=A*W*SIN(W*X+F)+C
    20 INPUT "ВВЕДИТЕ ГРАНИЦЫ ОБЛАСТИ
           И ШАГ СП,К,Н";N,K,H
    30 X=N
    40 INPUT "ВВЕДИТЕ ПАРАМЕТРЫ ФУНКЦИИ:
           A,W,F,C";A,W,F,C
    50 ? TAB(FNF(A,W,X,F,C))"X"
    60 X=X+H
    70 IF X<=K GOTO 50
    80 END
```

Организуйте цикл при помощи оператора FOR—NEXT.

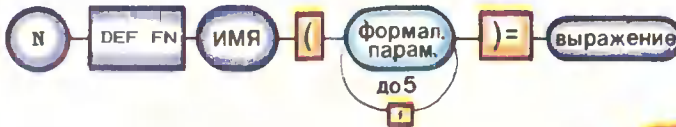
БЕЙСИК ОК4

ФУНКЦИИ

стандартные

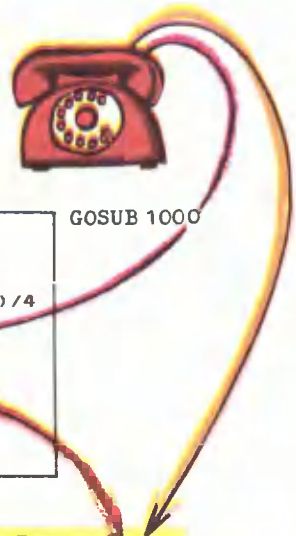
ЧИСЛОВЫЕ		литерные	
ABS(X) $ x $	LOG10(X) $\lg X$	ASK(X#)	код символа X#
ATN(X) $\arctg X$	RND(X) случайн. число	CHR(X)	Символ с кодом X
COS(X) $\cos X$	SIN(X) $\sin X$	LEN(X#)	Длина строки X#
EXP(X) e^x	SQR(X) \sqrt{x}	MID(X#,N,K)	Вырезка из X# K символов начиная с N-го
INT(X) $ x $	TAN(X) $\tg X$	STR(X)	X как символ
LOG(X) $\ln X$	TAB(X) вывод в X-позицию	VAL(X#)	Значение X#

пользователя



```
120 DEF FNE(A,W,X,F,C)=A*W*COS(W*X+F)+C
```

ПОДПРОГРАММЫ



GOSUB 1000

```
10 REM ФУНКЦИЯ FND - БОЛЬШЕЕ ИЗ ДВУХ ЧИСЕЛ
20 DEF FND(A,B)=(A+B+ABS(A-B))/2
30 REM ФУНКЦИЯ FNG - ФОРМУЛА ГЕРОНА
40 DEF FNG(A,B,C)=SQR(((A+B+C)*(A+B-C)*(A+C-B)*(B+C-A)))/4
50 PRINT "ВВЕДИТЕ ДЛИНЫ ОТРЕЗКОВ A, B, C";
60 INPUT A,B,C
70 GOSUB 1000
80 IF B1<S THEN PRINT "S=FNG(A,B,C)";
   ELSE PRINT "ТРЕУГОЛЬНИК НЕ СУЩЕСТВУЕТ";
90 END
```

Основная программа

```
1000 REM ПОДПРОГРАММА 1
1010 B1=FND(A,B)
1020 B1=FND(B1,C)
1030 S=A+B+C-B1
1040 RETURN

2070 RETURN

3100 RETURN
```

II. Составьте блок-схему алгоритма и напишите программу:

- а) в введенном слове удвоить каждую букву;
- б) удалить из слова заданную букву;

- в) заменить в слове одну букву другой;
- г) упорядочить в слове буквы по алфавиту;
- д) разорвать слово на две равные части;

сти и «склеить» полученные части в обратном порядке;

е) упорядочить слова по алфавиту;

ж) перевода названий дней недели с русского языка на английский (немецкий);

з) перевода названий месяцев с английского (немецкого) языка на русский;

и) по координатам трех точек (вершин треугольника) установить вид треугольника и его площадь;

к) по координатам двух векторов вычислить скалярное произведение и угол между векторами;

л) решить треугольник (длины сторон, величины углов, длины радиусов

вписанной и описанной окружностей и площадь), используя данные первого (2, 3) признака равенства треугольников;

м) решить систему двух линейных уравнений с двумя неизвестными (методом Крамера);

н) получить таблицы значений следующих функций:

$$1) y = x^3 + 2x - 3, \text{ для } x \in [a, b]$$

$$2) y = \begin{cases} \sin^2 x + x^3, & \text{если } x < 3 \\ \lg x - e^x, & \text{если } x > 3 \end{cases}$$

$$3) y = \begin{cases} \lg x - x^2 + 4, & \text{если } x < 2 \\ \cos x + \lg x, & \text{если } x = 2 \\ \ln(x-2) - x^2, & \text{если } x > 2 \end{cases}$$

Гродненский филиал Белорусского специализированного территориального управления производственного объединения «Конус»

готов бесплатно предоставить «Е-практикум» для «Агата»
любому учреждению системы народного образования —
достаточно выслать носитель информации.

Обращаться по адресу: 230023. Гродно, а/я 10.
Гродненский филиал БСТУ ПО «Конус».

Кооператив «ЭЛЕКТРОН»

● предлагает владельцам и пользователям ПЭВМ типов ДВК, УКНЦ, IBM XT/AT, «Синклер Спектрум», «Атари», «Коммодор», «Агат», «Специалист», «РК-86» 32К, «РК-86» 64К, «Микроша», БК-0010.01, «Львов», «Партнер», «Вектор», «Правец-8Д» широкий выбор системных, прикладных, игровых, учебных программ;

● предлагает учебные программы для классов УКНЦ, КУВТ-86, «Ямаха»;

● заключает с авторами договоры на тиражирование разработанного ими программного обеспечения с выплатой процентов от реализации (возможен обмен программами);

● продает программно-аппаратные комплексы и игротехи на базе компьютеров «Синклер Спектрум», «РК-86», «Специалист», ДВК; электронные диски (64—256К, с операционной системой, для всех типов бытовых компьютеров).

Адрес для справок и запросов каталогов:

103489, Москва, Зеленоград, корп. 705, кооператив «Электрон».
Телефон 536-12-81.

КуМир и другие

В «Инфо» № 2 было сообщение о бесплатной поставке программного обеспечения для ПЭВМ «Ямаха», разработанного на механико-математическом факультете МГУ. Эта статья более подробно знакомит с составом и возможностями поставляемого программного обеспечения.

В состав бесплатной поставки программного обеспечения для школьных ЭВМ «Ямаха» входят все программные средства, разработанные предприятием «ИнфоМир», механико-математическим факультетом МГУ и лабораториями НСК АН СССР до 1989 г. включительно. К ним относятся:

система программирования КуМир на основе школьного алгоритмического языка;

набор динамически заменяемых исполнителей, управляемых из программ, написанных на школьном алгоритмическом языке;

интегрированные системы текстообработки и организации архива МикроМир-86 и МикроМир-88;

системы редактирования файлов и дисков на физическом уровне;

сетевая операционная система и др.

Интерфейс «ИнфоМир»

Все поставляемые предприятием «ИнфоМир» программные средства построены на одних и тех же принципах

организации взаимодействия «человек — ЭВМ». Принципы изображения информации на экране компьютера и смысл клавиш не меняются от системы к системе и даже (по мере возможности) от одной ЭВМ к другой.

Поскольку разные системы имеют разное назначение, то у них и разные наборы команд. В АльфаМире и Клавире, например, нет вставки/удаления символов, а в КуМире нет выравнивания абзаца по шаблону и т. п. Однако если в двух разных системах есть одинаковые команды, то они вызываются одинаковыми клавишами и выполняются похоже. В этом и состоит главный принцип интерфейса «ИнфоМир».

Мир, окно, курсор. При работе с системами предприятия «ИнфоМир» удобно представлять себе все события происходящими в некотором плоском «Мире», часть которого видна на экране монитора. Иначе говоря, экран служит как бы окном, через которое человек смотрит на «Мир». Сам «Мир» при этом может быть большим и простираться далеко за пределы экрана как по вертикали, так и по горизонтали.

В большинстве случаев «Мир» за экраном представляет собой просто текст — последовательность строчек и символов. Удобно считать себя находящимися внутри «Мира» (внутри текста) — в месте, отмеченном курсором. Курсор можно двигать по «Миру», например, нажимая на клавиши со стрел-

ками. При перемещении за пределы окна курсор «упирается» в край окна и сдвигает его, а изображение на экране соответственно меняется.

Все действия в «Мире» производятся нажатиями на клавиши. С помощью таких нажатий (команд) можно перемещаться по «Миру» (перемещать курсор) и менять «Мир» вблизи курсора.

Обычно все команды выполняются плавно (непрерывно). Например, при перемещениях на экране последовательно изображаются все промежуточные состояния, как если бы вы действительно сами двигались из одной точки «Мира» в другую. Иногда (при больших перемещениях) это может оказаться утомительным. Если нет нужды смотреть на ход выполнения команды, то работу можно ускорить, нажав **Ctrl+O** («закрыть глаза»), — и на экране будет сразу (без промежуточных состояний) изображен результат выполнения команды.

Команды войти/выйти. Конечно, «Мир» не обязан быть таким уж плоским — он может состоять из нескольких плоских кусков, связанных между собою ссылками. Удобно представлять себе каждый кусок в виде отдельной плоскости (листа бумаги), как на приводимом рисунке. Строку, «за которой» находится целый «Мир» (на рисунке таких строк две), можно считать краем, торцом соответствующего «Мира». Поставив курсор на строку-торец, можно нажать **Esc Вниз** («войти») и попасть внутрь соответствующего «Мира». По окончании работы с этим «Миром» (нажав **Esc Вверх** — «вый-

ти») вы окажетесь в том самом месте, где скомандовали «войти».

Поля. В учебных «Мирах» — АльфаМире, КуМире — экран разделен на две части вертикальным «забором»: слева от «забора» записывается текст алгоритма или программы, а справа, на полях, появляются замечания системы по поводу действий человека (сообщения об ошибках и т. п.). Таким образом, зоны действия человека и ЭВМ разделены: человек работает слева от «забора», а система — справа.

Обозначения клавиш. В интерфейсе «ИнфоМир» используются дополнительные (темные) клавиши клавиатуры. Так, клавиша *Ins* используется для вставки символов, строк, для переключения в режим вставки и т. п.; клавиша *Del* используется для удаления; клавиша *Select* — для поиска. В документации предприятия «ИнфоМир» клавиши (или нажатия на клавиши) выделяются полужирным курсивом. При этом используются нанесенные на клавиши общепринятые международные обозначения (*F1, F2, Ins, Del, Ctrl, Bs* и т. д.), а также обозначения *Вправо/Влево/Вверх/Вниз* для клавиш со стрелками (клавиш управления курсором), ◀ — для клавиш с изогнутой стрелкой (*Return*, «исполнение»).

Обозначения нажатий на клавиши. Команды и символы набираются одним из трех способов:

простое нажатие одной клавиши (*A, B, Ins, Del*);

нажатие аккорда из нескольких клавиш (*Ctrl +C, Shift+Caps+A*);

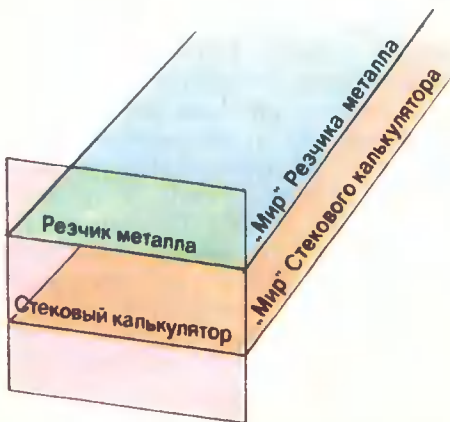
последовательное нажатие клавиш (*Esc Вверх; Home Вниз*).

Обозначение *Ctrl+C* означает, что надо нажать клавишу *Ctrl* и, не отпуская ее, нажать клавишу *C*. Аналогично *Shift+Caps+A* означает, что надо нажать клавиши *Shift* и *Caps* и, не отпуская их, клавишу *A*.

Обозначение *Esc Вверх* означает, что надо нажать клавишу *Esc*, затем отпустить ее и нажать на клавишу со стрелкой вверх. Обозначение *Home Вниз* означает, что надо нажать клавишу *Home*, отпустить ее и нажать на клавишу со стрелкой вниз.

Префиксы и синонимы. Особую роль

26



играют клавиши-префиксы *Home* и *Esc*. Нажатие на них не приводит к выполнению каких-либо действий, но изменяет действие клавиш, нажатых вслед за ними. *Home* обычно усиливает действие следующей клавиши (например, клавиша *Вправо* смещает курсор на один символ вправо, а *Home Вправо* — до конца строки). Клавиша *Esc* обычно используется для специальных действий: смены «Мира», вставки конструкций школьного алгоритмического языка и др.

Для удобства работы некоторые клавиши имеют синонимы. Так, например, *Ctrl+C* и *Stop* дают один и тот же эффект; аккордное нажатие *Ctrl+Вверх* или *Ctrl+Вниз* эквивалентно последовательному нажатию *Home Вверх* или *Home Вниз* соответственно.

Работа с большими/малыми, русскими/латинскими буквами. За смену больших/малых букв отвечает клавиша *Shift*, за смену русских/латинских — клавиши *Caps* и *Рус*, за установку регистра и алфавита — клавиша *Ctrl*:

Ctrl+Рус — одновременная смена алфавита (с русского на латинский и наоборот). Горящая лампочка на клавише *Рус* означает, что установлен русский алфавит;

Ctrl+Caps — одновременная смена регистра (с малых букв на большие и наоборот). Горящая лампочка на клавише *Caps* означает, что ввод будет производиться большими буквами.

Допустим, установлен режим ввода текста малыми русскими буквами, а нужно ввести несколько больших букв. Тогда можно нажать клавишу *Shift* и вводить текст, удерживая ее нажатой (клавиша *Shift* действует на время нажатия). Если же нужно ввести несколько малых латинских букв, то можно удерживать нажатой клавишу *Caps* (или *Рус* — это одно и то же). А для ввода нескольких больших латинских букв можно удерживать обе клавиши *Shift* и *Caps*.

Для перемещения курсора используются следующие команды:

Вправо/Влево — смещение на символ вправо/влево;

Shift+Вправо — смещение в начало следующего слова;

Shift+Влево — смещение в конец предыдущего слова;

Tab/Shift+Tab — вправо/влево по позициям табуляции;

Ctrl+Влево/Вправо или *Home Влево/Вправо* — в начало/конец строки;

Вверх/Вниз — смещение на строку вверх/вниз;

◀ — в начало следующей строки;

Ctrl+Вверх/Вниз или *Home Вверх/Вниз* — двигаться вверх/вниз непрерывно (до нажатия любой клавиши).

Итак, если нужно куда-то попасть — нажимайте на нужные стрелки. Если попасть нужно достаточно далеко — нажимайте *Shift*+стрелки, если совсем далеко — *Ctrl*+стрелки или, что то же самое, *Home* стрелки.

Команды элементарных изменений.

Ctrl+Ins — установить режим вставки;

Ctrl+Del — установить режим замены.

В режиме замены обычные символы, набираемые на клавиатуре, появляются «поверх» старых символов текста, загибая их. В режиме вставки, наоборот, новые символы вставляются, раздвигая старые символы текста.

Режим замены/вставки сказывается и на некоторых других командах. Например, команды удаления предыдущего символа (*Bs*) и начала строки (*Ctrl+U*) в режиме вставки «сжимают» строку, а в режиме замены лишь очищают часть строки пробелами.

Ins/Del — вставка пробела/удаление символа;

Bs — очистка или удаление символа слева от курсора;

Ctrl+U — очистка или удаление начала строки до курсора (в зависимости от режима);

Ctrl+D — удаление конца строки за курсором;

Shift+Ins — вставка пустой строки;

Shift+Del — удаление строки.

Команды «войти» и «выйти».

Esc Вниз — «войти» в «Мир», на краю (торце) которого находится курсор;

Esc Вверх — нормальный (обычный) выход. В системах КуМир и МикроМир при таком выходе результаты работы сохраняются в файлах на дисках;

Esc Stop — аварийный выход (без сохранения результатов работы). Файлы на дисках остаются нетронутыми — в том состоянии, в котором они были до «входа».

Выполнение программ.

F5 — выполнять программу по шагам;

Shift+F5 — выполнить программу целиком;

Ctrl+Stop — прервать (прекратить) выполнение программ.

Общий сервис.

Esc (цифры) — задать число повторений следующей команды;

Home Home — повторить предыдущую команду («еще»);

Ctrl+O — «закрыть»/«открыть» глаза.

Любую команду можно повторить любое заданное число раз. Для этого нужно сначала задать число повторений команды, а уж потом нажимать на соответствующие данной команде клавиши.

Последнюю введенную команду можно повторить еще раз с помощью команды «еще».

Например, последовательность нажатий **Esc 2 4 Вниз** приведет к смещению на 24 строки вниз. Если после этого нажать **Home Home**, то курсор сместится еще на 24 строки вниз.

Дополнительные возможности.

Home Select — поиск по образцу, который должен быть введен. После ввода образца направление поиска можно задать, нажав **Вниз** или **Вверх**.

Select — поиск вниз по образцу, введенному ранее;

Shift+Select — поиск вверх по образцу, введенному ранее.

Shift+F4 — начало/конец запоминания последовательности нажатий клавиш (макрокоманды);

F4 — повторение запомненной последовательности (выполнение макрокоманды).

F1 — добавление символа к запомненному;

Ctrl+F1 — запоминание символа с удалением его из текста;

Shift+F1 — вспоминание запомненных символов и внесение их в текст;

F2 — добавление строки к запомненному;

Ctrl+F2 — запоминание строки с удалением ее из текста;

Shift+F2 — вспоминание запомненных строк и вставка их в текст.

Вспоминать запомненное можно сколько угодно раз. Однако при первом (после вспоминания) запоминании память автоматически очищается.

Работа в классе MSX-2

В классе машин «Ямаха» MSX-2 для работы на компьютерах учеников используется программа поддержки работы в локальной сети SOS (Сетевая Операционная Система). Она позволяет работать на компьютере ученика в стандартной системе MSX DOS, используя в качестве диска электронный диск емкостью порядка 160K байт.

Сеанс работы начинается с рассылки операционной системы и файлов на PMU. Для этого нужно запустить программу SOS. Файлы, которые нужно переслать на PMU, должны быть перечислены в файле SOS.DAT. Если запустить программу SOS, указав ей в качестве параметра имя файла, начинающееся не с цифры, например,

SOS SEND.DAT

то список передачи будет взят из указанного файла (SEND.DAT).

Интерпретатор команд операционной системы MSX DOS, файл COMMAND.COM (или COMMAND.SYS), всегда пересылается на PMU, независимо от указания его в списке пересылки.

Файл SOS.DAT состоит из строк вида [〈адрес пересылки〉] 〈имя файла〉

Если адрес пересылки не указан, то файл пересылается на все PMU. Адрес пересылки — это номера ученических машин, которые должны получить указанный файл. Он должен быть отделен двоеточием от имени файла, например: 1,3,10—12:M80.COM

Здесь файл M80.COM будет передан на PMU с номерами, 1,3,10,11,12.

Чтобы в процессе работы учитель мог принять от ученика или послать ученику какой-нибудь файл, нужно установить связь с компьютером ученика. Для этого нужно запустить программу SOS, указав ей в качестве параметра номер PMU: SOS 12. Тогда РМП

будет связано с РМУ и электронный диск РМУ будет доступен как обычный диск РМП. Пока учитель работает с электронным диском ученика, ученик не может работать. Только по окончании сеанса связи (команда SOS — 12 или SOS —) ученик может продолжить работу.

Некоторые программы, например «МикроМир-88», не разрешают прерывать себя сеансом связи. Поэтому рекомендуется устанавливать связь с компьютером ученика, когда он не выполняет никаких действий, т. е. находится в ОС. Перезагрузка РМП никак не влияет на сеанс работы.

Система программирования КуМир

Система программирования КуМир на основе школьного алгоритмического языка была специально разработана для ввода, редактирования и выполнения программ, задач и упражнений из учебников по школьной информатике.

Создание программ. При работе КуМира на экране виден текст программы, с которой вы работаете. Эта программа записывается практически так же, как это делается в учебнике. Хранящийся в компьютере текст можно осматривать и менять. Если программа длинная, то она целиком на экране не видна — экран представляет собою как бы окно, через которое видно часть программы. Видимый в окне текст разделен на две части вертикальным «забором» из восклицательных знаков: слева от «забора» видна программа, а на правом поле — замечания ЭВМ по поводу программы (сообщения об ошибках и т. п.). Например, при начале работы с КуМиром на экране появляется картинка:

__ КуМир _ ИнфоМир _ мехнат МГУ _ Школа-1 _	
алг ■	Нет имени
дано	
надо	
нач	
.	
кон -----	

алгоритма. Курсор можно перемещать в области экрана левее «забора», а перенести его в правое поле нельзя. Эта область отведена для сообщений ЭВМ, и человек не может в ней что-либо явно изменить. Компьютер же «не вмешивается» в область текста программы, а лишь сообщает «свое мнение» на правом поле.

Рассмотрим процесс создания программы на примере программы решения квадратного уравнения. Если, например, человек в формуле для дискриминанта наберет $4ac$ вместо $4 \cdot a \cdot c$, то при попытке выхода из ошибочной строки система сообщит об ошибке на полях и установит курсор под место ошибки:

__ КуМир _ ИнфоМир _ мехнат МГУ _ Школа-1 _	
алг КВУР(вещ a,b,c, x1,x2, лит ответ)	
дано a,b,c	
надо ответ.x1,x2	
нач вещ d	
. d := b*x2 - 4ac	СИНТАКС.ОШИБКА
.	
кон -----	

В этот момент человек может нажать на клавиши *Ins* и * и получить $4 \cdot a \cdot c$. Если после этого он снова попытается выйти из строки (например, нажмет *Вниз*), то увидит следующее:

__ КуМир _ ИнфоМир _ мехнат МГУ _ Школа-1 _	
алг КВУР(вещ a,b,c, x1,x2, лит ответ)	
дано a,b,c	
надо ответ.x1,x2	
нач вещ d	
. d := b*x2 - 4*ac	ИМЯ НЕ ОПИСАНО
.	
кон -----	

Действительно, система воспринимает «ac» как имя новой, еще не описанной переменной, а не как имена «a» и «c», между которыми пропущен знак операции. Эту ошибку можно исправить, нажав на клавиши *Вправо*, *Ins*, *.

__ КуМир _ ИнфоМир _ мехнат МГУ _ Школа-1 _	
алг КВУР(вещ a,b,c, x1,x2, лит ответ)	
дано a,b,c	
надо ответ.x1,x2	
нач вещ d	
. d := b*x2 - 4*a*c	
.	
кон -----	

Курсор стоит за служебным словом «алг», приглашая человека ввести имя

Отсутствие сообщений на правом поле означает, что программа в данный момент не содержит синтаксических ошибок, хотя может и не быть законченной.

Процесс редактирования программы протекает в соответствии с интерфейсом «ИнфоМира». Есть, однако, и особенности. Например, чтобы вставить конструкцию «если—то— иначе—все», надо последовательно нажать всего на две клавиши — *Esc* и *E* (*E* — первая буква слова «если»). После этого там, где был курсор, появится конструкция «если—то—иначе—все» (раздвигая текст программы):

<pre> __ КуМир _ ИнфоМир _ мехнат МГУ _ Школа-1 _ алг КВУР(вещ a,b,c, x1,x2, лит ответ) дано a,b,c надо ответ,x1,x2 нач вещ d . d := b*x2 - 4*x*c . если Δ . . то иначе . . . все . кон ----- </pre>	нет условия
--	-------------

Курсор будет установлен сразу за словом «если», в место, куда следует ввести условие. Теперь остается лишь заполнить пустые места, написав условие после «если» и нужные команды после «то» и «иначе» (вставляя, если это необходимо, пустые строки):

<pre> __ КуМир _ ИнфоМир _ мехнат МГУ _ Школа-1 _ алг КВУР(вещ a,b,c, x1,x2, лит ответ) дано a,b,c надо ответ,x1,x2 нач вещ d . d := b*x2 - 4*x*c . если d<0 . . то ответ := "нет решений" . . иначе ответ := "есть решения" . . x1 := (-b - sqrt(d)) / (2*a) . . x2 := (-b + sqrt(d)) / (2*a) . все Δ . кон ----- </pre>	
--	--

Вставка других конструкций школьного алгоритмического языка производится аналогичным образом.

Разумеется, вставка конструкции происходит, только если мы вставляем ее в разрешенное место. Например, если курсор стоит между «дано» и «надо», то при нажатии *Esc E* раздастся звуковой сигнал (зуммер), но вставки команды ветвления не произойдет — это запрещено правилами алгоритмического языка.

После вставки каждой конструкции курсор попадает внутрь нее, в ближайшее незаполненное место.

Для удаления конструкции следует установить курсор в ее первую строку (например, в строку «если») и нажать *Shift+Del*. После этого конструкция целиком исчезнет с экрана вместе со всеми командами, входящими внутрь нее. Если нажать *Shift+Del*, когда курсор стоит в строке со словом «иначе», то исчезнет эта строка и останется команда ветвления в сокращенной форме.

Такой способ позволяет избежать многих ошибок. Тем не менее возможность для ошибок в записи программ остается: можно неправильно записать условие или выражение, использовать переменную как переменную целого типа, когда она описана как литерная, и т. п. Все такие ошибки будут отмечены на полях программы, и отметки сохранятся, пока не будет устранен источник ошибки.

Выполнение программ. В «КуМире» выполнить созданную программу можно либо целиком, либо по шагам — команда за командой. Чтобы выполнить программу целиком, надо нажать *Shift+F5*. Поскольку программа КВУР в данном случае не вызывается ни из какой другой программы и значения ее аргументов не заданы, то в процессе выполнения при первой же попытке использовать эти значения появится сообщение о неопределенности аргументов:

Значения аргументов можно задать прямо в строке «ДАНО» и опять нажать *Shift+F5*. В процессе выполнения программы на полях показывается ход выполнения, т. е. аргументы программ, а также результаты проверок ус-

пример, изменив значения аргументов, можно выполнить программу еще раз:

<pre> __ КуМир _ ИнфоМир _ мехнат НГУ _ Школа-1 _ алг КВУР(вещ a,b,c, x1,x2, лит ответ) дано a,b,c надо ответ,x1,x2 нач вещ d . d := bXX2 - 4XaXc . если d<0 . . то ответ := "нет решений" . . иначе ответ := "есть решения" . . x1 := (-b - sqrt(d)) / (2Xa) . . x2 := (-b + sqrt(d)) / (2Xa) . все . кон </pre>	<p>X Выполнение X</p> <p>не определено</p>
--	--

<pre> __ КуМир _ ИнфоМир _ мехнат НГУ _ Школа-1 _ алг КВУР(вещ a,b,c, x1,x2, лит ответ) дано a,b,c = 1,2,1 надо ответ,x1,x2 нач вещ d . d := bXX2 - 4XaXc . если d<0 . . то ответ := "нет решений" . . иначе ответ := "есть решения" . . x1 := (-b - sqrt(d)) / (2Xa) . . x2 := (-b + sqrt(d)) / (2Xa) . все . кон </pre>	<p>X Выполнение X</p> <p>0</p> <p>ложно</p> <p>есть решения</p> <p>-1</p> <p>-1</p> <p>XXXXX Конеч XXXXX</p>
--	--

<pre> __ КуМир _ ИнфоМир _ мехнат НГУ _ Школа-1 _ алг КВУР(вещ a,b,c, x1,x2, лит ответ) дано a,b,c = 1,2,1 надо ответ,x1,x2 нач вещ d . d := bXX2 - 4XaXc . если d<0 . . то ответ := "нет решений" . . иначе ответ := "есть решения" . . x1 := (-b - sqrt(d)) / (2Xa) . . x2 := (-b + sqrt(d)) / (2Xa) . все . кон </pre>	
--	--

Из этого рисунка видно, что при выполнении программы для аргументов $a=1$, $b=2$, $c=3$ серия после «иначе» вообще не выполнялась (на полях ничего не написано).

31

Выполнение программы в любой момент можно прервать, нажав **Ctrl+Stop**.

Нажимая только **F5**, можно выполнять программу и по шагам — команда за командой. После любого шага выполнение программы можно прекратить, нажав **Ctrl+Stop**.

ловий и выполнения команд присваивания. По окончании выполнения в строке «кон» появляется слово «Конец».

<pre> __ КуМир _ ИнфоМир _ мехнат НГУ _ Школа-1 _ алг КВУР(вещ a,b,c, x1,x2, лит ответ) дано a,b,c = 1,2,3 надо ответ,x1,x2 нач вещ d . d := bXX2 - 4XaXc . если d<0 . . то ответ := "нет решений" . . иначе ответ := "есть решения" . . x1 := (-b - sqrt(d)) / (2Xa) . . x2 := (-b + sqrt(d)) / (2Xa) . все . кон </pre>	<p>X Выполнение X</p> <p>-0</p> <p>истинно</p> <p>нет решений</p> <p>XXXXX Конеч XXXXX</p>
--	--

Исполнители КуМира

В КуМире имеется возможность подключения внешних исполнителей и составление алгоритмов управления ими. Исполнители Робот+Измеритель и Чертежник позволяют программе, написанной на алгоритмическом языке, рисовать картинки, управлять Роботом и т. д.

При работе с данными исполнителями имеются следующие дополнительные предписания, оформленные в виде встроенных вспомогательных алгоритмов: текст — перейти в режим только текстового отображения (все 24 строки экрана отводятся под текст). Этот переход может быть также осуществлен в режиме редактирования программы нажатием **Ctrl+T**;

В этот момент по нажатии любой клавиши слово «Выполнение» погаснет, поля очистятся от результатов и программу снова можно будет менять. На-

робот — при выполнении данного предписания КуМир переходит в режим отображения лабиринта и поля Чер-

тежника (в левой верхней части экрана изображается поле лабиринта, правая верхняя часть освобождается для исполнителя Чертежник, в нижней части — текст).

Следующая группа предписаний управляет действиями Робота:

шаг_на_восток / шаг_на_север / шаг_на_запад / шаг_на_юг — сделать шаг в указанном направлении;

закрасить — закрашивается клетка, в которой стоит Робот, он при этом остается на месте.

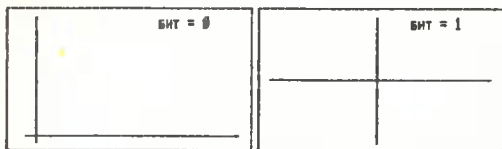
Группа предписаний типа «лог алг», вырабатывающих значение «истинно/ложно» в зависимости от обстановки в лабиринте: лог_на_востоке_свободно / лог_на_севере_свободно / лог_на_западе_свободно / лог_на_юге_свободно / лог_клетка_закрашена.

Группа предписаний Измерителя, вырабатывающих значение вещественного типа в зависимости от обстановки в лабиринте:

вещ_уровень_радиации — выдает значение уровня радиации в текущей клетке;

вещ_температура — значение температуры в текущей клетке.

Для Чертежника первое предписание — чертежник (цел бит). Если при его выполнении был включен режим текстового отображения, то в верхней части экрана освобождается место для поля лабиринта (в левой половине) и для Чертежника (в правой половине) и рисуются координатные оси.



Если верхняя часть экрана уже была освобождена от текста, то результатом выполнения предписания будет только рисование координатных осей.

В дальнейшем могут даваться предписания поднять перо / опустить перо / сместиться_в_точку (вещ x,y) / сместиться_на_вектор (вещ x,y).

Переход в режим редактирования лабиринта осуществляется путем нажатия *Esc* и *Л* (Лабиринт). В этом режиме

можно изменять структуру лабиринта (в том числе значения температуры и радиации), закрашенные клетки и начальное положение Робота:

Вверх / *Вниз* / *Вправо* / *Влево* — перемещение курсора;

Ctrl+Вверх / *Ctrl+Вниз* / *Ctrl+Вправо* / *Ctrl+Влево* — поставить (если стенки нет) или убрать (если она есть) стенку с соответствующей стороны;

Номе — закрасить клетку или убрать закрашивание;

Ins — установка начального положения Робота;

◀ — выйти из режима редактирования.

При редактировании лабиринта в верхней правой части экрана изображается значение радиоактивности и температуры в клетке, в которой стоит курсор.

Для изменения температуры и уровня радиации надо просто ввести новые значения в том же формате, в котором они изображаются на экране.

Работа в классе. Помимо очевидного выигрыша, который дает использование исполнителей при изложении алгоритмического языка, наличие системы программирования с такими исполнителями позволяет не только писать программы, но и гибко организовать работу в классе, например:

учитель заранее создает программы и лабиринты для исполнителей и в начале сеанса работы рассылает по локальной сети ученикам систему КуМир с готовым лабиринтом;

в процессе работы учитель может изменить объект выполнения программы ученика, разослав каждому ученику индивидуальный лабиринт;

ученик может самостоятельно провезти особые ситуации, отредактировав свой лабиринт, т. е. изменив объект выполнения программы.

Надо отметить, что КуМир является обучающей средой, а не автоматизированной обучающей системой: КуМир не предопределяет список задач и не оценивает действий ученика. Это остается прерогативой учителя.

Коллектив исполнителей

Идея создания исполнителя наглядного, динамичного, послушного не нова. И в литературе, и в программах уже давно встречаются «Муравьи», «Роботы», «Стрелочники», «Тараканы», «Корректировщики» и т. п. Однако круг задач, решаемых этими исполнителями, изначально строго ограничен. В системе «Звездочка и др.» реализована идея объединить различные исполнители в одной программе, поместить их в одном компьютерном пространстве, снабдить одной системой команд, однако интерпретировать ряд команд для различных исполнителей по-разному. Это привело к тому, что круг задач, решаемых исполнителями, расширился. Появилась возможность придумывать новые задачи, чему способствовало введение объектов пространства, наделенных различающимися свойствами.

Язык программирования системы удобен при запоминании, редактировании и отладке программы, близок к алгоритмическому языку и может быть использован при обучении программированию как младших школьников, так и их более старших товарищей. Имена команд языка отличаются друг от друга первыми (в крайнем случае вторыми) буквами, что дает возможность резко упростить набор текста и исключить синтаксические ошибки.

Следует сказать, что ограничение длины командной строки и условие несовпадения первых — вторых символов во многом повлияли на синтаксис языка, а ограниченный объем оперативной памяти БК-0010 не дал возможность реализовать все замыслы как в системе команд, так и в количестве исполнителей и значительно ухудшил красочность реализации системы.

Звездочка и другие

В системе «Звездочка и др.» главными действующими лицами являются роботы-исполнители Звездочка, Птичка

и Тараканчик. Проживают они на экране монитора в поле П. Поле это прямоугольное, размером 16×16 и расположено в правой верхней части экрана. Роботы имеют образ символа, который может перемещаться в поле П в любом из четырех направлений, исследуя при желании его участки, а также преобразуя или не преобразуя их. «Повадки» роботов в одной и той же ситуации могут различаться.

Звездочка может беспрепятственно перемещаться по всему полю П, оставляя за собой любой заданный символьный след. Для нее не составляет труда нарисовать квадрат, спираль, шахматное поле, сложный лабиринт или любую другую картинку.

Птичка как бы парит над полем П, умея при необходимости записать любой символ в точку поля, находящуюся под ней. Кроме того, Птичка может читать и запоминать символы, разбросанные по полю, а затем записывать их в другие участки поля. С Птичкой стоит иметь дело, если необходимо написать слово, подготовить арифметический пример, нарисовать картинку, состоящую из множества различных символов, или просто перелететь в другую точку поля, не затрагивая участки пути. С помощью Птички можно скопировать любой участок поля, нарисовать картинку, симметричную данной, зашифровать и расшифровать текст и т. д. и т. п.

Третий в компании — многим знакомый трудяга Тараканчик, который толкает перед собой встреченные на пути символы-кубики, складывая из них слова, решая математические задачи, соединяя разрозненные куски картинки. Вдобавок он умеет проходить сам и проводить кубики по запутанным ходам лабиринтов, чьи стены состоят из символов, которые Тараканчик при всем желании не может сдвинуть с места.

Надо сказать, что все три робота прекрасно уживаются в одной программе, взаимно дополняя друг друга.

Система команд языка «И»

Ключ	Имя	Параметр	Комментарий
			s — произвольный символ, f — число ($0 \leq f < 251$ или $f=x$)
			<i>Команды состояния</i>
Р	РОБОТ	s	* — Звездочка, & — Птичка, ☿ — Тараканчик. 0, 1, 2, 3 — черный, красный, синий, зеленый (черный, белый, серый, серый)
Ц	.ЦВЕТ	f	
			<i>Команды перемещения</i>
	.НА СЕВ.	f	Пошаговое
	.НА ЮГ	f	движение
	.НА ЗАП.	f	в заданном
	.НА ВОСТ.	f	направлении
ВП	.ВПЕРЕД	f	Движение в направлении ориентации
ПР	.ПРАВЫЙ		Поворот на 90° направо
Л	.ЛЕВЫЙ		Поворот на 90° налево
Д	.ДОМОЙ		Перемещение в точку (0, 0) поля «П»
З	ЗАПОМНИ		Запоминание координат текущей точки (X, Y)
ВЕ	ВЕРНИСЬ		Возвращение в точку с координатами (X, Y)
			<i>Команды преобразования</i>
О	ОЧИСТИ!		Установка исходного состояния
Ф	ФОН	s	Заполнение поля «П» символом
Ч	.ЧИТАЙ		Шаг вперед с одновременным запоминанием символа текущей точки поля «П» в памяти Птички
ПИ	.ПИШИ	s	Записать s в текущую точку и сделать шаг вперед; если $s=\&$, то в текущую точку записывается символ из памяти Птички
?	.???	s	Записать s скрыто (символ s маскируется знаком?)
С	.след	s	Звездочка и Тараканчик начинают оставлять след из символов
			<i>Арифметические команды</i>
=	.X=	f	Присваивание; X — числовая
+	.X=X+	f	переменная,
-	.X=X-	f	$0 \leq X < 32767$
			<i>Составные команды</i>
Е	ЕСЛИ	s	Проверка совпадения s с символом точки поля «П», расположенной вперед роботом
Т	ТО		Выполнение при совпадении (необязательная)
И	ИНАЧЕ		Выполнение при несовпадении (необязательная)
К	КОНЕЦ		Конец команды ветвления
ПО	ПОВТОРИ	f	Начало цикла
К	КОНЕЦ		Конец цикла
ВЫ	ВЫПОЛНИ	s	Вызов подпрограммы с именем s
А	АЛГ.	s	Подпрограмма s (вспомогательный алгоритм)
К	КОНЕЦ		Конец подпрограммы
			<i>Команды управления</i>
Ж	ЖДИ!		Прерывание выполнения программы в заданной точке. Выполнение возобновляется при нажатии клавиши «пробел»
Ш	ШАГ	f	Режим выполнения программы: 0 — без трассировки, 1 — пошаговый; $f > 1$ — скорость трассировки (исходно $f=2$)
Н	НОМЕР	f	Динамическая загрузка с диске и выполнение программного файла с номером f (имя «Т»+ f)
			<i>Дополнительные команды</i>
Б	БИП—БИП	f	Звуковой сигнал длительностью f Команда «нет операции» (используется для оформления программы)

Состав системы и язык программирования

Система «Звездочка и др.» состоит из интерпретатора языка И, экранного

редактора и управляющей программы, которые резидентно загружаются в ОЗУ БК-0010. Предоставляется возможность выполнять команды в непосредственном режиме, создавать программы

длиной до 230 строк, выполнять программу, редактировать текст и отлаживать программу, не выходя из системы. Кроме того, система позволяет в сети КУВТ-86 хранить тексты программ на дискете, считывать и выполнять их, что дает возможность комплектовать систему примерами и задачами.

Система команд языка И содержит 32 команды, которые делятся на следующие основные группы: состояния исполнителя, перемещения исполнителя, преобразования поля П, арифметические, составные (ветвление, цикл, подпрограмма) и управляющие команды.

Робот имеет следующие характеристики: тип (символ), цвет, ориентацию и координаты в поле П. Первые две характеристики определяются командами состояния. Команды перемещения изменяют его ориентацию, осуществляют движение в заданном направлении или мгновенное перемещение в заданную точку поля. Команды преобразования производят запись (или чтение) символа в данную точку поля, очищают поле или заполняют его заданным символом. Кроме того, имеется возможность записать символ, значение которого скрыто под знаком «?» и определяется лишь при дополнительном исследовании. Арифметические команды выполняют присваивание числового значения единственной переменной X и выполняют операции сложения (вычитания) X с заданным числом.

Команда ветвления сравнивает значение текущей точки поля с заданным символом; при равенстве выполняется серия 1, в противном случае — серия 2. Команда цикла позволяет выполнить серию команд заданное число раз. Подпрограммы формируются ниже тела основной программы. Обращение к ним возможно из любой точки программы с последующим возвратом в точку вызова.

Разрешены многоуровневые структуры и рекурсия. В сериях всякой составной команды могут находиться составные команды. Глубина вложенности не более 10.

Команды управления позволяют установить режим выполнения программы, выполнить трассировку командных строк

ЕСЛИ "(символ)"	ПОВТОРИ (число)
ТО	<серия>
<серия 1>	КОНЕЦ
ИНАЧЕ	
<серия 2>	в)
КОНЕЦ	
а)	
<серия>	
ВЫПОЛНИ "(символ)"	→ АЛГ. "(символ)"
<серия>	<серия>
	← КОНЕЦ

в)

с заданным временным интервалом или в пошаговом режиме, остановить выполнение программы в заданной точке, прочитать (записать) текст программы с дискеты в сети КУВТ-86 и выполнить ее.

Редактирование и управление

Экран редактора находится в левой части экрана монитора и постоянно отображает фрагмент программы длиной в 22 строки. Командная строка, расположенная в центре экрана редактора и выделенная инверсным отображением, называется текущей строкой. В редакторе есть пять управляющих клавиш-команд, выполняющих определенные функции относительно текущей строки.

При нажатии клавиши «пробел» управление передается интерпретатору, который выполняет программу начиная с текущей строки и возвращает управление редактору либо выполнив последнюю команду программы, либо при повторном нажатии клавиши «пробел».

Ввод текста программы осуществляется в режиме вставки клавишами, соответствующими первым (вторым) символам команд языка И. Клавишей ← осуществляется уничтожение текущей строки. Клавишами →, ← осуществляется просмотр текста программы сверху вниз и снизу вверх соответственно. Клавиша ВС текущей строкой устанавливает первую командную строку программы. Клавиши СУ+Г уничтожают все строки программы.

Объекты поля П

Объекты-символы поля П различаются по своему взаимодействию с роботами. Для Тараканчика существуют объекты трех типов: «кубик», «стена», «след». «Кубики» — это символы,

которые Тараканчик в любом допустимом размерах поля количестве может толкать перед собой до края поля П или «стены». «Стена» является непреодолимым препятствием для Тараканчика и кубиков, которые он толкает. «Следы» прозрачны для Тараканчика и куби-

ков и могут быть использованы для лучшей ориентации в поле П. «Следы» — это пробел, точка и подчеркивание; «стена» — символы # и ■; остальные символы — «кубики».

Для Звездочки и Птички все объекты поля П равноценны.

А. МАТВЕЕВ

«Микроша»: 48К байт

Модуль дополнительного оперативного запоминающего устройства (в дальнейшем — модуль ОЗУ) предназначен для расширения адресного пространства системного блока ПЭВМ «Микроша» с 32К байт до 48К байт. Рабочие ячейки модуля ОЗУ расположены в интервале адресов 8000—BFFF. Модуль подключается к внутренним шинам системного блока через разъем «Внутренний интерфейс».

Устройство

В составе модуля можно выделить следующие функциональные части:

схему временного мультиплексирования адреса;

схему формирования сигналов выборки («RAS» и «CAS»);

блок микросхем оперативной памяти.

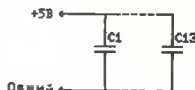
Особенностью микросхем КР565РУ6Д, применяемых в модуле ОЗУ, является временное мультиплексирование адресов и необходимость периодической регенерации содержимого ячеек памяти. Первая особенность обусловлена тем, что у ИБС КР565РУ6Д имеется только 8 адресных входов, в то время как разрядность шины адреса системы равна 16. Вследствие этого возникает необходимость в разделении по времени моментов поступления на адресные входы микросхем памяти младших и старших разрядов адреса. Код адреса заносится в адресные регистры микросхем через входы А0—А6 последовательно: сначала поступают коды семи младших разрядов шины адреса, сопровождаемые низким уровнем сигнала «RAS», а затем семи старших, сопровождае-

мые низким уровнем сигнала «CAS».

ИМС КР565РУ6Д имеет организацию 128×128. Это означает, что матрица запоминающих элементов имеет 128 строк и 128 столбцов. Младшие семь разрядов шины адреса служат для выбора строки запоминающих элементов, а старшие семь — для выбора столбца. Временное разделение адресов производится мультиплексорами D1 и D2, на входы которых с шины адреса поступают разряды А0—А13 кода адреса. В зависимости от уровня сигналов на входах V этих микросхем на входы А0—А6 микросхем памяти поступают сигналы либо с линий А0—А6, либо с линий А7—А13 шины адреса. Сигнал,

36

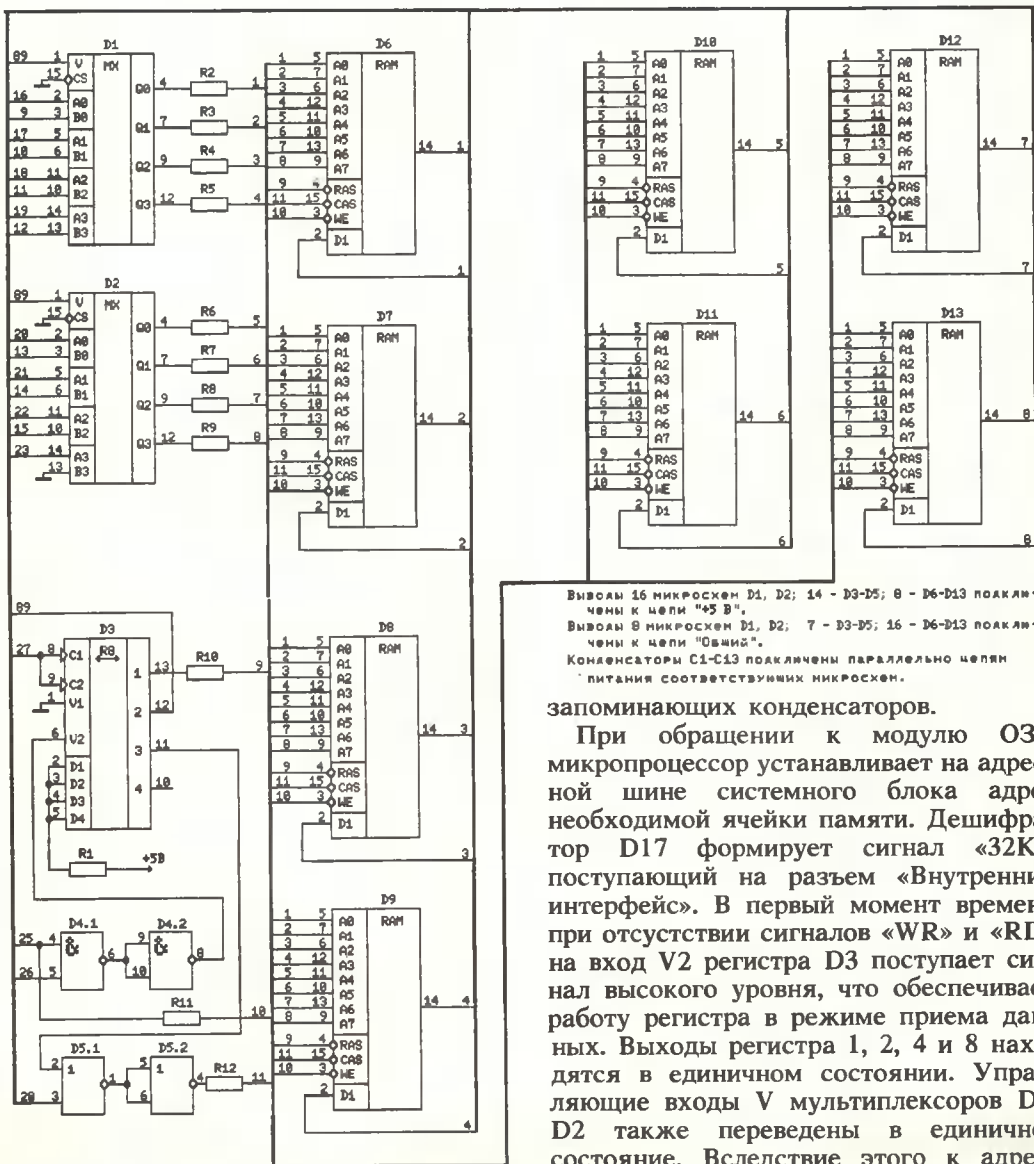
Наим.	Конт.	
D0	A15	1
D1	A16	2
D2	A17	3
D3	A18	4
D4	A22	5
D5	A21	6
D6	A20	7
D7	A19	8
A0	B8	9
A1	B9	10
A2	B10	11
A3	B12	12
A4	B13	13
A5	B14	14
A6	B15	15
A7	B16	16
A8	B17	17
A9	B18	18
A10	B24	19
A11	B23	20
A12	B20	21
A13	B21	22
A14	B22	23
A15	B19	24
CAS2	A9	
R̄B	B6	25
OC	B27	27
R̄B	A7	26
+5В	B30	
Общий	A2	
32к	B4	28



поступающий с выхода Q3 мультиплексора D2, для адресации памяти не используется.

Необходимость специальных мер по регенерации содержимого ячеек памяти обусловлена тем, что основным запоминающим элементом в микросхемах памяти динамического типа, в том числе и КР565РУ6Д, является конденсатор. Для предотвращения потери информации необходимо не реже чем через 2 мс производить обращение к каждой

ячейке. Обычно для этого используется режим прямого доступа к памяти. Для регенерации всего объема памяти оказывается достаточным производить обращение последовательно к каждой из 128 строк матрицы запоминающих элементов, сопровождая каждое обращение низким уровнем сигнала «RAS». При этом к внутренним усилителям считывания микросхем подключается соответствующая строка матрицы и автоматически производится подзарядка



запоминающих конденсаторов.

При обращении к модулю ОЗУ микропроцессор устанавливает на адресной шине системного блока адрес необходимой ячейки памяти. Дешифратор D17 формирует сигнал «32K», поступающий на разъем «Внутренний интерфейс». В первый момент времени при отсутствии сигналов «WR» и «RD» на вход V2 регистра D3 поступает сигнал высокого уровня, что обеспечивает работу регистра в режиме приема данных. Выходы регистра 1, 2, 4 и 8 находятся в единичном состоянии. Управляющие входы V мультиплексоров D1, D2 также переведены в единичное состояние. Вследствие этого к адрес-

ным входам микросхем памяти подключены младшие разряды шины адреса. При появлении одного из сигналов — «WR» или «RD» — происходит переключение логических элементов D4.1 и D4.2. В результате вход V2 регистра D3 переводится в нулевое состояние, а сам регистр — в режим параллельного сдвига информации. В этом режиме с приходом первого тактирующего импульса «OSC» на выходе 1 регистра появляется сигнал низкого уровня. Этот сигнал («RAS») стробирует запись кода, находящегося на входе микросхем памяти, во внутренние регистры. При поступлении второго тактирующего импульса нулевой уровень появляется на выходе 2 регистра, что приводит к изменению состояния входов V мультимплексоров и к подключению к адресным входам микросхем памяти старших разрядов шины адреса. При прохождении третьего импульса «OSC» нулевой уровень на выходе 4 регистра совместно с нулевым уровнем сигнала «32K» формирует сигнал «CAS». Считывание бита из ячейки БИС ОЗУ производится в момент действия сигнала «CAS», если предварительно был установлен сигнал «RAS». На время действия сигнала «CAS» информационный выход микросхем переходит из высокоимпедансного состояния в режим выдачи бита. Если одновременно с сигналом «CAS», при предварительном установленном «RAS», действует активный сигнал «WE», то бит данных со входа D1 будет записан в соответствующую ячейку запоминающей матрицы микросхемы. При этом информационный выход находится в высокоимпедансном состоянии.

Цикл регенерации дополнительной оперативной памяти осуществляется в процессе регенерации основной памяти системного блока.

Порядок изготовления

Сборка модуля ОЗУ является очень ответственной операцией; наибольшее число отказов модуля происходит в результате ошибок монтажа. Микросхемы КР565РУ6Д выполнены по ЛМОП технологии, поэтому при размещении и монтаже необходимо пользоваться заземленным браслетом и

электропаяльником с заземленным жалом. Для предотвращения термических повреждений ИМС мощность электропаяльника не должна превышать 30 Вт. В процессе распайки выводов микросхем обязательно применение теплоотвода.

Размещение элементов на монтажной печатной плате может быть произвольным, единственным условием является минимальная длина соединительных проводников. Минимальная длина необходима для повышения помехозащищенности модуля и предотвращения наводок на сигнальные цепи. Сечение проводника сигнальной шины не должно быть менее 0,1 мм, а шин питания и корпуса — 0,2 мм.

Внимание! При монтаже особое внимание следует уделить разводке шин питания и корпуса. Ошибочная разводка этих шин чаще всего приводит к выходу из строя микросхем как собираемого модуля, так и системного блока.

Если все комплектующие элементы исправны и монтаж произведен без ошибок, модуль дополнительного ОЗУ начинает работать без каких-либо регулировок. Подключение модуля к разъему «Внутренний интерфейс» системного блока производится до включения питания системного блока.

Если после включения питания и нажатия клавиши «Сброс» системный блок не выходит в программу «СИСТЕМНЫЙ МОНИТОР» (нет сообщения «—МИКРОША—», символа «→» и курсора), то это свидетельствует о неисправности модуля.

При обнаружении неисправности нужно сразу же отключить питание системного блока и еще раз тщательно проверить правильность разводки сигналов «Корпус» и «+5 В». Необходимо также убедиться в отсутствии замыканий между соседними выводами микросхем, контактными площадками и соединительными проводниками. После устранения обнаруженных ошибок можно произвести повторное включение модуля. Если и при повторном включении модуль окажется неработоспособным, то скорее всего неисправна одна из микросхем. Для ее обнаружения потребуется осциллограф. Перед началом регулиров-

ки необходимо соединить клемму «Корпус» осциллографа с шиной корпуса модуля ОЗУ.

Отыскание неисправности начинается с проверки наличия всех основных сигналов на контактах разъема. С помощью осциллографа проверяется наличие сигналов «WR», «RD», «OSC», сигналов адреса, данных и «32К». Последний сигнал должен быть единичного уровня. При подключении осциллографа к контрольным точкам схемы не допускайте замыкания щупом электрически не связанных контактов. Это может привести к отказам ИМС. Если один из сигналов отсутствует, нужно, выключив питание, отпаять проводник, соединяющий контакт разъема с соответствующим выводом микросхемы. Затем необходимо включить питание, нажать клавишу «Сброс» и вновь проверить наличие сигнала. При появлении сигнала на разъеме с достаточной степенью вероятности можно говорить о неисправности данной ИМС.

Например, после включения питания и сброса отсутствует сигнал «WR». Этот сигнал, как показано на принципиальной схеме модуля, поступает на вывод 4 элемента D4.1 и выводы 3 микросхем D6—D9. Последовательно отпаивая каждый из проводников, соединяющих указанные выводы с линией сигнала «WR», можно определить, какая из микросхем является источником неисправности. Такая методика является общей для отыскания неисправностей и в цепях других сигналов.

Если все сигналы присутствуют на разъеме, следует проверить работоспособность всех микросхем модуля.

При единичном уровне сигнала «32К» на выводе 1 элемента D5.1 должен присутствовать сигнал нулевого уровня, а на выводе 4 — единичного. Такой же уровень должен наблюдаться на выводах 15 микросхем D6—D13. Измерение сигналов производится непосредственно на ножках микросхем для обнаружения скрытых дефектов пайки. При прохождении сигнала «WR» или «RD» происходит переключение элементов D4.1, D4.2 и, следовательно, уровня сигнала на выводе 6 регистра D3. После этого необходимо убедиться в на-

личии тактовых импульсов на выводах 8,9 регистра. Если на выводы 2—5 регистра поступает сигнал высокого уровня, а на вывод 1 — низкого, то на каждом из выводов 11—13 регистра должны присутствовать импульсы низкого уровня. Кроме этого, необходимо проконтролировать поступление сигналов с выводов 13—11 регистра на выводы 4 микросхем D6—D13, выводы 1 мультиплексоров D1, D2 и вывод 2 элемента D5.1 соответственно. Затем следует перейти к проверке работоспособности мультиплексоров. Для этого нужно проверить, подключены ли выводы 15 микросхем D1, D2 к шине корпуса. Необходимо также убедиться в наличии сигналов кода адреса на выводах 2, 3, 6, 10, 11, 14, 15 микросхем D1 и выводах 2, 3, 5, 6, 10, 11 микросхем D2. Если указанные сигналы приходят на мультиплексоры, то на выводах 4, 7, 9, 12 D1 и 4, 7, 9 D2 должны периодически сменять друг друга 7 младших и 7 старших разрядов шины адреса. Необходимо проконтролировать поступление сигналов адреса на выводы 5—7 и 10—13 микросхем D6—D13. После этого целесообразно проверить прохождение сигнала «CAS». Для этого нужно набрать на клавиатуре системного блока D8000, BFFF, подключить щуп осциллографа к выводу 4 элемента D5.2 и нажать «BK». На экране осциллографа должен наблюдаться сигнал «CAS». На заключительном этапе проверки нужно проверить поступление сигналов с линий магистрали данных на выводы 2 и 14 микросхем D6—D13.

Обычно проверка в указанном объеме дает положительные результаты, однако же и она не дает стопроцентной гарантии работоспособности модуля. Время и качество настройки во многом зависят от опыта и квалификации радиолюбителя.

После того как удалось добиться начальной установки системного блока с подключенным модулем оперативной памяти, нужно проверить правильность записи и считывания информации из модуля. Для этого используются директивы программы «СИСТЕМНЫЙ МОНИТОР».

Наберите на клавиатуре директиву «M8000» и нажмите три раза клавишу «BK». На экран будет выведено содержимое первых трех ячеек памяти дополнительного ОЗУ. Если эти ячейки содержат байт 82, то это говорит о неработоспособности схемы выборки модуля. В этом случае необходимо провести проверку микросхем по вышеописанной методике. Если содержимое ячеек отлично от 82, попытайтесь записать во все ячейки интервала 8000—BFFF какую-либо константу. Используйте при этом директиву F «СИСТЕМОНО МОНИТОРА»:

F8000, BFFF,55.

Проверьте правильность записи с помощью директивы D:

D8000, BFFF

При полностью рабочем модуле во всех ячейках должен содержаться байт 55. Затем заполните интервал какой-либо другой константой, например AA. Произведите сравнение интервалов памяти:

C8000, 9FFF, A000

Если после выполнения директивы не было выведено сообщение об ошибке, то модуль ОЗУ можно считать полностью работоспособным.

М. ГИНЗБУРГ

Графика БК на уроке

Интерес учеников к информатике как предмету довольно быстро угасает в безмашинном варианте и не так быстро (и не у всех) в случае использования ПЭВМ. Его можно поддерживать, используя компьютерные игры, но такой путь затрудняет проявление интереса исследовательского, хотя бы на уровне: «А что будет, если...?» Практика использования предлагаемых здесь простых программ показывает, что на упомянутом выше элементарном уровне исследовательские мотивы пробуждаются практически у всех учеников. У более сильных интерес носит и более глубокий характер: предвидение результатов эксперимента, поиски и находки соответствующих физических моделей, исследование их поведения и т. п. Благодаря этим программам на экране дисплея БК-0010 появляются многоугольники, звездочки, фигуры Лиссажу, спирали и другие фигуры. Программы составлены на Фокале. Основная часть всех программ — программа построения правильного многоугольника.

Для проведения прямой линии на экране с помощью функции вектора (FV) нужно знать координаты начальной и конечной точек, т. е. координаты вершин N-угольника. Пусть правильный N-угольник вписан в круг радиуса R; координаты центра круга x_0, y_0 . Начало

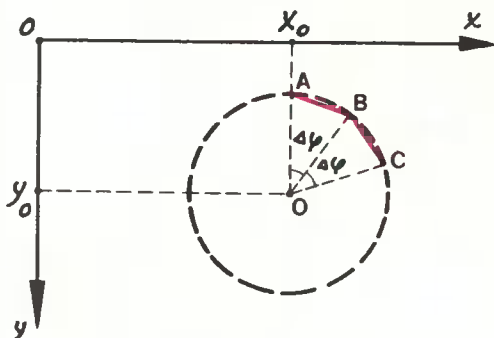
координат расположено в верхнем левом углу экрана, ось y направлена вниз (рис. 1). Будем рисовать многоугольник от точки A ($x_0, y_0 - R$). Угол между OA и OB равен $2\pi/N = \Delta\varphi$. Тогда

$$x_B = x_0 + R \sin \Delta\varphi, \quad y_B = y_0 - R \cos \Delta\varphi,$$

$$x_C = x_0 + R \sin 2\Delta\varphi, \quad y_C = y_0 - R \cos 2\Delta\varphi,$$

...

$$x_A = x_0 + R \sin N\Delta\varphi = x_0, \quad y_A = y_0 + R \cos N\Delta\varphi = y_0 - R.$$



1

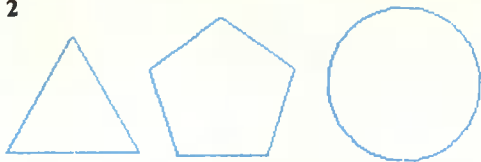
Отсюда сразу получается программа:

```

1.1 A "Количество сторон",N:
                                     S B=2*3.14159/N
1.2 X FT(0.250,60)
1.3 F K=1,N: X FV(1.250+100*FSIN(K*B),
                                     120-60*FCOS(K*B))
1.4 G

```

На рис. 2 показана работа этой программы для N=3, 5, 30. Координаты центра круга выбраны приблизительно в



центре экрана при работе на узком курсе (250, 120). Различные радиусы (100 и 60) при работе с x и y выбраны из-за разных масштабов этих направлений на экране. Для получения каждого многоугольника отдельно нужно перед выводом очередного N -угольника очищать экран (X FCHR(12)).

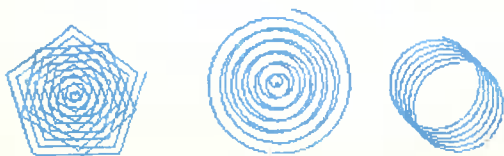
Уже на рис. 2 заметно, что при больших N многоугольник превращается в окружность. Если отношение сомножителей при SIN и COS будет отличаться от упомянутого выше (100/60), то вместо круга получится эллипс, а правильные многоугольники превратятся в многоугольники, вписанные в эллипс.

Теперь представим себе, что на каждом шаге вычерчивания многоугольника радиус описанной окружности увеличивается на некоторую величину. Тогда при больших N (окружность) мы получим один виток раскручивающейся плоской спирали. Если же мы увеличим конечный параметр счетного цикла до $10N$, то получим 10 витков спирали Архимеда. При небольших N получаются интересные узоры. Однако гораздо более интересен случай (и более симпатичные рисунки), когда спираль скручивающаяся (R уменьшается). При достаточно большом количестве витков спираль скручивается в точку и затем опять начинает раскручиваться, создавая красивый симметричный рисунок. На рис. 3 показана такая спираль для $N=5$. Программа имеет следующий вид:

```

1.1 A "Количество сторон",N:
      S B=2*3.14159/N
1.2 X FT(0.250.60):S R=1:X FCHR(12)
1.3 F K=1,30*N:
      X FV(1.250+100*K*R*FSIN(K*B).
      120-60*K*R*FCOS(K*B)):S R=R-0.015
1.4 G

```



При достаточно больших N мы получим иллюстрацию известной кинематической задачи о движении тела с ускорением, составляющим тупой угол с направлением движения (рис. 3, $N=30$, шаг изменения R равен 0,005, 6 витков).

Из нашей основной программы для N -угольника легко получается программа для вычерчивания спирали типа «винтовая линия» (рис. 3). Для этого, очевидно, нужно на каждом шаге цикла слегка менять координаты центра. Программа для такой спирали будет выглядеть следующим образом:

```

1.1 A "N".N:S B=2*3.14159/N
1.2 X FT(0.250.60):S C=1:X FCHR(12)
1.3 F K=1,10*N:
      X FV(1.250+C+100*K*FSIN(K*B).
      120-C-60*K*FCOS(K*B)):S C=C+0.1
1.4 G

```

Фигура Лиссажу — это результат сложения двух взаимно перпендикулярных колебаний с одинаковыми амплитудами (R), одинаковыми частотами ($K*B$) и постоянной разностью фаз, равной $\pi/2$ (SIN и COS). Разность фаз и амплитуду мы пока менять не будем, а для частот введем два множителя, отношение которых и дает нам отношение частот. Отсюда мы легко получим соответствующую программу:

```

1.1 A "N",N,"M",M,"L",L:S B=2*3.14159/N
1.2 X FT(0.250.60):X FCHR(12)
1.3 F K=1,N:X FV(1.250+100*K*FSIN(M*K*B).
      120-60*K*FCOS(L*K*B))
1.4 G

```

На рис. 4 показаны некоторые из вычерчиваемых ею фигур. Здесь $N=100$ везде, $M=4$, $L=3$; $M=3$, $L=4$; $M=7$, $L=9$. Каждая фигура вычерчивается несколько секунд, и ученик видит сложное колебание во времени, а не готовую траекторию, которая представлена на рисунках. Некоторые ученики пытаются «сбить машину с толку», т. е. задают дробные значения N , M , L . Получаются, естественно, почти такие же фигуры, но незамкнутые, так как количество



колебаний по осям x и y не будет целым. Кстати, это объяснение можно попросить у ученика.

Фигуры Лиссажу у нас получались благодаря тому, что $N > M$ и $N > L$ (квазинепрерывность). Если же эти величины будут сравнимы, то мы получим различные симметричные узоры. При $M \approx L \approx N/2$ получаются различные звезды, показанные на рис. 5 ($N=5$, $M=L=2$; $N=11$, $M=5$, $L=6$; $N=17$, $M=3$, $L=10$).



42

В заключение несколько замечаний.

Первое о фигурах Лиссажу. Автор сначала уклонился от рассмотрения общей задачи, а потом сделал вид, что решение уже получено. Дело в том, что если разность фаз между двумя колебаниями не равна $\pi/2$, а произвольна, то нежелательно брать координаты точки, с которой начинается рисунок, $A(250, 60)$,

так как на рисунке может появиться лишняя линия. Программа в этом случае может принять следующий вид (идея остается, лишь более корректное оформление):

```
1.1 A "N", N, "M", M, "L", L, "H", H:
      S B=2*3.14159/N
1.2 X FT(0.250.120-60*FSIN(H)):
      X FCHR(12)
1.3 F K=1, N: X FV(1.250+100*FSIN(M*K*B),
      120-60*FSIN(L*K*B+H))
1.4 G
```

Второе замечание касается изменения различных параметров исходной программы для получения новых программ. Все изменения рассматривались отдельно. А можно попробовать их различные комбинации, например фигуры Лиссажу в подвижной системе координат или раскручивающуюся спираль с подвижным центром. Можно усложнить все эти картины, приблизив их к моделям реальных физических процессов, если параметры будут изменяться нелинейно.

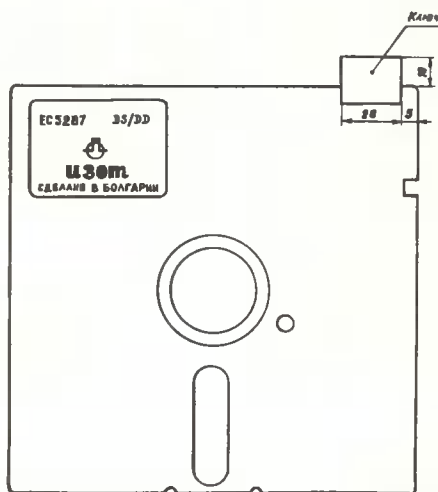
```
1.1 A "N", N, "E", E: S B=2*3.14159/N
1.2 X FT(0.250.60): S R=1: X FCHR(12)
1.3 F K=1.7*N:
      X FV(1.250+100*R*FSIN(K*B),
      120-60*R*FCOS(K*B)): D 1.5
1.4 G
1.5 S R=FCOS(K*B/E)
```

Ю. АЛЯЕВ, В. ОВЧИННИКОВ

Защита НГМД

Качественное проведение занятий в классе КУВТ-86 невозможно без использования дискета. К сожалению, дисководы порой выходят из строя: особенно обидно, когда это происходит из-за неверной установки дискеты.

Предлагаемая защита НГМД предусматривает доработку дискет путем установки на конверты гибких магнитных дисков специального ключа, изготовленного, например, из кусочка фотопленки. Размеры и место установки ключа показаны на рисунке. Доработанные таким образом дискеты вставляются в дисковод только в одном, правильном положении. Кроме того, появляется возможность визуального контроля наличия дискеты в НГМД.



Новости рынка

За последнее время кардинальных перемен на рынке ВТ не произошло. Впрочем, наметились приятные тенденции. Начали дешеветь советские компьютеры, договорные цены, похоже, становятся ценами договора изготовителя и покупателя, а не изготовителя и продавца. Так, БК-0011, в первое время стоивший 9000 рублей (фирменный магазин «Электроника»), теперь продается уже за 6000, причем можно купить его и без периферийных устройств — «всего» за 1900 рублей. Зато «Электроника-85» с цветным дисплеем стоит 22 772 рубля.

Появилась в этом магазине и новая модель — «Дельта-С». Выпускаемый Чебоксарским предприятием «Сеспель» компьютер полностью совместим с «ZX Спектрум»; микропроцессор — Z80A, тактовая частота 3,5 МГц, ОЗУ 48К байт, ПЗУ 16К байт, 8-цветная графика 192×256 точек, цена без периферийных устройств — 1500 рублей.

Итоги конкурса

В прошлом году были подведены итоги первого Всесоюзного конкурса бытовых ПК, проведенного ГКВТИ СССР, ЦК ВЛКСМ, ЦС ВОИР и ЦК профсоюза рабочих радиоэлектронной промышленности. Рассматривалось 19 моделей, разработанных различными предприятиями и организациями, а также самодельщиками. Первое место занял МК-88 (16-разрядный, микропроцессор K1810BM88, ОЗУ 128—256К байт, ПЗУ 16К байт, НГМД 320К байт, разрешение монитора до 640×200 точек, количество цветов — до 16); второе — MC1502 (в основном те же параметры, но без НГМД, ОЗУ 128К байт, ПЗУ до 64К байт, число цветов — до 4). Третье-пятое места заняли восьмиразрядные модели «Вектор 06Ц», «Гуру», «Эльф».

Устроители конкурса констатировали: вредное влияние чрезмерной самостоятельности предприятий в части разработки и выпуска бытовых компьютеров, что приводит к появлению программно несовместимых моделей, снижению их потребительских ка-

Процессор Z80 становится все более популярным. ПЭВМ «Хорус» на его основе, также совместимую со «Спектрумом», выпускает МНПК «Эхо» (Москва): ОЗУ 48К байт, ПЗУ — 16К байт, тактовая частота — 4 МГц, 15 цветов, 196×256 точек, 1200 рублей без периферийных устройств, с НГМД (720К байт) — 3000 рублей. Встречаются рекламные объявления и других изготовителей «Спектрумов».

Интересное сообщение прислал читатель В. В. Здоровик. Во Львове бывает в продаже ПК «Правец-8Д»: процессор 6502, ОЗУ 64К байт, ПЗУ 16К байт, графика 240×200 точек, 12 цветов, 3 канала звука. Цена в комплекте с цветным телевизором-монитором — 1600 рублей. Пожалуй, эта ПЭВМ могла бы стать лидером рынка бытовых компьютеров, если бы продавалась в достаточных количествах.

честв; высокие цены на них, порождаемые высокими ценами на интегральные схемы (в первую очередь памяти) и особенностями существующей системы ценообразования; недостаточность мощностей и культуры производства; нерешенность задач сервисного обслуживания, в том числе распространения программных средств; недостаточность и недоработанность периферийных устройств (отсутствие RGB-входов в телевизорах, дефицит дисководов и т. п.).

Во второй половине текущего года на ВДНХ должна состояться организуемая ГКВТ СССР компьютерная выставка, которая, по замыслу, должна способствовать устранению этих недостатков. На ней смогут установить контакты проектировщики, производители, потребители бытовых компьютеров. А пока предприятия, желающие производить ПК, могут получить в Главном управлении разработок вычислительной техники ГКВТИ СССР техническую информацию о компьютерах — победителях конкурса.

Экономия памяти БК-0010.01

Мизерный размер ОЗУ БК-0010 стал притчей во языцех; к тому же даже в режиме расширения памяти экран из шести строк остается графическим и занимает 4К ОЗУ, хотя соответствующую информацию можно было бы разместить в 256 байтах (64 символа \times \times 4 строки).

44 Однако БК еще долгое время будет самым распространенным бытовым компьютером, поэтому я предлагаю вниманию читателей «ИНФО» (ни в коей мере не претендуя на единоличное авторство) ряд способов экономии памяти БК-0010.

Итак, что делать, если ваша программа выросла до таких размеров, что перестала запускаться? На практике выглядит это так: программа благополучно записывается в память, но при попытке ее запуска выдается одно из следующих сообщений:

СТОП

СТОП в строке N

ОШИБКА 7

ОШИБКА 7 в строке N

Самая частая причина первых двух случаев — попытка присвоить значение символьной переменной до распределения памяти операторами CLEAR или DIM (причина этого осталась непонятой). Исправляется эта ситуация просто, хотя опять же непонятно — достаточно описать до строки остановки любой массив (даже если он не нужен в программе), например DIM A% (1). Кстати, все массивы желательно описать в самых первых строках программы, иначе подобные сюрпризы с остановками могут встретиться и в операторах INPUT, INKEY $\text{\textcircled{X}}$ и т. д.

Если номера строки в сообщении об ошибке нет, нужно включить режим трассировки директивой TRON. Нужный номер будет последним в списке номеров, выданных на экран. Затем режим трассировки должен быть отключен директивой

TROFF, в противном случае при попытке остановить работу программы клавишей СТОП может полностью очиститься память БК.

При появлении «ошибки 7» рекомендуются следующие действия:

определить характер переполнения; выбрать способ уменьшения длины программы;

оценить полученную экономию памяти;

попробовать запустить программу; если предполагается дальнейшее увеличение программы, сокращать все, что можно, до предела, не ожидая переполнения, а при добавлении новых фрагментов регулярно контролировать остаток свободной памяти;

регулярно (при малейших существенных доработках) записывать копии программы на МЛ.

Два последних пункта введены из-за склонности БК не только к зависанию, но и к сбросу программ.

Определить характер переполнения можно с помощью директив ?FRE и ?FRE («»). Если сообщение «ОШИБКА 7» будет выдано в обоих случаях, то, вероятнее всего, переполнен стек. Как правило, это происходит при некорректных выводах из подпрограмм и циклов (с помощью GOTO минуя RETURN и NEXT).

«ОШИБКА 7» в символьных переменных (т. е. выдающаяся при вводе директивы ?FRE («»)) или сообщение «ОШИБКА 14» могут быть устранены перераспределением памяти, т. е. введением в начале программы строки CLEAR MMM

(MMM должно превышать 200; поставьте 250, затем 300 и так — до исчезновения сообщения об ошибке).

Если же это не приводит к желаемому результату, а также при переполнении памяти, выделенной под числовые переменные (?FRE(0)), и малом

числе символьных переменных можно также попытаться перераспределить память, но уже в сторону уменьшения. В CLEAR MMM MMM должно быть меньше 200, например 50 или 10, согласно количеству и величине необходимых символьных переменных.

Распределение памяти сохраняется и после исполнения команды NEW, так что вызвать переполнение может и небольшая программа, если до ее запуска память была перераспределена директивой CLEAR или предыдущей программой.

Во всех остальных случаях приходится обращаться к специфическим способам экономии памяти БК, изложенным ниже в порядке нарастания трудоемкости их применения.

1. Убрать или сократить лишние тексты: комментарии, длинные сообщения, многосимвольные имена переменных. Получаемая экономия памяти — по две ячейки на каждый второй сокращенный символ.

2. Убрать «хвосты» пробелов в программных строках. Они могут появляться при редактировании текстов (в результате использования курсорных клавиш вместо клавиши ЗАБОЙ), при случайном срабатывании клавиши ТАБ и после номеров строк в GOTO, GOSUB, ON и т. д. при ренумерации (RENUM) программы или ее части.

«Хвосты» обнаруживаются по положению курсора (при выводе на экран данной строки в режимах AUTO и «.» — вызов на редактирование) или символа ввода — инверсной буквы J (в LIST при включенном режиме ИСУ).

Устраняются «сбросом конца строки» или клавишей «ЗАБОЙ». Экономия памяти — по две ячейки на каждый второй символ.

Для быстрого просмотра удобно изменять LIST в режиме инверсии символов (AP2/,) или ИСУ. Приостанавливать просмотр можно нажатием СУ/Ю: эта комбинация приостанавливает любые действия компьютера, возобновляя их по нажатии любой клавиши.

3. Если несколько циклов можно оканчивать в одном месте, то это надо делать одной строкой:
NEXT I%, J%, K%

Экономия памяти — шесть ячеек на каждый выброшенный NEXT.

Интересно, что неименованные NEXT занимают на 2—3 ячейки больше (!), чем именованные (NEXT I%).

4. Правильно использовать типы данных. Все целые числа, кроме номеров строк в GOTO, GOSUB, ON и аргументов оператора DRAW, снабдить символом % (удобно использовать ключ KEY N, CHR Q (23 %) + «%» — при его применении символы сами раздвигаются при впечатывании %). Экономия памяти — 4—6 ячеек.

Использовать переменные без чрезмерного запаса точности. Для решения большинства задач достаточно одинарная точность (символ ! после имени).

При изменении типов переменных нужно быть очень внимательным, ведь X, X!, X% для компьютера совершенно разные вещи, и менять имена нужно по всей программе, чтоб не пытаться потом «вытащить» из X то, что было присвоено X! или X%. Грамотнее и надежнее применять этот прием при написании заведомо большой программы сразу — небольшое увеличение длины текста и времени набора обернется изрядной экономией памяти.

Экономия памяти — две ячейки при переходе от X к X! и четыре при переходе от X к X%.

У этого способа есть исключение — глобальные переменные, точнее те из них, которые встречаются в тексте программы много раз. В этом случае, экономия на памяти, отведенной под значение переменной, мы расходуем память на многократное повторение длинного имени, ведь % и ! также занимают место.

5. Необходимо по возможности сокращать количество переменных, у всех локальных циклов, значения параметров которых не употребляется в других частях программы, сократить число параметров до минимума, то же сделать и с локальными переменными; иначе говоря, нужно стремиться использовать одно место в памяти в максимальном числе случаев. Экономия памяти — четыре ячейки при отказе от одной целой переменной (X%), шесть — при отказе от переменной одинарной точности (X!),

десять — при отказе от переменной двойной точности (X или X#).

6. Немало резервов заключено в использовании значений, задаваемых по умолчанию. Все аргументы, стоящие в списках через запятую и не в скобках (исключение — оператор ON), можно не писать в случаях неизменности их значений. При этом надо не забывать ставить нужное количество запятых, если далее по списку следует хоть одно числовое значение (например, коэффициент сжатия у окружности). Этот прием можно применять и в операторах LOCATE, DATA при считывании данных группами: в этих случаях отсутствие аргумента также оставит неизменным его прежнее значение.

46 Полезно помнить, что изображения, цвет которых не указан, будут нарисованы текущим цветом, в котором работает машина.

Последнее важно, например, при организации мигающих или движущихся рисунков и текстов, т. е. при использовании других цветов — мигание и стирание необходимо организовать таким образом, чтобы в любом случае машина оканчивала рисующий блок в рабочем цвете. Иначе возможно исчезновение изображения в результате случайного выхода в режим совпадения цветов изображения и фона.

При рисовании непрерывных линий оператором LINE нужно знать, что оператору графики «помнить» последнюю обработанную точку даже после работы с текстом и очистки экрана CLS. Этим можно пользоваться, например, при рисовании нескольких прямоугольников или рамок в чередующихся кадрах, если есть возможность использовать старые значения координат:

```
CLS
LINE (0%,0%)=(255%,239%),,B
... 'текст
CLS
LINE -(5%,5%),,B
... 'текст
CLS
LINE -(250%,220%),,B
... И Т.Д.
```

7. Имеет смысл «запаковывать» в подпрограммы не только большие фрагменты, но и почти любые строки,

повторяющиеся в программе более двух раз, например подготовку печати текстов, стандартные сообщения или задержки.

Обращение к подпрограмме

AAA GOSUB NNN
занимает 14 ячеек, оператор возврата
BBB RETURN

— десять, а, например,
MMM IF INKEY\$="«» THEN MMM
— 42 ячейки, т. е. уже при втором обращении к подпрограмме сэкономим $(42 \cdot 2) - (42 + 14 \cdot 2 + 10) = 6$ ячеек и по 28 с каждого последующего использования.

8. Использование «блоков» подпрограмм позволяет выполнить несколько подпрограмм с помощью одного обращения, например:

```
100 GOSUB 1000 'Титр, рулон и
      задержка - весь блок подпрограмм
      с одного обращения
200 GOSUB 3000 'Только рулон
300 GOSUB 1010 'Рулон и задержка
...
1000 GOSUB 2000 'Титр
1010 GOSUB 3000 'Рулон
1020 IF INKEY$="" THEN 1020 'Задержка
1030 RETURN
...
2000 ?AT(8%,22%)CHR$(156%)"НАЖМИТЕ
      ЛЮБУЮ КЛАВИШУ!"CHR$(156%) 'Титр
2010 RETURN
...
3000 ?AT(0%,19%)CHR$(19%); 'Рулонный
      сдвиг части экрана
3010 RETURN
```

9. Позиция печати может быть установлена операторами LOCATE X,Y,K и ? AT (X,Y) <Текст>. В зависимости от представления данных объем памяти, используемый этими операторами, сильно меняется. Например,

NN LOCATE X,Y	- 40 ячеек;
NN LOCATE X%,Y%	- 26 ячеек;
NN LOCATE X%,Y%,C%	- 34 ячейки;
NN LOCATE ,Y%,C%	- 30 ячеек;
NN LOCATE , ,1%	- 24 ячейки;
NN ? AT (X,Y)	- 46 ячеек;
NN ? AT (X%,Y%)	- 32 ячейки;
NN ? AT (X%,Y%) AT (X%,Y%)	- 50 ячеек.

Отсюда видно, что использовать LOCATE гораздо выгоднее; кроме того, в этом операторе возможно умолчание. Второй же оператор имеет смысл употреблять только при печати нескольких сообщений одной программой строкой:

```
NN ? AT(5%,5%) " МИГАЮЩИЙ
      ТЕКСТ "AT(5%,5%) " "
```


При необходимости начать печать с новой строки лучше вводить отдельную новую программную строку печати, чем использовать ?AT или LOCATE. Если же надо при этом стереть или пропустить одну-две строки текста, то экономнее для этого использовать запятые, чем оператор TAB. При этом нужно помнить, что операторы печати, встретив запятую, выводят пробелы до ближайшей позиции с номером, кратным 16.

```
10 ?AZ
20 ?
30 ?BZ
```

} - 20 ячеек;

```
10 ?AZ, ,BZ - 40 ячеек;
```

```
10 ?AZTAB(32)BZ - 50 ячеек;
```

```
10 ?AZ, , ,BZ - 48 ячеек;
```

```
10 ?AZTAB(64)BZ - 56 ячеек.
```

10. Относительные координаты в графических операторах обычно записываются короче абсолютных. Сравните:

```
LINE (220,140)-(225,142)
LINE @ (0,-10)-@(5,2)
```

Не забудьте, что операторы графики «помнят» координаты последней обработанной точки даже после работы с текстом и очистки экрана оператором CLS.

11. Оператор DRAW также допускает сокращения.

Во-первых, все единицы, кроме последней, можно не писать; у простых рисунков и схем с кратным шагом ради этого даже выгодно изменить масштаб (не забудьте вернуть прежний в этой же строке, иначе велики шансы забыть сделать это и потом удивляться странным размерам рисунка). Отсутствие числового аргумента оператор DRAW воспринимает как 1 у команд направления и как 0 у команд цвета. Не забывайте: во избежание «ОШИБКИ 5» последней в строке DRAW обязательно должна стоять цифра.

Сравните:

```
DRAW "C0R40D40L40U40C1L40D40R40U40"
DRAW "S10CRDLUC1LDRUS4"
```

Во-вторых, использование относительных координат дает выгоду и в этом операторе, особенно если выбрать направление обхода контура или кривой,

при котором большинство смещений по вертикали будет положительным («плюс» в координате Y в выражении $M+X, +Y$ можно не писать).

В-третьих, кавычки в DRAW можно не закрывать (как, кстати, и в операторе PRINT).

В-четвертых, если есть возможность присвоить литерной переменной текст целого фрагмента рисунка, следует пользоваться выражениями типа

```
DR XR
DR XR+"E7"+AR
```

12. О возможности сокращения имен операторов уже сообщалось (заметка А. Козлова в «ИНФО», № 3 за 1989 г.), но возможности этого приема были раскрыты далеко не полностью. Транслятор Бейсика БК опознает операторы (но не функции) по двум-трем буквам не только в начале строки, но и в любом другом месте программы, хотя и не дописывает их до конца, в отличие от первого служебного слова в строке. Приведем дополненный по сравнению с заметкой А. Козлова список допустимых сокращений команд и операторов (в скобках — необязательные части):

AU(TD)	BE(EP)
BL(OAD)	BS(AVE)
CA(LL)	CI(RCLE)
CLE(AR)	CLOA(D)
CLD(SE)	COL(OR)
CON(T)	CS(AVE)
DA(TA)	DEL(ETE)
DI(M)	DR(AW)
EL(SE)	EN(D)
FI(ND)	FO(R)
GOS(UB)	GOT(O)
IN(PUT)	KE(Y)
LE(T)	LIS(T)
LIN(E)	LL(IST)
LDA(D)	LOC(ATE)
LP(RINT)	MO(NIT)
NEX(T)	OU(T)
OP(EN)	PA(INT)
PO(KE)	PRE(SET)
PRI(NT) /?/	PS(ET)
REA(D)	REN(UM)
RES(TORE)	RET(URN)
RU(N)	SA(VE)
ST(EP)	ST(OP)
TH(EN)	TR(OFF)

Ходят слухи, что сокращения ST и EL срабатывают не всегда, но в нашей практике такого не встречалось.

13. Еще один резерв экономии памяти — пробелы. Обязательны они в единственном случае — после имен переменных двойной точности, ибо самостоя-

тельно транслятор не может определить конец имени переменной. Однако разделителем, кроме пробела, могут служить и некоторые другие символы (! # \$ % () , ;); при наличии любого из них пробел нужен лишь для удобочитаемости.

Сравните три записи — нормальную, сокращенную и сверхкороткую:

```
IF I=0 THEN CIRCLE (100,100),10 ELSE
PRINT A$;"=";I%;K
IF I=0 TH CI @(0,0),10 EL ? A$;"=";I%;K
IFI=0THSIE@(0,0),10EL?A$;"="I%K
```

Сверхкороткая запись выглядит, конечно, не очень эстетично, но, если нет другого выхода, до эстетики ли...

Отметим, что транслятор Бейсика БК ставит по одному пробелу до и после первого служебного слова в программной строке, сколько бы пробелов ни было набрано. Этим можно воспользоваться при форматировании программных сообщений: достаточно после знака «?» ввести столько пробелов, чтобы открывающие кавычки оказались в последней позиции экранной строки; тогда в тексте все переносы будут на своих «законных» местах, а введенные пробелы транслятор уберет.

Исключение — многооператорные конструкции типа IF...THEN?«...»ELSE?«...», где пробелы надо убирать вручную.

Напомним, что способы экономии памяти заложены в порядке возрастания сложности их применения. Некоторые из них, например 12, 13, создают столько неудобств, делают программу настолько трудночитаемой, что их применение имеет смысл лишь при сокращении свободной памяти до нескольких сотен ячеек. В то же время способы 4—6 при работе над заведомо длинной программой желательно применять с самого начала, так как практически невозможно переименовать все переменные сложной программы без массы ошибок; времени на их исправление потом будет затрачено гораздо больше, чем было сэкономлено при написании текста.

В заключение несколько слов о «сюрпризах» транслятора Бейсика BK-0010.01.

1. В операторе DRAW иногда проис-

ходит «выброс» последней линии — она без видимых причин может стать очень длинной. В этом случае достаточно в конце строки DRAW поставить пробел (кавычки не обязательны).

2. При включении в оператор DRAW подстрок или переменных иногда вместо символов печатаются какие-то странные нерегулярные точки (даже в ответ на нажатие клавиши СТОП).

При ближайшем рассмотрении оказывается, что высвечиваются только одна-две «вертикали» символа — машина переходит в «полосатый» цвет. Устранение — «насильное» включение цвета, например AP2/HP/1. Причина — во многих руководствах пропущен символ «точка с запятой», а он в этих случаях обязателен. Корректное включение подстроки выглядит так:

```
DRAW"CI XA$;"
```

Между X и именем переменной подстроки не нужен знак равенства, а точка с запятой должна стоять после \$ хоть в середине, хоть в конце строки.

3. Нельзя делать перенумерацию части программы, если в ней используется оператор ON. Его список меняется только при перенумерации всей программы.

4. Будьте осторожны с режимом трассировки (TRON): если при этом встретится цикл типа

```
NN IF INKEY$="" THENN
```

то при попытке остановить работу клавишей СТОП иногда происходит полный перезапуск Бейсика с не менее полной очисткой памяти. Выход — попробовать сначала СУ/Ю, а потом уже СТОП; еще лучше впредь всегда вводить в программу TROFF.

5. При работе с программами связи БК—ДВК (NET, NET3) также возможны сюрпризы: при загрузке программа читается дважды, причем в БК остается только вторая ее половина (!). При распечатке же на терминал ДВК или на принтер не обнаруживается ничего подозрительного.

На диске может обнаружиться файл длиной 0 блоков. Не пытайтесь стереть его — ничего хорошего из этого не выйдет. При детальном исследовании оказывается, что БК этот файл прекрас-

но читает, правда не то, что там было... А на диске есть еще один файл, начинающийся с этого же блока. Другими словами, этот файл отзывается и на имя, и на «псевдоним». Но «псевдоним», как и положено, не содержит «анкетных данных» (в данном случае — сведений о длине), хотя попытка уничтожения приводит к успеху, как бы вы к нему не обращались.

Такое случается при многократной записи на диск программы под одним именем.

Кроме этого сюрприза NET3 изредка подкидывает и другой — он иногда не читает каталог диска при записи. Кончается это тем, что на диске оказывается записан каталог предыдущего диска и программа вписана на место, которое было свободным на предыдущем диске. Спасать в этом случае программой SOS.SAV можно только текстовые и .ASC файлы. Поэтому не рекомендуется запись на диск директивой CLOAD, а NET3 при смене диска

необходимо перезапускать.

Заканчивая обсуждение сюрпризов NET3, приведем программу, позволяющую читать каталог диска с места ученика.

```

1 *Чтение каталога диска,
2 *созданного так:
3 *R DIR <BK>
4 *CAT=/A/F/C:4 <BK> CУ/C
10 OPEN "TT:CAT.DIR"FORINPUT
20 INPUT #A$
30 ? A$
40 IF NOT (EOF)TH20
50 CLOSE

```

Если NET3 будет запущена не командой, а командным файлом CAT.COM

```

NET3
R DIR
CAT=/A/F/C:4
^C

```

то по выходу из программы связи каталог для БК будет обновляться автоматически. Файл создается в редакторе EDK и запускается командой @ CAT.

М. ЛАРКИН
при участии И. Левенчука, А. Новикова

49

Еще раз о плавающей арифметике Фокала

Публикация в журналах текстов программ для ЭВМ сопряжена с техническими трудностями и приводит к увеличению вероятности опечаток. Не явилась исключением и наша статья «Использование плавающей арифметики Фокала в БК-0010» в № 5 за 1989 г. К счастью, имеющаяся опечатка в тексте программы ПИФ не делает ее полностью неработоспособной в отличие от аналогичной статьи в научно-популярной серии «Вычислительная техника и ее применения», № 5, 1989. На с. 67 в тексте программы ПИФ (столбец 5, вторая строчка сверху) вместо числа 014412 должно стоять 104412. Данная ошибка приводит к тому, что на экран не выдается сообщение об ошибочных ситуациях при выполнении операций Фокала, например ПОПЫТКА ДЕЛЕНИЯ НА НОЛЬ. В остальном программа ПИФ функционирует нормально.

На с. 65 в таблице 4 в последнем столбце необходимо вставить пустую строку после числа 132552, т. е. фактически сдвинуть все числовые значения вниз от этого числа на одну строку.

По вине авторов при подготовке публикации в примере внизу с. 63 исчезла команда ввода символа с клавиатуры EMT 6. Первоначально данная программа в мнемоническом виде должна была выглядеть следующим образом:

```

MOV #A, R1 : Поместить адрес сим-
              : вольной строки в R1
40: EMT 6    : ввести код
              : с клавиатуры в R0
EMT 16      : вывести эхо на экран

```

К сожалению, в данном случае, чтобы исправить программу, необходимо вставить команду. Это приведет к сдвигу чисел во всем примере. Однако если в третьей строке сверху команду EMT 16 заменить на EMT 6 (числовой код 104006), то программа будет работать, правда «вслепую», т. е. без выдачи вводимого числа на экран.

На с. 64 в центре четвертая строка примера должна содержать число 002050 вместо 002052.

В процессе использования плавающей арифметики Фокала выяснились некоторые

особенности. Оказалось, что функция FSQT требует для своей работы копию сумматора в рабочих областях Фокала. Поэтому перед использованием FSQT необходимо выполнить копирование сумматора операцией FPUT @ # 1704.

Использование операции FREAD для перевода чисел из символьного вида в двоичный имеет один нюанс для случая отрицательного числа. В статье на с. 64, второй абзац сверху, сказано, что при переводе отрицательного числа операция FREAD устанавливает разряд условия N. Однако это не так. Разряд N устанавливается автоматически при загрузке в регистр R4 первого байта переводимого числа, который в этом случае является минусом. Чуть ранее в статье объяснялось, что код минуса в Фокале равен 202. С точки зрения архитектуры ЭВМ этот байт является отрицательным числом, что приводит к установке признака N. Поэтому в примере внизу с. 63 необходимо поменять местами две команды:

```
002064 112737   MOV# #20, @#1723 ;Обья-
002066 000020   : зательно заполнить
002070 001723
002072 112304   MOV# (R3)+, R4   ;Обья-
: зательно поместить
: первый символ в R4
```

Правда, разряд N установится также в случае использования знака плюс (код 201) в начале строки с положительным числом, поэтому здесь следует соблюдать осторожность и придерживаться обычной практики кодирования положительного числа без явно заданного знака «плюс».

Некоторые программисты в своей работе часто используют операцию FPRINT. При

этом их интересует возможность изменять значение формата выводимых чисел.

Значение формата, как известно, состоит из двух частей: первая указывает общее количество десятичных цифр в записи выводимого числа, а вторая — количество знаков в мантиссе. В нотации Фокала это записывается, например, в виде %8.04. Перед использованием операции FPRINT в регистр R2 помещается требуемый формат, при этом старший байт содержит первую часть формата. В младший байт необходимо поместить удвоенное число позиций мантиссы, включая точку. Для приведенного в статье формата %8.04 регистр R2 содержит (числа восьмеричные):

старший байт = 10
 младший байт = $(4+1) \times 2 = 12$
 $R2 = \langle 10 \rangle \langle 12 \rangle = 4012$

Многие пользователи проявили заинтересованность в использовании плавающей арифметики на ЭВМ с Бейсик-интерпретатором. К сожалению, авторы не занимались изучением заложенных в микросхеме Бейсика средств плавающей арифметики. Однако ими разработан пакет подпрограмм, который позволяет использовать механизм псевдокоманд плавающей арифметики Фокала на любом БК, в том числе и на ЭВМ с Бейсиком. В состав пакета входит также библиотека стандартных математических функций.

В заключение авторы хотят выразить благодарность всем пользователям, уже высказавшим свои замечания, а также тем, кто в дальнейшем сообщит об обнаруженных неточностях. Все замечания и предложения просим направлять по адресу: 226063, г. Рига, ул. Кенгарага, 8. НИИ физики твердого тела.

С. ГВОЗДЕВ, Г. ЭРНЕСТСОНС

Универсальный каталогизатор

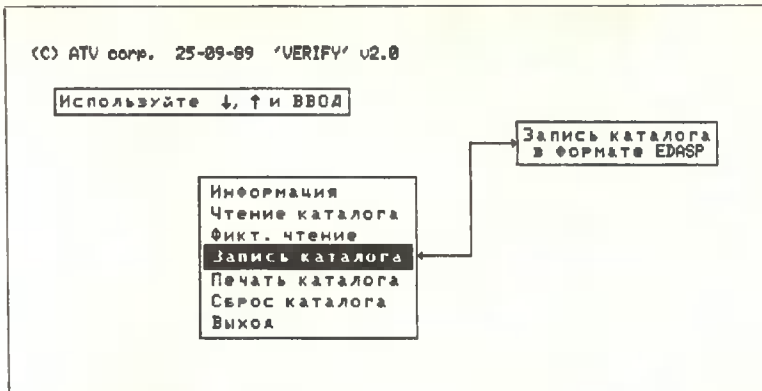
Широкое распространение БК породило огромное количество программных средств для него. В связи с этим возникла необходимость в программах, облегчающих процесс каталогизации. Такие программы должны обладать следующими функциями: чтение файлов с ленты, вывод на дисплей имени, адреса загрузки в ОЗУ и длины текущего файла, подсчет количества копий на магнитной ленте (МЛ), регистрация наличия ошибок при чтении с МЛ, чтение имен файлов без занесения в каталог. При этом необходимо предусмотреть возможность записи полученного каталога на МЛ в формате какого-либо редактора и возможность последующей печати каталога на принтере. Существующие

программы каталогизации обеспечивают только отдельные из перечисленных функций.

В описываемом каталогизаторе VERIFY реализована вся совокупность перечисленных функций, а также добавлен ряд сервисных возможностей, облегчающих работу пользователя. Имеется шесть основных режимов работы:

информация (выдача на экран кратких сведений о программе);

чтение оглавления ленты (чтение имен файлов, проверка качества записи файлов, подсчет числа копий и создание каталога); запись каталога (запись каталога на МЛ в формате редактора EDASP);



печать каталога (распечатка на печатающем устройстве каталога прочитанных файлов);

фиктивное чтение (чтение имен файлов для поиска заданного места на МЛ);

сброс каталога (подготовка программы для создания нового каталога).

В режимах записи и печати каталога для каждого прочитанного файла указываются адрес загрузки, длина, наличие ошибок чтения и число копий на МЛ. В конце каталога подсчитывается и регистрируется общее число файлов.

Программа VERIFY отличается от других каталогизаторов простотой управления: все операции осуществляются в режиме меню, на экране постоянно находится подсказка, объясняющая функции текущего режима. Одно из состояний экрана во время работы с программой представлено на рисунке. Программа написана на ассемблере и занимает примерно 5,5К байт.

```

TRA: CLR CRT      ; УСТАНОВИТЬ ТЕКУЩИЙ
      JSR PC, MNU  ; РЕЖИМ В 0
      JSR PC, CUT  ; ОТОБРАЗИТЬ НА
                  ; ЭКРАНЕ МЕНЮ ДЛЯ
                  ; РАБОТЫ И ВСПОМОГА-
                  ; Тельное окно
      ADD #7, R1   ; НАРИСОВАТЬ СТРЕЛКУ
                  ; ОТ ТЕКУЩЕГО РЕЖИМА
                  ; ДО ВСПОМОГАТЕЛЬ-
                  ; НОГО ОКНА
      EMT 24
      MOV #WNW, R1
      CLR R2
      EMT 20
      MOV #ATV, R2 ; ВЫВОД НОМЕРА
                  ; ВЕРСИИ В
                  ; СЛУЖЕБНУЮ СТРОКУ
      CLR R1
      JSR PC, MEN
TR5: JSR PC, CUT  ; УСТАНОВИТЬ КУРСОР
                  ; СПРАВА ОТ ТЕКУЩЕГО
                  ; РЕЖИМА
      MOV #SL, R1  ; (ТЕКУЩИЙ РЕЖИМ
                  ; ОТЛИЧАЕТСЯ
                  ; ИНВЕРСНЫМ
                  ; ИЗОБРАЖЕНИЕМ)
      CLR R2
      EMT 20
      JSR PC, SEE  ; НАРИСОВАТЬ СТРЕЛКУ
                  ; ВЫВЕСТИ ДОПОЛНИ-

```

```

; ТЕЛЬНУЮ ИНФОРМАЦИЮ
; 0 РЕЖИМЕ
; ПРОЧИТАТЬ КОД
; НАЖАТОЙ КЛАВИШИ
; АНАЛИЗ НАЖАТОЙ
; КЛАВИШИ
JSR PC, KEY
CMP R0, #12
BEQ K12
CMP R0, #32
BEQ T01
CMP R0, #33
BNE TR6
; НАЖАЛИ СТРЕЛКУ ВНИЗ
JSR PC, CUR ; ВЫВОД В НОРМАЛЬНОМ
; (НЕИНВЕРТИРОВАННОМ)
; ВИДЕ ТЕК. РЕЖИМА
; СТРЕТЬ И ПРОДЛИТЬ
; СТРЕЛКУ ОТ ВСПОМО-
; ГАТЕЛЬНОГО ОКНА
; ДО НОВОГО ТЕКУЩЕГО
; РЕЖИМА
JSR PC, CUT
MOV #PPS, R1
CLR R2
EMT 20
INC CRT ; СДЕЛАТЬ ТЕКУЩИМ
; СЛЕДУЮЩИЙ РЕЖИМ
CMP CRT, #7 ; БЛОКИРОВКА ОТ
; НЕКОРРЕКТНОГО ВВОДА
BMI T02
DEC CRT
T02: MOV #234, R0 ; УСТАНОВИТЬ РЕЖИМ
; ИНВЕРСИИ СИМВОЛА
EMT 16
JSR PC, CUR ; ОТОБРАЗИТЬ В ИНВ.
; ВИДЕ ТЕКУЩИЙ РЕЖИМ
MOV #234, R0 ; УСТАНОВИТЬ
; НОРМАЛЬНЫЙ РЕЖИМ
; ВЫВОДА
EMT 16
BR TR5 ; ВОЗВРАТ
; НАЖАЛИ СТРЕЛКУ ВВЕРХ
T01: JSR PC, CUR ; ВЫВЕСТИ В
; НОРМАЛЬНОМ ВИДЕ
; ТЕКУЩИЙ РЕЖИМ
JSR PC, CUT ; УКОРОТИТЬ СТРЕЛКУ
MOV #231, R0 ; ОТ ВСПОМОГАТЕЛЬНОГО
EMT 16 ; ОКНА ДО НОВОГО
; ТЕКУЩЕГО РЕЖИМА
DEC CRT ; СДЕЛАТЬ ТЕКУЩИМ
; ПРЕДЫДУЩИЙ РЕЖИМ
BGE T03 ; БЛОКИРОВКА ОТ
; НЕКОРРЕКТНОГО ВВОДА
CLR CRT
T03: BR T02 ; НАРИСОВАТЬ СТРЕЛКУ
; И ВЕРНУТЬСЯ
; НАЖАЛИ <ВВОД>
K12: TST CRT ; ВЫБОР НУЖНОЙ ПОД-

```

```

: ПРОГРАММЫ В ЗАВИСИ-
: МОСТИ ОТ ЗНАЧЕНИЯ
BNE KD1 : ПЕРЕМЕННОЙ CRT
: (НОМЕР ТЕК. РЕЖИМА)
JSR PC, HLP : CRT=0 ИНФОРМАЦИЯ
BR TR6 : 0 ПРОГРАММЕ

KD1: CMP CRT, #1
BNE KD2
JSR PC, RDR : CRT=1 ЧТЕНИЕ
BR TRA : ФАЙЛОВ С ЛЕНТЫ

KD2: CMP CRT, #2
BNE KD3
JSR PC, CAT : CRT=2 ФИКТИВНОЕ
BR TR6 : ЧТЕНИЕ

KD3: CMP CRT, #3
BNE KD4
JSR PC, SAV : CRT=3 ЗАПИСЬ
BR TR6 : КАТАЛОГА НА МЛ

KD4: CMP CRT, #4
BNE KD5
JSR PC, PRI : CRT=4 ПЕЧАТЬ
BR TR6 : КАТАЛОГА

KD5: CMP CRT, #5
BNE KD6
JSR PC, NWT : CRT=5 СБРОС
BR TR6 : КАТАЛОГА
: CRT=6 ВЫХОД В МОНИТОР
KD6 JSR PC, SUR : ЗАПРОС ПОДТВЕРЖДЕ-
: НИЯ У ПОЛЬЗОВАТЕЛЯ

TST R0
BNE Q00
JMP TR6
Q00: EMT 14 : СБРОС СИСТЕМЫ
HALT : ОСТАНОВ

```

```

:
CRT: : #0
SL: : A: <-----+
: B: 0
PPS: : A: !
: B: 34, A: !
: B: 34, A: +-----
: B: 276, B: 0
ATV: : A: (C) ATV corp.
: E
: 25-09-89 'VERIFY' V2.0

```

Имя файла	Адрес	Длина	Ошибки	Число
ONEAN	001000	037000	Нет	02
FOKEY	030000	000434	Нет	02
FOKEY 2.0	033000	003310	Да	01
FOKEY 2.0	033000	003310	Нет	01
СНАРЯД	001140	002210	Нет	02
RALLI-A	001140	004126	Нет	02
FANTOM2	001752	016106	Да	01
FANTOM2	001752	016106	Нет	01
TOR	001752	004572	Нет	02
LOOK&PRINT	000732	005434	Нет	02

Остаток число файлов 0016

На одной из распечаток приведен фрагмент программы, реализующий диалог с пользователем в режиме меню, на другой — пример каталога, созданного программой.

А. ТЕРЕХОВСКИЙ

Убыстрение поиска файла

Поиск нужного файла на магнитофонной кассете (МК) отнимает, как правило, много времени. Эта процедура несколько упрощается при наличии счетчика расхода ленты. При отсутствии же счетчика поиск усложняется многократно, потому что вести его приходится методом проб и ошибок со все-



ми вытекающими из этого последствиями...

Между тем существует достаточно простой способ ускорить эту работу.

Первое условие его реализации — уточнение имеющихся на кассете измерительных линеек. На применяемых мною кассетах их заменили этикетки, показанные на рисунке. Шаг разметки равен 1,5 мм.

Второе условие — запись двух копий для каждого файла с разной плотностью записи и со строго установленной их последовательностью для всех записываемых файлов; при этом в списке файлов отмечаются положение диска МЛ, сторона и номер кассеты.

При соблюдении этих условий поиск нужного файла значительно ускоряется за счет более точного определения его местоположения на МЛ и разной точности записанных копий.

А. БАРСУКОВ

От редакции. Предлагаемый способ не универсален: его второе условие является одновременно и его ограничением. Действительно, соблюсти его можно только в случае

копирования файлов с помощью некоторых копировщиков, позволяющих менять плотность записи. Использование более точных линеек теоретически предпочтительнее для

приблизительной ориентации на ленте, но доступность их для обзора в подавляющем большинстве магнитофонов, мягко выражаясь, оставляет желать лучшего.

Программа «Музыкальная заставка»

Чтобы разнообразить работу на БК-0010 по озвучиванию программ, написанных учащимися на Бейсике, был разработан метод создания музыкальных подпрограмм (заставок) к этим основным программам. Музыкальные операторы в Бейсике (Вильнюс, 1987) отсутствуют, поэтому пришлось написать подпрограмму в машинных кодах.

В приводимой в качестве примера программе музыки генерирует подпрограмма на машинном языке, определенная как функция пользователя. Ее текст записан в операторе DATA. Она обрабатывает символьную строку A\$, в которой закодирована мелодия.

Мелодия подбирается и кодируется строками вида

```
A$="SSDSE5F5G5A5B5@5"
  до ми соль си
  ре фа ля до
```

Здесь буквами обозначаются соответствующие ноты, а цифрами — длительности нот. Каждая стоящая справа цифра (длительность) соответствует стоящей слева букве (ноте). Длительность может меняться от 1 (максимальная) до 9 (минимальная).

Набрав текст программы и запустив ее (обычным RUN), вы услышите несколько известных мелодий.

```
10 *** GSB-подпрограмма ***
20 CLS
30 L=FEEK(&D2014)
40 DEF USR=L
50 DATA 7617,60,4928,5599,240,&D177716,
  32257,4928,5599,144,&D177716,32257,
  32331,6080,&D177716,135,31,350,35,
```

Знаете ли вы, что...

младший байт ячейки ОЗУ с номером 36 управляет режимом подчеркивания на мониторе? Если в нем записан 0, то режим выключен, иначе режим включен. Подача кода ПОДЧ в драйвер монитора меняет содержимое этого байта с 0 на 255 и с 255 на 0; если же в нем содержится число от 1 до 254, то режим подчеркивания будет включен и не будет управляем кодом ПОДЧ. Старший же байт этой ячейки таким же об-

```
310,37,290,42,260
60 DATA 48,220,24,455,27,405,225,500
70 RESTORE 50
80 FOR L=L TO L+62 STEP 2
90 READ B
100 POKE L,B
110 NEXT
120 POKE L,600
130 INPUT "MELODY (1-5)";N%
140 ON N% GOSUB 220,240,260,280,300
150 K=1
160 H=(ASC(MID$(A$,K,1))-63)*4
170 POKE L,PEEK(L-H+2)*
  (58-ASC(MID$(A$,K+1,1)))
180 I=USR(PEEK(L-H))
190 K=K+2
200 IF K<LEN(A$) GOTO 160
210 GOTO 130
220 A$="F6E6D6C6G6G6G6F6E6D6C6G6G6F6
  A6A6F6E6G6G6E6D6F6F6D6C6C6"
  Веселые гуси
230 RETURN
240 A$="C6C6F3G6A6G6F6C6C6F3G6A6A6G"
  Пусть всегда будет солнце
250 RETURN
260 A$="BVB8VAV8G8G8G8V8B8B8V8A8G4F8
  FBVE8E8D8D8E8E8F8"
270 RETURN
280 A$="G4E4G4E4D7E7F7D7E5C4G4E4G4E4
  D7E7F7D7E4A4F4A4A4F4E7F7G7E7F4
  D4A4F4A4F4E7E7G7E7F4"
  Мама, мама, что я буду делать
290 RETURN
300 A$="D6F6A6F6G6F6E6A3G3D2"
  Подмосковные вечера
310 RETURN
```

53

Если необходимо изменить темп звучания мелодии, то надо в строке 60 после кода 500 поставить число от 10 до 200, а в строке 80 заменить число 62 на 64.

А. БУНЦЕЛЬМАН

разом управляет режимом инверсии. Аналогично функционируют:

32 (старший байт) — управляет режимом инверсии экрана;

34 (младший байт) — режимом расширения памяти;

38 (младший байт) — режимом ИСУ;

38 (старший байт) — режимом БЛР;

40 (младший байт) — режимом ГРАФ;

40 (старший байт) — режимом ЗАП;

42 (младший байт) — режимом СТИР;
 42 (старший байт) — режимом 32/64
 символа в служебной строке;
 44 (младший байт) — режимом ПОДЧ
 в служебной строке;
 44 (старший байт) — режимом инверсии
 символа в служебной строке;
 46 (младший байт) — включением/выключением курсора.

Младший байт ячейки 32 управляет числом символов в строке (0 — 64, 1 — 32).

В ячейке 88 обычно хранится число 1120. Если сюда занести 0, то блокируется клавиша К1 (первый ключ). Здесь как-то закодированы длина и начальный адрес текста первого ключа. Ключами 2—10 аналогично управляют ячейки 90—106.

Ячейка 112 содержит адрес, с которого будет выводиться на экран следующий символ. Так, если занести сюда 40 000₈, то Ок БК выдст в левом верхнем углу экрана (предварительно нужно нажать 2 раза клавиши AP2+СБР). Таким образом можно занести любой текст в любое место памяти (не обязательно экранной; можно занести слово и в область с 0₈ до 40 000₈).

Ячейка 114 (обычное содержимое 2) — это количество пробелов между символами, умноженное на 2. Так, если занести сюда 4, то вся информация будет отображаться на

экране с одним пробелом между соседними буквами.

Ячейка 116 (обычное содержимое 1536) — это число строк экрана, доступных пользователю, умноженное на 64. Если сюда занести 128, то можно будет пользоваться лишь двумя строками.

Ячейка 132 (обычное содержимое 1024) задает смещение начала строки от левого края экрана. Попробуйте занести сюда 1000₈ и набрать затем несколько символов.

В ячейках 134 и 136 закодировано число строк, стираемых по коду 12 или клавишей СБР; как именно — установить не удалось. Отменяется двукратным нажатием клавиш AP2+СБР.

Каждый символ в режиме 32 символа в строке представляется в матрице 16×10 точек (16 столбцов, 10 строк). Перед тем как вывести на экран очередную строку символа, БК производит операцию логического сложения этой строки и ячейки 138; результат сложения выводится на экран. Если занести сюда, например, &B11001100 11001100, то все символы будут в красную полоску. Отменяется оператором COLOR. Аналогично используется ячейка 140, но производится логическое умножение.

И. КАНИВЕЦ

Как совместить

высококачественный цветной дисплей с доступным для вас процессором?

Цветной видеоконтроллер IMSG 300, разработанный фирмой «Инмос», навсегда избавит вас от этих проблем. На одном его кристалле размещены графический контроллер, программируемый видеотаймер, схема управления видеопамятью, сдвиговый регистр и 8-битный цифроаналоговый преобразователь для формирования видеосигнала. Но главное достоинство видеоконтроллера — программируемость. Это качество позволяет пристыковать с его помощью любой доступный вам видеомонитор практически к любому необходимому процессору. Видеоконтроллер имеет 32-битный мультиплексный порт, работающий со скоростью до 110 МГц, и позволяет формировать 24-битный пиксел, что дает возможность иметь цветовую гамму, ограничиваемую только возможностями вашего видеомонитора.

**ЧТО
МОЖЕТ
ЭВМ**

Наконец-то появился

компьютер, имеющий полное право называться карманным. Модель «Portfolio» фирмы «Атари компьютер» весит 450 г и по размерам не превышает стандартную видеокассету. В ней использован процессор фирмы «Интел» 80С88 с тактовой частотой 4,92 МГц; объем ОЗУ — 128 Кбайт, возможно расширение до 640К байт; ПЗУ — 256К байт, включает текстовый процессор, электронную таблицу типа «Лотус», простую СУБД. Экран на жидких кристаллах вмещает 8 строк по 40 знаков. На миниатюрной клавиатуре поместилось 63 клавиши. Вместо дисков в качестве внешней памяти используются компактные карты памяти (размером с игральную карту) на 32К или 128К байт. Батареек хватает на 4—6 недель

(конечно, имеется в виду не непрерывная работа компьютера, а «нормальный деловой» режим).

Операционная система «Portfolio» близка к MS DOS. Обмен данными с другими компьютерами возможен через модем, а с IBM PC еще и через дополнительный драйвер карт или специальный соединительный кабель. Есть последовательный и параллельный порты.

Цена новинки — 400 долларов.

Осторожно: «Лотус»!

Весьма полезная программа «Лотус 1-2-3» (электронные таблицы) завоевала большую популярность у пользователей ПЭВМ. Ее следующая версия еще более совершенна. Кто же откажется сменить хорошее на лучшее? Однако «Лотус 1-3-3» может доставить крупные неприятности. Она бывает заражена малозаметным вирусом, который при каждом использовании программы меняет на небольшую величину одно из чисел в одной из таблиц. Обнаружить это, сами понимаете, нелегко, а потенциальный вред...

«Альтернатива» существует

Альтернатива — 1) необходимость выбора между взаимоисключающими возможностями; ... 2) каждая из исключаящих друга возможностей. (Словарь иностранных слов. М.: Рус. яз., 1984.)

В связи с публикацией в нашем журнале (1989, № 5) рекламного объявления о пакете учебных программ «Альтернатива» для КУВТ-86 читатели задают вопросы об этой программной системе. Воспользовавшись пребыванием в Москве руководителя группы ее разработчиков А. Н. Гриценко, мы попросили его ответить читателям.

— *Алексей Николаевич, какова область применения пакета «Альтернатива»?*

— Наш пакет предназначен в первую очередь для преподавания курса ОИВТ с помощью КУВТ-86 всех типов и модификаций. Все программы пакета составлены в кодах БК-0010, поэтому «Альтернатива» одинаково работает в Фокал- и Бейсик-классах.

— *А каковы другие особенности программной реализации и пользовательского интерфейса пакета?*

— Прежде всего это программирование в кодах. Мы считаем заведомо непригодными для учебного процесса, по крайней мере на КУВТ-86, любые программы, написанные на языках более высокого уровня, чем машинный, напри-

мер на Бейсике. В частности, мы считаем весьма неудачной разработкой известную систему «Рига».

Уже упомянутый монитор обеспечивает все пересылки по сети КУВТ, в том числе пересылку в РМП, иницируемую запросом программы, работающей на РМУ, и рассылку информации с РМП по инициативе учителя — то, чего так сильно не хватает в КУВТ-86 с Бейсиком.

Кроме того, использована система меню, обеспечивающая ученику удобный, легкий доступ к любой из программ пакета.

Большинство наших программ допускают использование манипулятора «джойстик». Это устройство дублирует стрелочные клавиши БК и клавишу «ввод». Благодаря этому делается более дружелюбным интерфейс и резко снижается износ клавиатуры БК.

— *Расскажите, пожалуйста, об основных компонентах «Альтернативы» поподробнее.*

— О базе данных BASE в журнале уже рассказывалось*. О редакторе WORDS достаточно сказать, что он совместим с WORDSTAR по командам, и ученик, овладевший им, впоследствии сможет применить полученные навыки в работе с профессиональной ВТ. Кроме того, в нашем редакторе заложена уникальная и необходимейшая для КУВТов

* В № 6 за 1989 г.

возможность: он допускает свободное создание алфавитов, иначе говоря, кроме кириллицы и латиницы в него можно ввести любой национальный алфавит и с легкостью переключаться на него. После этого на экране БК можно набирать текст на соответствующем языке. Естественно, созданный текст можно записать на диск или распечатать на принтере.

PAINT — графический редактор, подмножество известного редактора PAINTEP для «Ямахи MSX-2»; имеет графическое меню, управляется джойстиком или «мышью», допускает запись картинок на дискету, распечатку на принтере.

— *Каковы главные особенности пакета?*

— Нашей основной задачей было превратить программное обеспечение современных ПЭВМ в предмет для самостоятельного изучения. Мы считаем нездоровой ситуацию, когда изучение информатики в школах в значительной части сводится к изучению примитивных языков программирования (Бейсика или Фокала), и пытаемся ее изменить.

Пакет включает все необходимые компоненты программного обеспечения современной ПЭВМ: текстовый редактор (старая его версия EDASP была описана в «ИНФО», № 3 за 1989 г.; сейчас подготовлена новая версия, аналог известного редактора WORDSTAR для IBM-совместимых ПЭВМ), электронную таблицу (аналог известной SuperCalc), базу данных.

В него включены также исполнитель РОБОТ, близкий к описанному в учебнике Кушниренко, версия Е-практикума, графический редактор и графический конструктор, музыкальный редактор, клавиатурный тренажер и др.

— *Видевший работу вашего редактора WORDS и электронных таблиц CALC должен согласиться, что они пригодны для профессиональной работы. При необходимости обмена данными с внешней памятью пересылка по сети идет очень быстро, возникает даже впечатление, что диск не читается. Как это удается?*

— Наш монитор «Виртуальный диск»

использует память ДВК в качестве электронного диска. Все программы, на которые приходили запросы с БК, во-первых, пересылаются на БК, во-вторых, сохраняются в ОЗУ ДВК. Если приходит запрос на программу, находящуюся в ОЗУ ДВК, что весьма типично для учебных занятий, с диска она не считывается, а посылается прямо из ОЗУ.

— *Это, конечно, весьма эффективное решение, но проблема скорости пересылки, особенно при одновременном поступлении нескольких запросов, видимо, остается?*

— Монитор способен осуществлять параллельную пересылку файлов в разные БК. Благодаря этому при пересылке, скажем, одного файла на два БК время пересылки увеличивается не на 100 %, а лишь на 15—20 %. В целом скорость пересылки примерно соответствует скорости пересылки в фокаловском классе через программу PUTP.

CALC — электронная таблица профессионального уровня. Вообще, все компоненты нашего пакета — это настоящие программы, а не демонстрационные имитации, как в системе «Рига». Наши допускают настоящее профессиональное использование. Даже в клавиатурном тренажере используются осмысленные тексты на различных языках, в том числе и на языках программирования, которые может создать сам учитель, и при загрузке тренажера они автоматически будут загружаться вместе с ним... За эргономичностью, удобством программ мы следим, это наша большая забота.

— *А какие есть программы для поддержки других предметов?*

— Наш пакет ориентирован в первую очередь на поддержку курса ОИВТ. Серьезных программ по предметам, соответствующих высоким критериям, пока существует немного.

— *Расскажите о разработчиках пакета.*

— Над пакетом работала группа программистов и методистов. Работаем мы в Барнауле, до последнего времени — в качестве кооператива при кабинете информатики Алтайского краевого ИУУ. Сейчас кооператив преобразуется в государственное предприятие по произ-

водству программных средств при ИВТ АПН СССР.

Многие программы создавались другими разработчиками, зачастую независимыми. Однако, встретив достойную программу, мы всегда ищем ее разработчика, вступаем с ним в переговоры и, урегулировав вопросы авторского права, включаем ее в свой пакет. Мы не считаем возможным впасть в амбицию и лишать пакет хорошей программы по той лишь причине, что она сделана не в Барнауле. Каждая хорошая программа должна поступить на вооружение народного образования. Другое дело, что ее разработчик должен быть соответствующим образом вознагражден.

Кроме того, мы проводим линию на расширение нашей группы: всех программистов, которые делом доказали свой высокий класс, мы приглашаем к сотрудничеству. Например, с известной группой ASP из Зеленограда заключен договор на создание для «Альтернативы» Турбо-Паскаля.

— *Что «физически» вы предлагаете покупателям и по какой цене?*

— Пакет поставляется на восьми дискетах. На первом записана ОС RT-11 версии 5.1 (мы постарались включить в ОС все последние достижения). На втором — системные программы для учебного процесса. На третьем и четвертом — комплект игровых программ для БК. На пятом — подробнейшая документация на все программы пакета; получив пакет, ее можно распечатать на своем принтере. В нее входят кроме описаний всех программ описание ОС, КУВТ-86 и методические рекомендации по использованию пакета в учебном процессе. Шестой диск содержит учебные программы по отдельным предметам школьного курса, седьмой — методические указания, восьмой — системные и игровые программы для ДВК.

Диски от копирования не защищены: мы считаем, что периодическое их копирование является необходимым элементом работы учителя на КУВТ-86. Получение нашего пакета предполагает, что с пользователем заключается письменный договор, строго ограничивающий его право на копирование. Пакет

может быть приобретен для использования в одной школе, тогда его стоимость составит 1000 рублей по перечислению. Пакет для всех классов КУВТ-86 (настоящих и будущих), установленных в городе без районного деления, может быть приобретен за 3000 рублей (с правом копирования для любого учебного заведения этого города).

Для города с районным делением цена пакета 5000 рублей, для области, автономной республики или союзной республики без областного деления — 10 000 рублей.

— *Цены, я думаю, будут восприняты должным образом; они намного ниже сложившихся для подобных пакетов.*

— Да, цены учитывают реальные возможности школ. Препятствием ни для кого они пока не послужили, иначе говоря, кто видел наш пакет, тот его купил. Конечно, плохо то, что у нас до сих пор не отработана торговля программным обеспечением для ПЭВМ, — за это мы заплатили многими годами застоя в этой области. В области ППС, в частности, тремя годами: многие КУВТы даже сейчас остаются практически «голыми». Систему торговли нужно отработать, нужно твердо усвоить, что хорошее программное обеспечение стоит денег; только это позволит оградить школы от людей, спекулирующих на этом дефиците или предлагающих негодное программное обеспечение.

— *«Альтернатива» активно обновляется. Нужно ли для получения новинок заново покупать пакет?*

— Мы хорошо понимаем эту проблему и поэтому предлагаем нашим клиентам авторское сопровождение пакета. Заключается договор о том, что мы обязуемся не менее двух раз в год обновлять пакет, т. е. поставлять новые программы и улучшенные версии старых. Абонентская плата за сопровождение, видимо, не будет превышать 50 % стоимости первоначальной поставки в год. Заключившие с нами подобный договор в 1990 г. получают, например, Пролог нашей разработки (мы намерены превзойти разработку Григорьева), Турбо-Паскаль, новейшую систему управления базой данных — аналог известной

dBASE II для IBM PC,— и некоторые другие программы, в том числе игровые.

Для заключения договора на поставку пакета и/или его последующее авторское обслуживание необходимо прислать нам заявку по адресу, указанному в нашем объявлении. В ответ мы вышлем необходимые для заключения договора бланки и счета для оплаты по перечислению. После выполнения всех формальностей и оплаты счета мы немедленно вышлем диски с «Альтернативой». Если какие-то диски не читаются на дисковом заказчике, мы заменяем их. При желании покупателя мы комплектуем наш пакет манипуляторами «джойстик» (60 рублей штука плюс небольшая наценка на программы за их адаптацию к манипуляторам). Подчеркиваю: только комплектуем пакет, джойстиком мы не торгуем!

58

— *А как обстоят дела с перенесением вашего пакета на КУВТ УКНЦ?*

— Это одна из наших неотложных задач. Работа только начата, и завершить ее мы сможем не ранее июля 1990 г. Все заявки, которые нам поступили на «Альтернативу» для УКНЦ, мы поставили на учет, и, как только пакет будет готов, мы вышлем нашим заказчикам необходимые для его приобретения документы. Цены, видимо, будут аналогичными. Пакеты «Альтернатива» для КУВТ-86 и КУВТ УКНЦ будут совместимы между собой по интерфейсу, что резко облегчит методическое сопровождение. Нельзя допускать, чтобы разные терминальные классы требовали различной методики преподавания.

— *И каков же на сегодня коммерческий успех вашего пакета?*

— Наша основная задача не коммерция, а улучшение программного обеспечения КУВТов, для чего мы стремимся расширить коллектив разработчиков, обеспечить их персональной вычислительной техникой на основе аренды. Мы с радостью примем в наши ряды программистов, имеющих опыт высококлассной работы на ассемблере для ЭВМ серии СМ, ДВК, БК, УКНЦ (место жительства и род занятий не играют для нас роли). Мы верим, что, создав

достойное программное обеспечение для советской вычислительной техники, можно вывести ее на мировой уровень. По крайней мере учебной ВТ.

Что же касается коммерции... Мы, например, демонстрировали наш пакет в Московском городском ИУУ, и в результате Главное управление народного образования Москвы приняло решение о закупке пакета — такое впечатление пакет произвел на всех присутствовавших учителей. Все школы Алтайского края получили пакет бесплатно. К настоящему моменту мы выполнили несколько десятков заказов для различных территорий — от Магаданской области до Прибалтики.

Оптовая закупка нашего пакета органами народного образования и заключение договора об авторском сопровождении может сэкономить значительные суммы школам территории и дать им качественное программное обеспечение.

— *Не проще ли поставлять ваш пакет через завод-производитель КУВТ-86?*

— Для этого пакет должен быть закуплен Гособразованием СССР. Переговоры на эту тему идут. Проблемы здесь в тиражировании как самого пакета, так и методической документации к нему.

— *Что бы вы хотели пожелать нашим читателям — пользователям КУВТ-86?*

— Уважаемые пользователи! Мы хорошо знаем ваши проблемы, знаем о непригодности поставляемого заводом-изготовителем программного обеспечения. Но «Альтернатива» существует! А следовательно, существует альтернатива тому бедственному положению с программным обеспечением, в котором находится большинство владельцев КУВТ-86. Существует альтернатива системе «Рига». Пусть практика покажет, какая из них лучше.

Пишите нам — мы вам поможем. А если ваши органы народного образования говорят о недостатке материальных средств на приобретение программного обеспечения, то это свидетельствует, как правило, лишь о недооценке роли ПС в обучении и недостатке организаторских способностей.

Вот что сказал о пакете «Альтернатива» заведующий кабинетом информатики Московского ГИУУ *Н. Д. Угринович*:

— Мы весьма положительно оцениваем этот пакет. Он неоднократно демонстрировался в Москве; имевшиеся замечания разработчики учли. Как в аспекте программирования, так и в методическом аспекте это продукт высокого качества. В частности, он содержит программные средства, необходимые для преподавания по учебнику В. А. Каймина и др.

А вот мнение ведущего инспектора-методиста отдела информатизации Московского ГКНО *Е. М. Кузнецкого*:

— Мое отношение к «Альтернативе» двойственное. С одной стороны, не все собранное в пакет программы завершены: встречаются ошибки, нереализованные функции и т. п. С другой — пакет очень хорошо использует возмож-

ности БК-0010 и КУВТ-86 и в целом оставляет благоприятное впечатление.

И, наконец, мнение учителя информатики СШ № 128 Фрунзенского р-на г. Москвы *Э. П. Комарова*:

— Главное достоинство пакета — «Виртуальный диск», позволяющий преподавателю предоставлять ДВК самому себе и работающий очень устойчиво: за два месяца не было ни одного сбоя. Он также существенно экономит время урока благодаря ускорению загрузки программ в РМУ.

Хороша система загрузки с помощью меню, весьма представительный набор программ записан на учебной дискете.

К недостаткам следует отнести превалирование игровых программ над учебными (две дискеты против одной — лучше бы наоборот) и невозможность использовать без модификации программы в кодах, написанные независимыми авторами.

59

Купите интерфейс!

В Советском Союзе ситуация с охраной авторских прав на программное обеспечение печальная, и о признании права собственности на что-то большее, чем сам текст программы, речь не идет. Запад же проблемы авторства программ решил и, похоже, «направляется» дальше.

Три крупные американские компании — «Эппл», «Эштон-Тейт» и «Лотус девелопмент» — намерены защищать в суде свои права на «внешний вид и стиль» своих программ, иначе говоря, на пользовательский интерфейс.

«Эппл» утверждает, что «Майкрософт» и «Хьюлетт-Паккард» незаконно заимствовали некоторые элементы (например, используемые в меню пиктограммы), применяемые в программах фирмы — а ведь «Эппл» гордится узнаваемостью изображений на экране «Макинтоша!» «Лотус» судится с двумя небольшими фирмами — «Мозаик» и «Пейпербэк Софтвер», выбросившими на рынок пакеты, выглядящие в точности как знаменитый «Лотус 1—2—3», но работающие быстрее и стоящие дешевле. Даже не пытайтесь анализировать тексты программ, фирма «Лотус» отказывает им в оригина-

льности и требует сатисфакции. «Эштон-Тейт» защищает пакет dBASE: «Компания «Фокс» заявляет, что переделала код. Однако самое ценное не программный код, а конструктивные особенности, которые и были заимствованы».

В такой ситуации программисты, конечно, нервничают. Дело идет к тому, что пользовательские интерфейсы, традиционно считающиеся общим достоянием, станут предметом лицензионной защиты. Э. Херцфельд, бывший сотрудник «Эппл», говорит: «Когда мы создавали программы и интерфейсы «Макинтоша», нашей единственной заботой было сделать систему настолько необычной, насколько позволяла наша фантазия. Сегодня же я дважды подумаю, прежде чем скопировать что-либо, а завтра уже не обой-

дусь в своей конторе без адвоката».

Так как же отнестись к происходящему? Более широкая защита интеллектуальной собственности — это хорошо. Но покупать лицензию на использование в своей программе, например, идеи меню?!

Лет двадцать назад в одном американском фантастическом рассказе описывалась страна, в которой патентовались словосочетания — и пользоваться запатентованными можно было только за плату.

Виват провидцам!

Что входит в периферию?

Конечно, монитор, если повезет — дисковод, если очень повезет — принтер... Еще существуют плоттеры...

А вот среди пользователей компьютеров «Макинтош», тысяча которых была опрошена журналом InfoWorld, 47 % имеют в своем распоряжении, как минимум, один сканер (устройство для автоматического ввода в ЭВМ графической информации, например изображения текста) и 26 % пользуются системами оптического распознавания текстов.



Положение о кабинете вычислительной техники всех типов средних учебных заведений

1. Общие положения

1.1. Кабинеты вычислительной техники создаются в соответствии с постановлениями ЦК КПСС и Совета Министров СССР от 12 апреля 1984 г. № 313 и от 28 марта 1985 г. № 271.

Кабинет вычислительной техники (КВТ) — это учебно-воспитательное подразделение средней общеобразовательной и профессиональной школы, межшкольного учебно-производственного комбината, оснащенное комплектом учебной вычислительной техники (КУВТ), учебно-наглядными пособиями, учебным оборудованием, мебелью, оргтехникой и приспособлениями для проведения теоретических и практических, классных, внеклассных и факультативных занятий по курсу «Основы информатики и вычислительной техники» (ОИВТ). КВТ используется в преподавании различных учебных предметов, трудового обучения, в организации общественно полезного и производительного труда учащихся, для эффективного управления учебно-воспитательным процессом.

1.2. Занятия в КВТ должны служить:

формированию у учащихся компьютерной грамотности — знаний об устройстве и функционировании современной вычислительной техники, умений и навыков решения задач с помощью ЭВМ;

ознакомлению учащихся с применениями вычислительной техники на производстве, в проектно-конструкторских организациях, научных учреждениях, учебном процессе и управлении;

— согласовано с ЦК профсоюза работников просвещения, высшей школы и научных учреждений. Письмо № 04.39 от 31.10.88 г.

Утверждено первым заместителем председателя Государственного комитета СССР по народному образованию, министром СССР Ф. И. Перегудовым 9.12.88 г.

Авторы: Шаповаленко С. Г., Смирнов Е. П., Роберт И. В., Владимиров Я. В., Петров А. Е. (НИИ ШОТСО АПН СССР), Гельтицева Е. А. (МНИИ гигиены им. Ф. Ф. Эрисмана Минздрава РСФСР).

развитию у учащихся теоретического мышления в процессе изучения основ информатики;

совершенствованию методов обучения и организации учебно-воспитательного процесса в школе.

1.3. В КВТ проводятся:

занятия по ОИВТ и отдельным общеобразовательным учебным предметам с использованием электронно-вычислительной техники, кинофильмов, диапозитивов, таблиц и других учебно-наглядных пособий;

составление учащимися прикладных программ по заданиям учителей и руководства школы для удовлетворения потребностей школы и базовых предприятий (работы учащихся по составлению программ проводятся с использованием магнитных носителей); внеклассные и факультативные занятия по ОИВТ;

экспериментальные уроки и практические занятия.

1.4. Директор школы организует работу приемной комиссии с участием представителей профсоюзной организации школы, базового предприятия, органов народного образования и государственного надзора (санитарного, пожарного, а в необходимых случаях — Госгортехнадзора) для оформления разрешения на эксплуатацию КВТ.

Без соответствующего акта разрешения эксплуатация КВТ запрещается*.

1.5. КВТ может быть школьным (обслуживать одну школу) или межшкольным (обслуживать учащихся нескольких школ с подчинением роно или руно)**. Число рабочих мест для учащихся может быть 9, 12, в зависимости от наполняемости классов. На рабочем месте предусматривается работа одного-двух учащихся.

* Приказ Министерства просвещения СССР № 241 от 08.12.86 г. «Об утверждении и введении в действие Положения об организации работы по охране труда в учреждениях системы Министерства просвещения СССР».

** Рекомендации по организации работы межшкольного КВТ в приложении 6.

Для проведения практических занятий с компьютерами классы делятся на две подгруппы (в городских школах — классы с количеством учащихся 25 и более человек, а в сельских школах — 20 и более человек).

1.6. КВТ должен быть выполнен как психологически, гигиенически и эргономически комфортная среда, организованная так, чтобы в максимальной степени содействовать успешному преподаванию, умственному развитию и идейно-политическому воспитанию учащихся, приобретению ими прочных знаний, умений и навыков по ОИВТ и основам наук, при полном обеспечении требований к охране здоровья и безопасности труда учителя и учащихся.

2. Оснащение КВТ учебным оборудованием, учебно-наглядными пособиями, оргтехникой и мебелью

2.1. Для реализации задач и содержания работ, предусматриваемых п. 1.2 и 1.3, КВТ оснащается материальными средствами согласно «Перечням технических средств, учебно-наглядных пособий и мебели для кабинетов вычислительной техники всех типов учебных заведений» (приложение 1).

2.2. Кроме того, КВТ оснащается: заданиями для осуществления индивидуального подхода при обучении, организации самостоятельных работ и упражнений учащихся на компьютерах;

комплектom научно-популярной, справочной и методической литературы;

набором лучших программ для ЭВМ, созданных учащимися;

журналом инструктажа учащихся по охране труда (приложение 5);

журналами использования КУВТ на каждом рабочем месте;

журналом отказов машин и их ремонта; держателями для демонстрации таблиц и стендами для экспонирования работ учащихся;

инвентарной книгой для учета имеющегося в кабинете учебного оборудования, годовыми и пятилетними планами дооборудования КВТ, утвержденными директором школы; аптечкой первой помощи (приложение 8); средствами пожаротушения.

2.3. Учебное оборудование и учебно-наглядные пособия для КВТ приобретаются школами в установленном порядке в соответствии с «Перечнями технических средств, учебно-наглядных пособий и мебели для кабинетов вычислительной техники всех типов учебных заведений» (приложение 1).

3. Оборудование рабочих мест учащихся и учителя

3.1. Рабочие места учащихся, оснащенные персональными ЭВМ (ПЭВМ) в соответствии с «Перечнем» (приложение 1), оборудуются двухместными столами, разработанными с учетом основных размеров ГОСТ 11015-86 «Столы ученические. Типы и функциональные размеры», и стульями. Размер крышки стола должен быть 1300×700 мм², чтобы разместить ПЭВМ, учебник и иметь место для работы с книгой (приложение 2). Столы и стулья для учащихся должны подбираться по номерам 4 и 5 в отношении 1:1. К столам подводятся электропитание и кабель локальной сети. Столы оборудуются в соответствии с требованиями безопасности и крепятся к полу.

3.2. Рабочее место учителя, оснащенное аппаратурой в соответствии с «Перечнем» (приложение 1), оборудуется столом, стулом и двумя тумбами для принтера и графопроектора. Размеры стола учителя: длина крышки не менее 1300 мм, ширина не менее 700 мм. В тумбах предусматривается 1—2 ящика размером не менее $350 \times 500 \times 100$ мм³ для принадлежностей, магнитных носителей и транспарантов из расчета на текущий день занятий. Стол используется для установки ПЭВМ и ведения записей. Он может иметь нишу для тетрадей, классного журнала и т. п. (приложение 3).

61

4. Помещение для КВТ, размещение рабочих мест, хранение учебного оборудования

4.1. Расстановка рабочих мест учащихся в КВТ (приложение 4) должна обеспечить свободный доступ учащихся и подход педагога во время урока к каждому рабочему месту учащихся.

Расстановка рабочих мест с ПЭВМ в КВТ может быть трех вариантов: периметральная, рядами (1—3-рядная), центральная. Оптимальным вариантом с точки зрения безопасности труда и создания постоянных уровней освещенности при работе является периметральная расстановка рабочих столов с ПЭВМ. При периметральной расстановке рабочих мест необходимо соблюдать следующие расстояния:

а) по ширине КВТ:

расстояние между стенкой с оконными проемами и столами должно быть не менее 80 см;

расстояние между стенкой, противоположной оконным проемам, и столами с ПЭВМ должно быть порядка 10 см, а в ряду

случаев, в зависимости от используемых видеомониторов, столы могут быть установлены непосредственно у стены;

б) по длине КВТ:

столы с ПЭВМ могут быть расставлены без разрыва и с расстоянием между ними. Последнее определяется количеством рабочих мест с ПЭВМ.

При небольшом количестве рабочих мест с ПЭВМ предпочтение следует отдавать расстановке их у стены, противоположной оконным проемам.

При наличии периметральной расстановки столов с ПЭВМ КВТ должен быть оборудован дополнительно двухместными столами (ГОСТ 11015-86. СТ СЭВ 5418-85) из расчета количества занимающихся. Эти учебные столы необходимы для теоретических занятий, опроса учащихся, выполнения контрольных работ, составления или модификации программ, решения задач в тетрадях. Расставляются эти столы в один или два ряда. Соединять двухместные столы не разрешается.

При одно-трехрядной расстановке одно-местных столов с ПЭВМ необходимо соблюдать следующие расстояния:

а) по длине КВТ:

расстояние в каждом ряду между рабочими столами должно быть 1,0—1,1 м;

расстояние в каждом ряду между последним рабочим столом и стеной не менее 0,8 м;

при расположении двери в КВТ у последних столов расстояние между ними и торцовой стеной должно быть не менее 1,2 м;

б) по ширине КВТ:

расстояние между стеной с оконными проемами и первым рядом столов с ПЭВМ должно быть от 0,8 до 1,1 м;

расстояние между рядами рабочих столов — 0,8—1,0 м;

расстояние между третьим (от оконных проемов) рядом столов и стеной — 0,8—0,9 м.

При двухрядной расстановке рабочих столов с ПЭВМ проходы между рядами соответственно увеличиваются.

При расположении столов с ПЭВМ рядами каждый стол должен иметь защитный экран со стороны тыльной части видеомонитора, предохраняющий от контакта с электропроводами. Экран крепится к столу на расстоянии 3—5 см, площадь его должна быть достаточной для защиты проводов электропитания.

Менее удачна с гигиенических позиций центральная расстановка столов с ПЭВМ, когда два ряда столов стоят без разрыва в центре класса и экраны видеомониторов обращены в противоположные стороны. При

этом видеомониторы должны располагаться в шахматном порядке.

4.2. Примерное размещение рабочих мест учащихся и учителя в КВТ приведено в приложениях 4а, б.

Вариант 4а расположения рабочих мест учащихся и учителя в КВТ (приложение 4) рассчитан на 18 рабочих мест учащихся и 9 ЭВМ. Он выполнен в традиционном стиле расположения рабочих мест учащихся и учителя.

Вариант 4б рассчитан на 24 рабочих места учащихся, оборудованных специализированными столами для ПЭВМ (ГОСТ дан выше в п. 3.1), и 18 дополнительных рабочих мест, оборудованных двухместными столами (ГОСТ 11015-86. СТ СЭВ 5418-85). Этот вариант, в частности, предполагает следующую форму работы в КВТ: те учащиеся, которые заняты подготовительной или контрольной работой (не заняты непосредственно работой на ПЭВМ), располагаются за столами (1), расположенными в центре кабинета. Это, во-первых, не «призывает» ученика, занятого письменной работой, к ПЭВМ и, во-вторых, позволяет напарнику работать на ПЭВМ в индивидуальном режиме. В случае коллективной работы все учащиеся рассаживаются парами за каждой ПЭВМ.

4.3. Общая электрическая схема питания для КВТ включается в сопроводительную документацию, поставляемую с комплектом электрооборудования кабинета вычислительной техники.

4.4. Передняя стена КВТ оборудуется классной доской, экраном, шкафом для хранения учебно-наглядных пособий и носителей информации.

Под доской устанавливаются два ящика для таблиц. На верхней кромке доски укрепляются держатели (или планка с держателями) для подвешивания таблиц.

4.5. Учебные пособия и оборудование размещаются и хранятся в кабинете по разделам программы. Демонстрационные пособия и оборудование для самостоятельных работ хранятся отдельно.

Для хранения учебно-наглядных пособий и оборудования КВТ оснащается шкафом, устанавливаемым справа от классной доски.

Демонстрационные пособия хранятся в КВТ следующим образом:

таблицы и репродукции — в ящиках под доской или в специальных отделениях шкафов по разделам программы и классам с учетом габаритов;

аудиовизуальные пособия, записи на магнитных лентах хранятся на полках шкафов, диафильмы и диапозитивы — в укладках с выемками для коробок;

диски (кассеты) с программным материалом — в специальном небольшом ящике, размером не менее $150 \times 150 \times 300$ мм³, защищенном от пыли и света, по классам и разделам программ; ящик помещается в шкаф, а места хранения в нем дисков отмечаются надписями;

справочная, методическая и научно-популярная литература — на полках шкафа (поставляется школьной библиотекой).

В КВТ создается картотека имеющегося учебного оборудования с указанием мест хранения (карточки хранятся в алфавитном порядке) и методическая картотека, облегчающая учителю подготовку оборудования к занятиям.

4.6. На стене, противоположной окнам, размещаются щиты с постоянно находящимися в кабинете справочными таблицами, знающими учащимися с правилами охраны труда, основными узлами ПЭВМ, их функциями, видами алгоритмов и т. д.

4.7. Демонстрационный телевизор устанавливается на высоте 2 м от пола на кронштейне слева от классной доски.

4.8. Пособия, необходимые для изучения отдельных тем и разделов программы, экспонируются в КВТ на задней стене.

Для экспозиции пособий, книг и материалов КВТ оснащается съемными стендами 90×90 см². Экспозиции устраиваются по наиболее важным или трудным темам курса, а также по темам, по которым учащиеся провели большую самостоятельную работу. При переходе к изучению другой темы материалы экспозиции предыдущей темы заменяются новыми.

Следует предупредить перегрузку кабинета стендами с указанными материалами. Часть материалов может быть вынесена в коридор на стены перед входом в КВТ.

4.9. В КВТ используется комплект технических средств обучения (ТСО), имеющийся в школе (приложение 1).

5. Гигиенические рекомендации к организации КВТ*

5.1. Во время работы учащихся на ПЭВМ в кабинете повышается температура и снижается относительная влажность воздуха, ухудшается ионный и качественный состав воздуха: увеличивается содержание в воз-

* Методические указания по профилактике переутомления студентов вузов при работе с видеотерминалами. М.: Изд. Мин-ва здравоохранения СССР и Главн. сан.-эпид. упр. 1988. (Утверждено начальником Главного управления по производственно-техническому обеспечению вузов 09.03.88, № 96-21-05/65 и зам. главного государственного санитарного врача СССР 05.03.88, № 4563-88).

духе органических веществ и двуокиси углерода. Содержание в воздухе указанных веществ может в несколько раз превышать предельно допустимые величины. По этой причине необходимо поддерживать следующие оптимальные параметры микроклимата:

температура воздуха 19—21 °С;

относительная влажность воздуха соответственно 62—55 %;

скорость движения воздуха менее 0,1 м/с.

Допустимыми параметрами микроклимата являются:

температура воздуха минимальная 18 °С, максимальная 22 °С;

относительная влажность соответственно 39, 31 %;

скорость движения воздуха менее 0,1 м/с.

В школах, выстроенных по проектам, в которых не предусматривались КВТ, для поддержания надлежащего микроклимата и улучшения качественного состава воздуха, в том числе и ионного, необходимо перед учебными занятиями, а если позволяют погодные условия — то во время занятий и перемен, хорошо проветривать помещение кабинета через открытые окна и фрамуги. Последние должны находиться в верхней зоне окна и открываться с помощью механической регулировки (на уровне 1,2—1,4 м от пола) под углом 30—40° в сторону учебного помещения. Проветривание КВТ должно осуществляться независимо от погодных условий, длительность должна быть не менее 10 мин. Наилучший обмен воздуха осуществляется при сквозном проветривании. Другой путь обеспечения воздухообмена может быть достигнут установлением в оконных проемах автономных кондиционеров типа БК-1500, БК-2500, БК-2000Р. Расчет необходимого количества кондиционеров должен обеспечить максимально возможное поступление наружного воздуха, но не менее 50 % от производительности кондиционера.

На воздухообмен в КВТ оказывает влияние площадь и кубатура КВТ. Минимальная площадь, приходящаяся на одну ПЭВМ, должна быть не менее 6 кв. м. Зона перед классной доской, где располагается рабочий стол учителя и демонстрационный телевизор, по длине кабинета должна быть не менее 2,5 м. Высота КВТ должна быть не менее 4 м.

5.2. КВТ должен быть оборудован умывальником с подводкой горячей и холодной воды.

При КВТ должно быть организовано подсобное помещение (18—20 кв. м), в котором располагаются радиомонтажный стол, канцелярский стол, сейф и стеллажи для хранения инструментария и оборудования.

5.3. Для исключения влияния солнечной радиации на микроклимат и создания равномерного естественного освещения КВТ следует ориентировать на северную сторону горизонта: север, северо-восток (С и С-В).

При невозможности организации КВТ с оконными проемами на северную сторону горизонта необходимо предусмотреть для окон солнцезащитные регулируемые устройства типа жалюзи, расположенные снаружи или в межстекольном пространстве.

Для повышения влажности воздуха можно использовать увлажнители или устанавливать емкости с водой типа аквариумов вблизи отопительных приборов.

5.4. КВТ не должен граничить с помещениями, имеющими повышенные уровни воздушного и ударного шума, а также располагаться вблизи этих помещений.

Звукоизоляция ограждающих конструкций КВТ должна отвечать нормативным требованиям согласно главе СНиП 11-12-77.

Для снижения уровня шума потолок или стены выше панелей (1,5—1,7 м от пола), а иногда и стены и потолок должны облицовываться звукопоглощающим материалом с максимальным коэффициентом звукопоглощения в области частот 63—8000 Гц.

Дополнительным звукопоглощением в КВТ могут быть занавеси, подвешенные в складку на расстоянии 15—20 см от ограждения, выполненные из плотной, тяжелой ткани. Ширина занавеси должна быть в два раза больше ширины оконного проема.

Для снижения шума при работе кондиционеров можно рекомендовать шумозаглушающее устройство для автономного кондиционера, перенаправляющее воздушный поток, разработанное МНИИ гигиены им. Ф. Ф. Эрисмана и ЦНИИ экспериментального проектирования учебных зданий (удостоверение на рационализаторское предложение № 280 от 09.12.88 г.).

5.5. В КВТ следует ограничивать использование полимерных материалов для отделки интерьера и оборудования класса.

Пол в КВТ должен иметь поливинилхлоридное антистатическое покрытие (ТУ 21—29—108—84). Двери и стенные шкафы могут быть облицованы поливинилхлоридным декоративным антистатическим материалом (ТУ 400—20—38—3—82).

Запрещается применять для отделки интерьера КВТ строительные материалы, содержащие органическое сырье: древесностружечные плиты (ДСП), декоративный бумажный пластик, поливинилхлоридные пленки, моющиеся обои и др.

5.6. Для снижения пыли в КВТ рекомендуется:

не входить в помещение КВТ в уличной обуви;

при входе в КВТ необходимо предусмотреть шкаф с полкой для хранения портфелей и сумок. Во вновь проектируемых классах следует предусмотреть встроенный шкаф;

классную доску, на которой пишут мелом, можно заменить доской, на которой пишут фломастером, легко снимающимся влажной тряпкой или губкой. Можно применять демонстрационные телевизоры, соединенные с ПЭВМ, установленной на столе учителя; ежедневно перед началом учебных занятий в КВТ проводить уборку помещения влажным способом и протирать экраны.

5.7. Освещенность КВТ во многом определяется цветовой и световой обстановкой.

Для уменьшения поглощения света потолок и стены выше панелей (1,5—1,7 м), если они не облицованы звукопоглощающим материалом, окрашиваются белой вододисперсионной краской (коэффициент отражения ρ должен быть не менее 0,7). Допускается окраска стен до потолка цветом панелей. Для окраски стен и панелей рекомендуется применять светлые тона красок ($\rho=0,5—0,6$). Предпочтение следует отдавать холодным тонам: светло-голубому, светло-зеленому, светло-серому. Допускается окраска стен светло-желтым, светло-бежевым цветом или цветом слоновой кости.

Нельзя окрашивать стены, расположенные напротив экранов видеомониторов, более темными тонами красок ($\rho=0,3—0,4$).

Поверхность рабочего стола должна быть цвета натуральной древесины, светло-зеленого, светло-голубого или светло-серого цвета, коэффициент отражения которых должен быть не менее 0,45.

Классная доска должна иметь темно-зеленый, темно-коричневый цвет ($\rho=0,3—0,4$).

Поверхности ограждающих конструкций КВТ, классной доски, рабочих столов должны быть матовыми.

Оконные переплеты рам, подоконники следует окрашивать белой масляной краской.

На окнах монтируются занавеси, по цвету гармонирующие с окраской стен. Занавеси не должны пропускать естественный свет и полностью закрывать оконные проемы.

Запрещается применять для окон черные занавеси!

В осветительных установках (ОУ) КВТ следует использовать систему общего освещения, выполненную потолочными или подвесными люминесцентными светильниками, равномерно размещенными по потолку рядами параллельно светопроемам, так, чтобы экран видеомонитора находился в зоне защитного угла светильника и его проекция

не приходилась на экран.

Работающие на ПЭВМ не должны видеть отражение светильников на экране.

Применять местное освещение при работе на ПЭВМ в КВТ не рекомендуется!

Выбор светильников проводится с учетом ограничения прямой и отраженной блескости. Для ограничения отраженной блескости необходимо тесно увязывать взаиморасположение светильников и экранов видеомониторов.

Люминесцентные светильники должны включаться рядами.

Минимальная освещенность рабочей поверхности стола и классной доски должна быть 400—500 лк.

Яркость экрана должна быть равной 0,5 или более яркости рабочей поверхности стола при освещенности 400—700 лк.

Для освещения КВТ рекомендуется применять светильники серий ЛПО13, ЛПО31, ЛПО33 исполнение 001 и 006, ЛСО02, ЛСО04 с металлической экранирующей решеткой и непрозрачными боковинами.

Для освещения классной доски должны применяться светильники серии ЛПО-125.

В качестве источников света рекомендуется использовать люминесцентные лампы мощностью 40 Вт или энергоэкономичные мощностью 36 Вт типа ЛБ, ЛХБ и ЛЕЦ как наиболее эффективные и приемлемые с точки зрения спектрального состава, цветовой температуры ($T_{цв}$) излучения которых находится в диапазоне 3500—4200 °К.

Не разрешается применять люминесцентные лампы типа ЛД и ЛДЦ ($T_{цв} = 6000—6500$ °К)!

Светильники должны освобождаться от пыли не менее двух раз в течение учебного года: перед началом учебного года и во время зимних каникул.

Перегоревшие лампы должны своевременно заменяться новыми.

Для обеспечения гигиенических требований к освещенности рабочих мест с ПЭВМ, учитывая особенности их расстановки и применяемых светильников, в каждом конкретном случае должен быть проведен светотехнический расчет.

Работа на ПЭВМ во время учебных занятий и при производственной практике может осуществляться при следующих видах освещения:

общем люминесцентном освещении, когда видеомониторы располагаются по периметру помещения или при центральном расположении рабочих мест в два ряда по длине класса с экранами, обращенными в противоположные стороны, а также при одно-трехрядной расстановке рабочих мест с ПЭВМ,

когда на улице темно, окна должны быть зашторены;

совмещенном освещении (естественное + искусственное) только при одно- и трехрядном расположении рабочих мест, когда экран и поверхность рабочего стола находятся перпендикулярно светонесущей стене; естественном освещении, когда рабочие места с ПЭВМ располагаются в один ряд по длине класса на расстоянии 0,8—1,0 м от стены с оконными проемами и экраны находятся перпендикулярно этой стене.

Основной поток естественного света при этом должен быть слева. Не допускается направление основного светового потока естественного света справа, сзади и спереди работающего на ПЭВМ.

Солнечные лучи и блики не должны попадать в поле зрения работающего на ПЭВМ.

5.8. При работе на ПЭВМ необходимо соблюдать правильную посадку. Работающий на ПЭВМ должен сидеть прямо, опираясь в области нижних углов лопаток на спинку стула, не сутулясь, с небольшим наклоном головы вперед (до 5—7°). Предплечья должны опираться на поверхность стола, снимая тем самым статическое напряжение мышц плечевого пояса и рук. Угол, образуемый предплечьем и плечом, а также голенью и бедром, должен быть не менее 90°.

Уровень глаз должен приходиться на центр экрана или 2/3 высоты экрана. Линия зрения должна быть перпендикулярна центру экрана, и оптимальное ее отклонение в вертикальной плоскости должно находиться в пределах $\pm 5^\circ$, допустимое $\pm 10^\circ$. Оптимальный обзор в горизонтальной плоскости от центральной оси экрана должен быть в пределах $\pm 15^\circ$, допустимый $\pm 30^\circ$. При рассмотрении информации, находящейся в крайних положениях экрана ПЭВМ, угол рассматривания, ограниченный линией зрения и поверхностью экрана, должен быть не менее 45°. Особенно важно соблюдать этот угол при работе вдвоем на ПЭВМ: чем больше угол рассматривания, тем легче воспринимать информацию с экрана видеомонитора и меньше будут уставать глаза.

Оптимальное расстояние глаз до экрана видеомонитора должно составлять 60—70 см, допустимое не менее 50 см.

Рассматривать информацию на экране видеомонитора ближе 50 см не рекомендуется.

Учащиеся, которым показана работа в очках на близком расстоянии, должны работать на ПЭВМ в очках.

5.9. Длительность работы учащихся на ПЭВМ в течение урока не должна быть более 20—25 мин («Режим занятий учащихся»

ся за видеотерминалом ЭВМ». Временные рекомендации, утвержденные Минздравом СССР 295-38/14 от 05.02.87 г.).

На мониторах, изготовленных на базе бытовых телевизоров («Агат», «Ямаха», «Корвет» и др.), необходимо через 10 мин работы проводить гимнастику для глаз (приложение 7), а на профессиональных мониторах гимнастика для глаз должна проводиться через 15 мин работы.

Целесообразно для работы учащихся на ЭВМ отводить время во второй половине урока.

6. Организация работы в КВТ

6.1. Состав учителей, преподающих курс ОИВТ и ведущих практические занятия на ПЭВМ, и персонала, обслуживающего КВТ, определяется в соответствии с типовыми штатами школьных и межшкольных КВТ.

66 6.2. Работа КВТ организуется в соответствии с настоящим Положением, Правилами внутреннего трудового распорядка базовой школы, приказами, инструкциями и указаниями вышестоящих органов народного образования, правилами, нормами и стандартами охраны труда, нормативными и инструктивными документами.

6.3. Работу КВТ возглавляет заведующий — из преподавателей ОИВТ, который является организатором оборудования кабинета, работы учителей и учащихся по ОИВТ и применению ВТ в преподавании отдельных тем программ по основам наук. Под его руководством составляются перспективный план обучения учащихся ОИВТ и план оборудования кабинета; планы на год, пятилетку и перспективу; перечни работ по самообслуживанию; расписание занятий, кружковой работы и факультативных занятий, распределяется работа между преподавателями и учащимися. Планы утверждаются директором школы.

Заведующий КВТ несет ответственность за сохранность оборудования, ведение журнала инвентаризационной записи, содержание оборудования в постоянной готовности к применению, правильное его использование, регистрацию отказов машин и организацию их отладки или ремонта, за исправность средств пожаротушения и средств первой помощи при несчастных случаях, за своевременное проведение обучения и инструктажа учащихся по охране труда, за соблюдение преподавателями и учащимися правил и норм техники безопасности, производственной санитарии, регистрацию в журнале времени начала и окончания каждого занятия, включение и выключение электропитания.

6.4. Подключение электропитания к рабочим местам учащихся и выключение его производит преподаватель и отмечает это в журнале использования КВТ на каждом занятии.

6.5. Монтаж, ввод в эксплуатацию, техническое обслуживание и ремонт комплектов учебной вычислительной техники кабинета в гарантийный и послегарантийный сроки осуществляют сервисные предприятия Государственного комитета СССР по вычислительной технике и информатике.

Работы по вводу в эксплуатацию и разовым ремонтам КУВТ выполняются на основе письма-заявки учебного заведения, гарантирующего оплату работ в соответствии с утвержденным прейскурантом, либо на основе договоров, заключенных пользователем ПЭВМ (учебным заведением или базовым предприятием) с сервисным предприятием, обслуживающим данный регион.

Техническое обслуживание КУВТ в гарантийный и послегарантийный сроки проводится только на основе договора. Отдельные операции технического обслуживания по согласованию сторон допускается проводить преподавателям, заведующим кабинетом, имеющим соответствующее удостоверение.

В течение гарантийного срока эксплуатации устранение дефекта КУВТ проводится за счет предприятия — изготовителя комплекта при условии соблюдения пользователем правил эксплуатации и хранения КУВТ.

Средние учебные заведения, базовые предприятия (заказчики, застройщики) обеспечивают строительную готовность кабинетов вычислительной техники под установку и монтаж КУВТ в соответствии с нормами и правилами, включая работы по пожарно-охранной сигнализации, проводке силовых кабелей до щита в кабинете вычислительной техники, окраске стен и потолков, выполнение электромонтажных работ с учетом правил по технике безопасности.

Сервисные предприятия обеспечивают электромонтажные работы в кабинетах вычислительной техники в соответствии с типовым проектом при условии комплексной поставки КУВТ, включая электрооборудование (силовые щиты, электрические розетки и привод).

Ремонт и техническое обслуживание бытовых телевизоров и магнитофонов, используемых в кабинете, могут осуществлять предприятия бытового обслуживания.

6.6. К оборудованию кабинетов привлекаются базовые предприятия (изготовление мебели, учебного оборудования и т. д.).

При кабинете создается актив учащихся (учеников-лаборантов), который работает по оформлению кабинета, принимает активное

участие в кружковой и факультативной работе.

6.7. Загрузка кабинета учебными, классными, внеклассными и факультативными занятиями, проводимыми учителями и специалистами, приглашенными для преподавания ОИВТ по штатному совместительству, определяется директором школы с участием заведующего кабинетом и преподавателей. Внеклассные занятия по ОИВТ проводятся в присутствии преподавателей.

6.8. В начале изучения курса ОИВТ учащиеся распределяются учителем и закрепляются по рабочим местам КВТ с учетом роста, состояния зрения и слуха.

6.9. Преподаватели строго следят за выполнением учащимися требований безопасности труда и правил работы в КВТ.

6.10. На каждом занятии в журналах использования ПЭВМ отмечаются время начала и окончания работы, содержание ее, состояние рабочего места, отказы машин.

Приложение 1

ПЕРЕЧНИ ТЕХНИЧЕСКИХ СРЕДСТВ, УЧЕБНО-НАГЛЯДНЫХ ПОСОБИЙ И МЕБЕЛИ ДЛЯ КАБИНЕТОВ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ ВСЕХ ТИПОВ УЧЕБНЫХ ЗАВЕДЕНИЙ*

Пояснительная записка

Перечни технических средств, учебно-наглядных пособий и мебели для кабинетов вычислительной техники всех типов учебных заведений составлены во исполнение постановления ЦК КПСС и Совета Министров СССР от 28 марта 1985 г. № 271 в соответствии с учебной программой курса «Основы информатики и вычислительной техники».

В перечни включены изделия, которые: а) серийно выпускаются предприятиями по заказам Министерства просвещения СССР;

б) изготавливаются промышленностью для нужд народного хозяйства, но признаны полезными и необходимыми для использования в учебных заведениях;

в) предстоит разработать, организовать их серийное производство и обеспечить ими учебные заведения (такие средства и пособия обозначены буквой «н»).

Разработка и постановка новых средств обучения будут осуществляться на протяжении 1986—1990 гг.

Поставка учебной вычислительной техники учебным заведениям будет производиться в

комплектах технических средств и оборудования кабинетов вычислительной техники (КВТ).

Количество ученических ПЭВМ, необходимых для оснащения кабинетов, определяется из расчета одна машина на двух учащихся с учетом деления классов и групп на две подгруппы (в городских школах с количеством учащихся 25 и более человек, а в сельских школах — 20 и более человек).

Независимо от количества создаваемых в кабинете ученических мест в состав КВТ включается одна машина для учителя.

В состав КВТ входит документация на магнитных и бумажных носителях. Комплект магнитных носителей на гибких магнитных дисках (НГМД) предназначен для записи программного обеспечения. Базовое программное средство (от 2 до 3 НГМД) поставляется предприятием-изготовителем. Остальные НГМД предназначаются для записи прикладных программ, в том числе копий прикладных программ, тиражируемых централизованно, а также разрабатываемых в учебных заведениях.

В состав базового программного обеспечения входят:

операционная система локальной сети; инструментальная система разработки программ;

тестовая система.

Комплект документации на бумажных носителях содержит руководство по эксплуатации базового программного обеспечения.

Включенные в перечни учебно-наглядные пособия представлены в разделах: «Пособия печатные», «Диапозитивы», «Транспаранты», «Диафильмы», «Кинофильмы».

Проекционная аппаратура, указанная в разделе «Технические средства обучения», для кабинетов вычислительной техники используется из общешкольного комплекта ТСО.

В разделе «Оргтехника» предусмотрены средства для хранения магнитных носителей, подвешивания учебных таблиц и выполнения графических работ на пленке для графопроектора и на классной доске.

В разделе «Мебель» предусмотрены комплекты специализированных ученических столов для кабинетов вычислительной техники со стульями и специализированный стол учителя с приставками для печатающего устройства и графопроектора, классная доска и шкафы для учебных пособий.

В перечнях приведены коды Общесоюзного классификатора промышленной и сельскохозяйственной продукции, присвоенные учебно-наглядным пособиям и учебному оборудованию.

* Утверждены приказом министра просвещения СССР, приказ № 161 от 30.08.85 г.

Перечни составлены НИИ школьного оборудования и технических средств обучения АПН СССР при участии управления снабжения школ УНП и УО Минпроса СССР, ВМ центра Госпрофобра СССР, НИИ высшей школы Минвуза СССР, Института проблем информатики АН СССР и согласованы с НИИ СиМО АПН СССР.

Код	Наименование	Количество
1	2	3
Средства вычислительной техники		
40 000	Комплекты технических средств и оборудования кабинетов вычислительной техники:	1 комплект
	Оборудование рабочих мест учащихся ¹	9, 12 машин
	ПЭВМ ученические: системный блок клавиатура устройство отображения информации	
	Оборудование рабочих мест учителя 1 машина	
	ПЭВМ учительские: системный блок с интерфейсами для подключения внешних устройств и рабочих мест учащихся клавиатура устройство внешней памяти устройство отображения информации	
	Устройство печати графической и символьной информации (принтер)	1 шт.
	Демонстрационный цветной видеомонитор (экран не менее 61 см по диагонали)	1 шт.
	Наборы кабелей и адаптеров локальной сети кабинета ²	1 набор
	Комплекты ЗИП	1 комплект
	Базовое программное обеспечение: операционная система локальной сети инструментальная	1 »

1	2	3
	система разработки программ	
	тестовая система	
	Магнитные носители ³	
Приборы		
96 6211 9700	Комплекты электрооборудования кабинета вычислительной техники КЭВТ — н	1 комплект
Технические средства обучения		
44 4370 0000	Кинопроекторы 16 мм ⁴	1 шт.
44 4352 0000	Диапроекторы ⁴	1 »
44 4351 4000	Графопроекторы ⁴	1 »
96 6211 4510	Экраны ⁴	1 »
96 6211 4510	Устройства для зашторивания	1 комплект
Пособия печатные		
Демонстрационные		
96 6133 9401	Перспективы развития вычислительной техники в свете решений XXVII съезда КПСС — н	1 »
96 6133 9402	Структура и принципы работы ЭВМ — н	1 »
96 6133 9403	Алгоритмы и алгоритмический язык. Алгоритмы работы с величинами — н	1 »
96 6133 9404	Построение алгоритмов для решения задач — н	1 »
96 6133 9405	Языки программирования — н	1 »
96 6133 9406	Требования безопасности при работе на ЭВМ — н	1 »
96 6133 9408	Состав и назначение базового программного обеспечения комплекта учебной вычислительной техники — н	1 »
Раздаточные		
96 6133 9101	Карточки с индивидуальными заданиями для учащихся ⁵ — н	20 »

¹ Оборудование рабочих мест учащихся приобретается с учетом наполняемости групп (одна ПЭВМ на 2 учащихся).

² Наборы кабелей и адаптеров локальной сети комплектуются с учетом количества рабочих мест учащихся.

³ В том числе 2—3 носителя с базовым программным обеспечением.

⁴ Заимствуются из общешкольных технических средств.

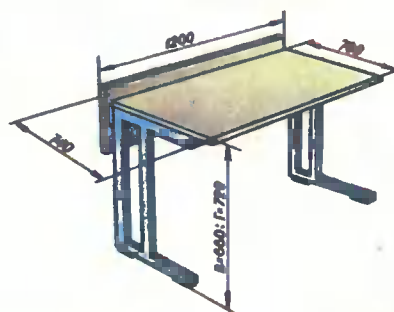
⁵ Приобретаются с учетом количества учащихся в группах.

1	2	3
	Диапозитивы	
96 6142 5501	Устройство учебной микро-ЭВМ и ее основных узлов — н	1 серия
96 6142 5502	Устройства ввода и вывода информации — н	1 »
96 6142 5503	Элементная база микро-ЭВМ — н	1 »
	Транспаранты	
96 6142 3901	Устройство и принцип действия ЭВМ — н	1 комплект
96 6142 3902	Представление информации ЭВМ — н	1 »
96 6142 3903	Представление простой команды — н	1 »
96 6142 3904	Представление команды ветвления — н	1 »
96 6142 3905	Представление команды повторения — н	1 »
96 6142 3906	Основные элементы языков программирования Бейсик, Рапира, Паскаль — н	1 »
96 6142 3907	Схема выполнения программы на ЭВМ — н	1 »
96 6142 3908	Работа на ЭВМ с графической информацией — н	1 »
96 6142 3909	Общая схема микропроцессорной системы — н	1 »
	Диафильмы	
96 6142 1001	Роль ЭВМ в современном обществе — н	1 шт.
96 6142 1002	Основные сведения об архитектуре и функционировании ЭВМ — н	1 »
	Кинофильмы⁶	
96 6143 2003	Что такое информатика ⁷	
96 6143 2001	Возникновение и развитие электронно-вычислительной техники — н	
96 6143 2002	Применение ЭВТ и микропроцессоров в народном хозяйстве — н	
	Оргтехника	
22 4452	Губки для стирания с доски	1 »

1	2	3
96 6311 3601	Держатели для таблиц «ДГ»	4 »
22 900	Держатели для мелков — н	1 »
22 900	Коробки для магнитных носителей	2 »
42 6140	Фломастеры для работы на пленке, черные и цветные	5 »
43 2124 0011	Термометр комнатный	1 »
	Бумага для устройства печати в перерасчете на лист формата А4 ⁸	12 000
	Мебель	
56 2200	Комплекты специализированные двухместные ученические для кабинетов ВТ (стол и два стула) ⁵ — н	9, 12 шт.
56 2200	Столы специализированные для учителя с приставками для графопроектора и принтера — н	1 комплект
96 6320 1000	Доска раздвижная, тип ДР ГОСТ 20064-74 классная с ящиком для таблиц	1 »
56 2200	Стулья для учителя	1 шт.
56 2200	Шкафы для учебных пособий, в том числе:	
	секция нижняя	2 »
	секция верхняя	2 »
96 6312 1103	Стенды для экспозиции учебных материалов и работ учащихся	8 »

Приложение 2

РАБОЧЕЕ МЕСТО УЧАЩЕГОСЯ

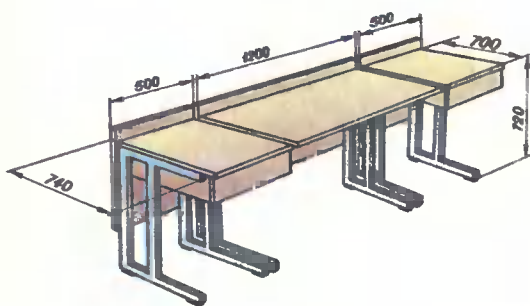


⁶ Школами не приобретаются.

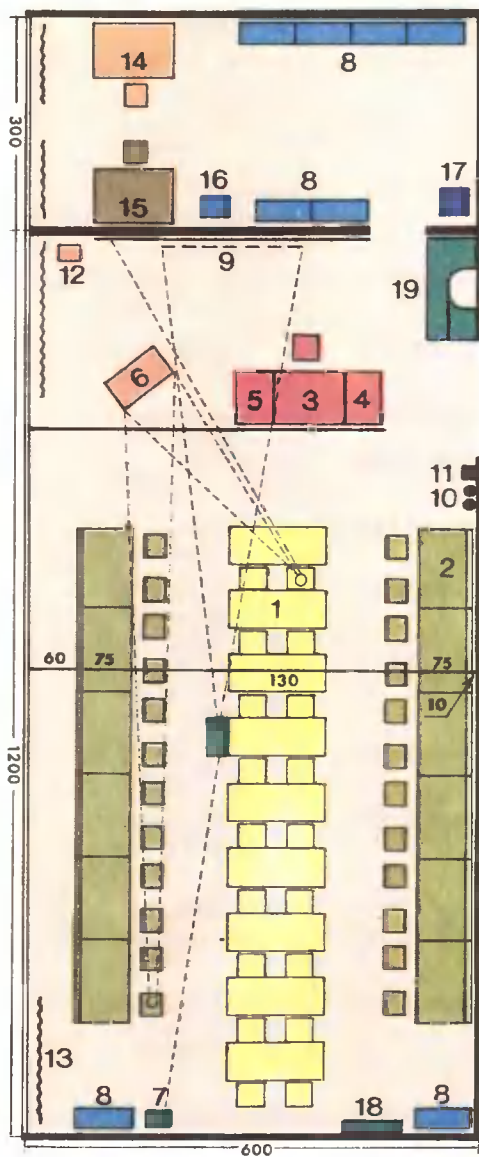
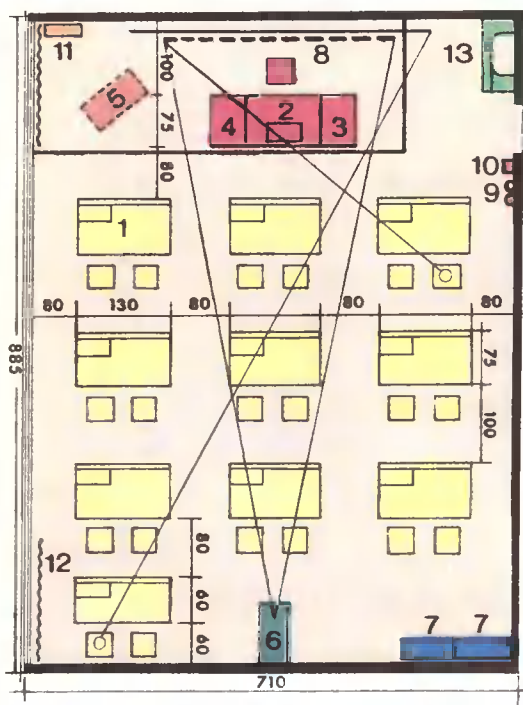
⁷ Фильм методический.

⁸ Норматив дается на один год.

РАБОЧЕЕ МЕСТО УЧИТЕЛЯ



Приложение 4а



1. Стол двухместный с ПЭВМ 130 (120)×75
2. Стол преподавателя с ПЭВМ 130 (120)×75
3. Стол-подставка под графопроектор
4. Стол-подставка под принтер
5. Телевизор
6. Подставка под проектор
7. Шкаф пристенный 90×45
8. Доска меловая
9. Огнетушители
10. Сигнализация
11. Электроштитовая
12. Шторы
13. Умывальник

1. Стол двухместный ученический 130 (120)×50
2. Стол двухместный с ПЭВМ
3. Стол преподавателя с ПЭВМ 130 (120)×75
4. Стол-подставка под графопроектор
5. Стол-подставка под принтер
6. Телевизор
7. Подставка под проектор
8. Шкаф пристенный 90×45
9. Доска меловая
10. Огнетушители
11. Сигнализация
12. Электроштитовая
13. Шторы
14. Стол рабочий 130 (120)×75

15. Стол радиомонтажный с местным отсосом
16. Тумба для инструментов
17. Сейф-шкаф
18. Настенный стенд
19. Умывальник

Приложение 5

1. ПРАВИЛА РАБОТЫ УЧАЩИХСЯ В КАБИНЕТЕ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ*

1.1. В кабинете вычислительной техники (КВТ) установлена дорогостоящая, сложная и требующая осторожного и аккуратного обращения аппаратура — компьютеры (ПЭВМ), принтер, другие технические средства.

Поэтому:

бережно обращайтесь с этой техникой; спокойно, не торопясь, не толкаясь, не задевая столы, входите в кабинет и занимайте отведенное вам место, ничего не трогая на столах.

1.2. На вашем рабочем месте размещены составные части ПЭВМ — системный блок, клавиатура и монитор (дисплей).

Во время работы лучевая трубка монитора (дисплея) работает под высоким напряжением.

Неправильное обращение с аппаратурой, кабелями и мониторами может привести к тяжелым поражениям электрическим током, вызвать загорание аппаратуры.

Поэтому строго запрещается:

трогать разъемы соединительных кабелей; прикасаться к питающим проводам и устройствам заземления;

прикасаться к экрану и к тыльной стороне монитора, клавиатуры;

включать и отключать аппаратуру без указания преподавателя;

класть диск, книги, тетради на монитор и клавиатуру;

работать во влажной одежде и влажными руками.

1.3. При появлении запаха гари немедленно прекратите работу, выключите аппаратуру и сообщите об этом преподавателю.

1.4. Перед началом работы:

убедитесь в отсутствии видимых повреждений рабочего места;

сядьте так, чтобы линия зрения приходилась в центр экрана, чтобы не наклоняясь пользоваться клавиатурой и воспринимать передаваемую на экран монитора информацию;

разместите на столе тетрадь, учебное посо-

бие, журнал использования ПЭВМ так, чтобы они не мешали работе на ПЭВМ; внимательно слушайте объяснение учителя и старайтесь понять цель и последовательность действий; в случае необходимости обращайтесь к преподавателю;

хорошо разберитесь в особенностях применяемых в работе аппаратов;

запишите в журнал регистрации время начала работы на ПЭВМ;

расчехлите машину;

начинайте работу только по указанию преподавателя «Приступить к работе».

1.5. Во время работы ПЭВМ лучевая трубка видеомонитора является источником электромагнитного излучения, которое при работе вблизи экрана неблагоприятно действует на зрение, вызывает усталость и снижение работоспособности.

Поэтому надо работать на расстоянии 60—70 см, допустимо не менее 50 см, соблюдая правильную посадку, не сутулясь, не наклоняясь; учась, имеющим очки для постоянного ношения, — в очках.

1.6. Работа на ПЭВМ требует большого внимания, четких действий и самоконтроля.

Поэтому нельзя работать:

при недостаточном освещении;

при плохом самочувствии**.

1.7. Во время работы:

строго выполняйте все указанные выше правила, а также текущие указания учителя;

следите за исправностью аппаратуры и немедленно прекращайте работу при появлении необычного звука или самопроизвольного отключения аппаратуры. Немедленно докладывайте об этом преподавателю;

плавно нажимайте на клавиши, не допуская резких ударов;

не пользуйтесь клавиатурой, если не подключено напряжение;

работайте на клавиатуре чистыми руками; никогда не пытайтесь самостоятельно устранить неисправность в работе аппаратуры;

не вставайте со своих мест, когда в кабинет входят посетители.

1.8. По окончании работы:

отключайте тумблер «Сеть»;

запишите в журнале регистрации использования ПЭВМ время окончания работы;

протрите аппаратуру куском мягкой чистой ткани;

зачехлите аппаратуру.

1.9. Вы должны хорошо знать и грамотно выполнять эти правила, точно следовать указаниям преподавателя, чтобы:

* Этот раздел рекомендуется размножить (10—15 экз.), чтобы передавать на рабочие места учащихся при изучении правил.

** В этом случае надо обратиться к врачу.

избежать несчастных случаев; успешно овладеть знаниями, умениями, навыками;

сберечь государственное имущество — вычислительную технику и оборудование.

Вы отвечаете за состояние рабочего места и сохранность размещенного на нем оборудования.

Невыполнение правил — грубейшее нарушение порядка и дисциплины.

2. УКАЗАНИЯ ПО ПРОВЕДЕНИЮ ОБУЧЕНИЯ И ИНСТРУКТИРОВАНИЯ УЧАЩИХСЯ ПО БЕЗОПАСНОСТИ ТРУДА.

2.1. Для воспитания у учащихся сознательного отношения и усвоения правильных и безопасных методов и приемов работы учителя обязаны проводить инструктирование и обучение учащихся по соблюдению требований техники безопасности и гигиены труда.

2.2. Инструктаж и обучение по охране труда проводятся со всеми учащимися на вводном занятии в кабинете, а затем перед практической работой на ПЭВМ.

2.3. На вводном инструктаже учитель должен ознакомить учащихся с правилами распорядка в кабинете, правилами техники безопасности и гигиены труда; с опасными моментами, с которыми можно столкнуться в процессе работы, и с соответствующими мерами предосторожности.

2.4. Вводный инструктаж проводится заведующим кабинетом (учителем) в виде лекции, беседы по программе, разработанной с учетом требований стандартов ССБТ, правил по технике безопасности для КВТ, действующих в системе ГКНО СССР, МЗ СССР, а также всех особенностей работы данного кабинета, утвержденной руководителем и профсоюзным комитетом учрежде-

ния просвещения, структурным подразделением которого является КВТ.

2.5. Инструктаж перед работой на ПЭВМ (первичный на рабочем месте) дополняет вводный инструктаж и имеет целью ознакомить учащихся с требованиями правильной организации и содержания рабочего места, назначением приспособлений и ограждений, с безопасными методами работы и правилами пользования защитными средствами, с возможными опасными моментами при выполнении конкретной работы, с обязанностями работающего на своем рабочем месте, а также опасными ситуациями и правилами поведения при их возникновении. Первичный инструктаж на рабочем месте проводят в соответствии с инструкциями по охране труда, разработанными и утвержденными согласно требованиям стандартов ССБТ администрацией совместно с профсоюзным комитетом данного учреждения просвещения.

2.6. Первичный инструктаж на рабочем месте должен содержать четкие и конкретные указания по безопасности труда и в необходимых случаях сопровождаться показом правильных и безопасных приемов выполнения работы.

2.7. По окончании первичного инструктажа на рабочем месте учитель разрешает приступить к самостоятельной работе только после того, как убедится, что все учащиеся усвоили инструктаж.

2.8. В процессе выполнения работы учитель и лаборант обязаны систематически контролировать выполнение каждым учеником данных ему при инструктаже указаний о безопасном способе выполнения работы.

2.9. Все сведения по проведению инструктажа учащихся регистрируются в классном журнале, а при проведении внеклассных занятий — в специальном журнале.

ЖУРНАЛ

инструктажа учащихся по охране труда при организации занятий в КВТ и проведении внеклассных и внешкольных мероприятий

№ п/п	Фамилия, имя инструктируемого	Дата инструктажа	Класс	Содержание инструктажа с указанием названия инструкции	Фамилия, имя, отчество, должность проводившего инструктаж	Подпись проводившего инструктаж	Подпись в получении инструктажа
1	2	3	4	5	6	7	8

Примечание. Учащиеся расписываются в журнале по технике безопасности начиная с VIII класса.

**РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ РАБОТЫ
МЕЖШКОЛЬНОГО КАБИНЕТА
ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ
(на базе персональных ЭВМ)**

1. Общие положения

1.1. Межшкольный кабинет вычислительной техники (МКВТ) организуется на базе средней общеобразовательной школы или УПК в соответствии с постановлением ЦК КПСС и Совета Министров СССР от 28 марта 1985 г. № 271.

1.2. Межшкольный кабинет вычислительной техники — это учебно-воспитательное подразделение средней общеобразовательной школы или межшкольного учебно-производственного комбината, оснащенное комплектом учебной вычислительной техники (КУВТ), учебно-наглядными пособиями, учебным оборудованием, мебелью, оргтехникой и приспособлениями для проведения с учащимися базовой и прикрепленных школ теоретических и практических, классных, внеклассных и факультативных занятий по курсу «Основы информатики и вычислительной техники» (ОИВТ) и другим общеобразовательным дисциплинам с использованием ПЭВМ. Базовой считается школа, имеющая стандартный КУВТ и способная обеспечивать учебно-воспитательный процесс во всех прикрепленных классах. Межшкольный кабинет ВТ является учебным кабинетом базовой школы и показывается в статотчетности данной школы с пометкой «межшкольный». При этом количество учащихся X—XI классов, прикрепленных к МКВТ школ, учитывается с коэффициентом 0,25 при установлении должностного оклада директору базовой школы. Статус МКВТ присваивается приказом по району.

2. Организация и содержание работы МКВТ.

2.1. Работа кабинета организуется администрацией базовой школы в соответствии с настоящими рекомендациями, Уставом средней общеобразовательной школы, внутренним распорядком работы базовой школы, нормативными и инструктивно-методическими документами по ОИВТ.

Учебная нагрузка комплекта учебной вычислительной техники в МКВТ должна быть не менее 36 ч в неделю. Исходя из этого, определяется общее число X—XI классов, проходящих обучение в МКВТ (учащиеся одиннадцатых классов во время практических занятий делятся на 2 группы).

Эффективное использование МКВТ в малокомплектных школах должно осуществ-

ляться за счет привлечения к внеклассной и кружковой работе учащихся младших и средних классов.

В случае, если базовая школа не в состоянии выделить помещение для МКВТ, по рекомендации роно (руно) он может быть размещен в другом, специально оборудованном для этого помещении (например, в помещении базового предприятия или ближайшего УПК).

Расписание занятий, проводимых в МКВТ, составляется администрацией базовой школы по согласованию с администрацией прикрепленных школ (УПК).

Кружки по различной тематике, связанной с использованием ЭВМ, проводятся преподавательским составом школы или привлеченными для этих целей сотрудниками ВЦ или других организаций и контролируются учителем, на которого возложено заведование МКВТ.

Расписание факультативных занятий и кружковой работы составляется учителем, на которого возложено заведование МКВТ, по согласованию с администрацией базовой и прикрепленных школ (УПК).

2.2. Преподавательский состав МКВТ

Заведование кабинетом возлагается директором базовой школы на одного из учителей ОИВТ, которому в установленном порядке производится дополнительная оплата. Этот учитель является организатором работ по оборудованию кабинета, работы учителей и учащихся в МКВТ. Под его руководством составляются в соответствии с действующей программой курса план обучения, перспективные планы развития МКВТ на год, пятилетку; состав работ по самооборудованию, распределяется работа между учителями и учащимися. Все планируемые работы по МКВТ утверждаются директором базовой школы. Учитель, на которого возложено заведование МКВТ, может быть освобожден от классного руководства.

Учитель, на которого возложено заведование МКВТ, несет ответственность за поддержание оборудования в постоянной готовности к применению, своевременность и тщательность профилактического технического обслуживания вычислительной техники, правильное использование ее, регистрацию отказов машин и организацию их отладки или ремонта, за исправность средств пожаротушения и первой медицинской помощи, за своевременное проведение обучения и инструктажа учителей и учащихся по охране труда, за соблюдение учителями и учащимися требований безопасности труда, включение и выключение электропитания.

В состав учителей, преподающих курс ОИВТ и ведущих практические занятия на

ПЭВМ, должны входить:

учитель, на которого возложено заведование МКВТ;

учитель из учителей школ (УПК) (рекомендуется привлечение к работе в качестве лаборантов МКВТ наиболее активных учащихся базовой школы. В этом случае их работа должна быть учтена как общественно полезный труд).

Расписание работы учителей в МКВТ составляется заместителем директора по учебной работе базовой школы.

2.3. Контроль преподавания ОИВТ осуществляется директором и завучем базовой школы в соответствии с должностными инструкциями.

Работа учителя, на которого возложено заведование МКВТ, регулируется планом работы МКВТ и расписанием занятий.

2.4. При МКВТ организуется учебно-методическое объединение учителей, преподающих курс ОИВТ. Заседания методического объединения проводятся регулярно, не реже чем один раз в квартал, со всей сопровождающей эти заседания документацией. Учебно-методическое объединение осуществляет знакомство учителей информатики с новыми методическими материалами и программными средствами для курса ОИВТ, проводит обучение учителей-предметников основам работы с ЭВТ, организует обмен опытом преподавания ОИВТ, факультативной, внеклассной и кружковой работы МКВТ (работа объединения курируется методистами методического кабинета роно (руно)).

Приложение 7

УПРАЖНЕНИЯ ДЛЯ ГЛАЗ

Для снятия напряжения глаз можно рекомендовать ряд упражнений, которые могут индивидуально проводить работающие на ЭВМ в зависимости от уставаемости глаз.

Вариант 1.

Закрывать глаза на несколько секунд, сильно напрягая глазные мышцы, затем раскрыть их, расслабив мышцы глаз. Дыхание ритмичное. Повторить 4—5 раз.

Посмотреть на переносицу и задержать взор в течение трех полных ритмичных дыханий. До усталости глаза доводить нельзя. Затем посмотреть вдаль. Повторить 4—5 раз.

Не поворачивая головы, посмотреть направо и зафиксировать взгляд на несколько секунд, затем посмотреть вдаль прямо. Аналогичным образом проводятся упражнения, но с фиксацией взгляда влево, вверх, вниз. Повторить 4—5 раз.

Часто открывать и закрывать глаза (моргать) в течение 20—30 с.

Вариант 2.

Очень медленно вращать глазами. Выдыхая, смотреть вправо, затем вверх, выдыхая — влево и вниз. Затем устремить взор вдаль. Повторить комплекс упражнений, направляя взгляд в противоположную сторону. Прodelать упражнения 4—5 раз в одну и 4—5 раз в другую сторону.

Для улучшения фокусировки глаза посмотреть на кончик носа, потом сразу перевести взор вдаль. И в том и в другом случае взгляд задерживается на несколько секунд. Повторить эти упражнения 4—5 раз.

Закрывать глаза на 3—5 с и затем открыть глаза на такое же количество времени. Упражнение повторить 4—5 раз.

При неподвижной голове перевести взор вверх — прямо — вниз, затем влево — прямо — вправо, аналогичным образом проделывать движение глаз по диагонали в одну и в другую сторону. Повторить весь комплекс упражнений 4—5 раз.

Вариант 3.

Посмотреть вправо, не поворачивая головы, и зафиксировать взор на несколько секунд, затем влево на несколько секунд и посмотреть прямо. Поднять глаза вверх на несколько секунд и опустить глаза вниз, расслабив мышцы глаз, посмотреть прямо. Повторить весь комплекс упражнений 4—5 раз.

Посмотреть несколько секунд на кончик пальца, удаленного на 30 см от глаз, затем перевести взгляд вдаль и задержать взор на 3—5 с. Упражнение повторить 4—6 раз.

Закрывать глаза на 5—10 с, расслабив глазные мышцы, и открыть их на 5—10 с. Повторить 4—5 раз.

Прodelать несколько круговых движений глазами (4—5 движений) в одну сторону и такое же количество движений — в другую сторону. Темп средний. Перевести взгляд прямо на 5—7 с. Повторить 3—4 раза.

Поморгать несколько раз глазами, не напрягая глазные мышцы.

Упражнения для глаз осуществлять в период выполнения отдельных упражнений физкультурных пауз и физкультурных минуток.

Приложение 8

ОКАЗАНИЕ ПЕРВОЙ МЕДИЦИНСКОЙ ПОМОЩИ ПОСТРАДАВШИМ ОТ ЭЛЕКТРОТОКА

Поражение электротоком может варьировать от незначительных болевых ощущений при отсутствии органических и функциональных изменений до ожогов III—IV степени (обугливание) и шока.

При поражении электротоком следует быстро отключить ток, освободить пострадавшего от проводов с соблюдением мер предосторожности по нераспространению тока на лиц, участвующих в оказании помощи (освободить пострадавшего в резиновых перчатках или обернуть руки сухой тканью, встать на сухую доску или коврик и т. п.).

При поражении электротоком следует обратить внимание на дыхание и сердечно-сосудистую систему.

В случае остановки дыхания и сердечной деятельности немедленно начать непрямой массаж сердца (надавливание ладонями быстрыми толчками в области середины грудной кости, ближе слева, в ритме 40—60 толчков в минуту) и искусственное дыхание рот в рот (предварительно закрыть нос пострадавшему и, вдохнув полной грудью, выдохнуть через марлю или носовой платок в рот пострадавшему) или рот в нос (предварительно закрыть рот пострадавшему и через марлю или носовой платок аналогич-

ным образом осуществлять выдох в нос). Искусственное дыхание проводят с частотой 16—18 раз в минуту. Одновременно вызывают специализированную бригаду скорой помощи.

Непрямой массаж сердца и искусственное дыхание проводят до полного восстановления или до оказания специализированной помощи.

При ослаблении сердечной деятельности и дыхания необходимо обеспечить подачу свежего воздуха, дать понюхать на ватке нашатырный спирт, влить в рот чайную ложку или 25 капель кордиамина, провести похлопывание по щекам.

При повреждении кожи наложить стерильную повязку и направить пострадавшего к врачу. Поврежденную поверхность кожи вокруг ожога следует обработать зеленкой или розовым раствором марганцовки.

В КВТ должна быть аптечка первой доврачебной помощи, состав которой представлен ниже.

75

Содержание аптечки первой помощи

Наименование медицинских средств и медикаментов	Назначение	Кол-во
Индивидуальные перевязочные антисептические пакеты	Для наложения повязок	3 шт.
Бинты	То же	3 »
Вата в пакетах	»	3 »
Жгут для остановки кровотечения	Для остановки кровотечения	1 »
Настойка йода	Для смазывания окружности раны, свежих ссадин, царапин на коже и т. д.	1 склянка
Нашатырный спирт	Применять при обмороке, накапав на ватку 2—3 капли и поднося к носу пострадавшего	1 флакон
Раствор 2—4 % борной кислоты	Для промывания глаз, для примочек на глаза, при ожогах	1 »
1 % спиртовой раствор бриллиантовой зелени	Для смазывания ран, царапин, ссадин на коже и т. п.	1 »
Кордиамин	Для стимулирования центральной нервной системы, сердечной и дыхательной деятельности; 25 капель (1 чайная ложка на прием)	1 »
Резиновые перчатки		1 пара
Марганцевокислый калий (марганцовка)	Для смазывания повреждений поверхности кожи	1 упаковка
Липкий пластырь		1 »

Г. ЖИТКОВ, М. ПРОХОРОВА
г. Казань

Комплексное применение средств ВТ при обучении по курсу ТОР

Одним из фундаментальных курсов, изучаемых студентами радиофизических специальностей, является курс «Теоретические основы радиотехники» (ТОР). В течение ряда лет в Казанском университете в различных формах занятий по этому курсу успешно используются разнообразные средства вычислительной техники (ЭВМ ДЗ-28, БК-0010, ДВК, ЕС-1033; АВМ МЕ-10М, АВК-31).

Форма занятий по курсу ТОР	Применяемые средства ВТ		
	микро-ЭВМ	АВМ	ЭВМ
Лабораторный практикум	Р, М	М	
Практические занятия по решению задач	Р		Р, И
Курсовые и дипломные работы	Р	М	Р
Контрольные работы и коллоквиумы	Р		Р, И

Р — расчеты, М — моделирование, И — информация.

Целесообразность применения той или иной вычислительной машины диктуется как формой занятий и особенностями изучаемого раздела курса, так и дидактическими целями, ставящимися преподавателем. Так, для проведения аналитических расчетов при выполнении лабораторных работ и при решении задач по основным разделам курса используются микро-ЭВМ. Программы расчетов на Бейсике и Фокале составляются обучаемыми самостоятельно. Темы «Последовательный колебательный контур» и «Параллельный колебательный контур» в практической части по решению задач реали-

зованы в виде обучающих курсов в АОС КГУ, базирующейся на ЭВМ ЕС-1033. Проверка теоретических результатов осуществляется на моделях, где в качестве основных технических средств использованы аналоговые машины МН-10М, АВК-31 и групповые модели. Применение различных вычислительных машин, таким образом, преследует две цели в обучении: изучение разделов курса ТОР и привитие общего навыка работы с вычислительными машинами.

На применении АОС и аналоговых моделирующих средств остановимся подробнее. Курс по решению задач по ТОР в АОС КГУ рассчитан на применение во время практических занятий. Он имеет четыре раздела, состоящих из информационной, обучающей и контролирующей частей. Все разделы относятся к темам «Последовательный колебательный контур» и «Параллельный колебательный контур».

Задачи в курсе разбиты на несколько уровней с учетом сложности их логических структур. В зависимости от уровня каждой задаче приписан вес (от 5 до 9 баллов), в зависимости от которого определяется порядок предъявления задач обучаемому для решения. В результате студенты постоянно ощущают рост сложности выполняемых заданий, что повышает уровень мотивации. Имеется возможность учесть, насколько самостоятельно решена та или иная задача: при обращении обучаемого за помощью к машине (а это предусмотрено на нескольких этапах решения) из веса предложенной задачи вычитаются баллы. Оценка, выставленная за решенную задачу, влияет на то, дается ли очередная задача более сложной или повторяется предыдущая задача в некоторой вариации. Кроме этого,

в каждом разделе имеется обучающая задача (дающаяся с подробным объяснением). Для подсчетов по формулам имеется программа «Вычислитель», позволяющая обращаться к ЕС-1033. Продвижение обучающего по курсу зависит от того, насколько успешно он справился с предложенными задачами.

При проведении практических занятий особенно важно осуществлять дидактические принципы сознательности и самостоятельности студентов. Для этого в курсе предусмотрен случайный набор числовых данных в задачах с общим для всех условием, что увеличивает самостоятельность выполнения заданий. Студенты, продвигающиеся по курсу быстро и результативно, по ходу работы поощряются, а те, кто набрал более 50 баллов, освобождаются от контрольной задачи и получают зачет. Курс является одним из наиболее сложных в АОС КГУ, однако его сравнительно широкие логические возможности позволяют повысить эффективность обучения за счет индивидуализации и самостоятельности при прохождении указанных тем. Эксперименты и работа с курсом показали, что в результате обучения менее подготовленные студенты быстрее подтягиваются к более высокому уровню, прививаются навыки расчетов, более уверенным становится переход к следующим темам курса ТОР. Кроме того, преподаватель освобождается от проверки нескольких сотен задач в семестре.

Известно, что АВМ наиболее приспособлена для решения обыкновенных дифференциальных уравнений (ОДУ), описывающих поведение динамической системы. Возможности АВМ МН-10М и АВК-31 сравнительно невелики, однако их вполне достаточно для проведения экспериментов по тем разделам курса ТОР, которые связаны с изучением теории радиотехнических цепей, в особенности переходных процессов в цепях. Модели строятся двух типов: структурные и прямой аналогии. В первом случае структура моделирующей схемы полностью повторяет структуры системы ОДУ, чем достигаются наглядность и удобство проведения экспериментов. Возмущающие воздействия, как правило, также моделируются. В процессе подготовки своей задачи к моделированию на АВМ студент приводит уравнение к машинному виду, выбирает масштабы по переменным и времени, составляет структурную схему и т. д. Хотя учебные задачи требуют решения систем ОДУ невысоких порядков (3—6), правильный выбор масштабов и коэффициентов передач операционных блоков связан с определенными трудностями. Для упрощения этих этапов студентам

предлагается предварительно решить задачу на ЦВМ ДЗ-28, ДВК или БК-0010. На основе этого решения подбираются необходимые данные для машинного уравнения на АВМ. Точностные вопросы моделирования жестко не ставятся, тем не менее структурное моделирование дает хорошие не только качественные, но и количественные результаты.

Широкие возможности дает использование моделей прямой аналогии. Их идея состоит в замене пассивных элементов радиосхем схемами активных четырехполосников. Путем изоморфных преобразований удалось получить более удобные схемы четырехполосников. Эти модели также требуют подготовительной работы, связанной с выбором значений масштабов, коэффициентов передачи блоков и величин сопротивлений и конденсаторов в модели. Возможность эквивалентной замены элементов дуальными схемами исключила необходимость использования в моделях индуктивностей. Еще одна особенность этого способа моделирования: для изучения работы цепи не требуется знания уравнений, модель получается простой заменой элементов цепи оригинала, а не заменой членов уравнения, как в структурной модели. Однако после приведения схемы к окончательному виду она совпадает со схемой, которую можно было бы получить по уравнению (рис. 1). Такое единство форм моделей позволяет обучаемым более углубленно изучать не только исходные вопросы из курса ТОР, но и вопросы, связанные с возможностями машинного моделирования на АВМ. Адекватность моделей прямой аналогии весьма высока. Для моделирования на АВМ составлен банк типовых задач (около 60 вариантов) с возможностью изменения числовых значений элементов схем.

Задачи на моделирование радиоцепей на ЭВМ и АВМ предлагались учащимся выпускных классов подшефной школы в качестве выпускной работы во время практики в университете. Эти учащиеся ориентированы на поступление на физфак КГУ, поэтому знакомство школьников с современными методами моделирования обеспечивает преемственность их обучения. Опыт показывает, что

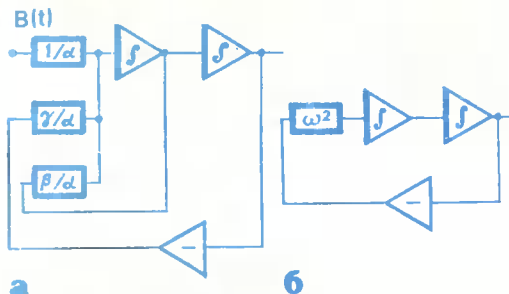


знаний учащихся достаточно для усвоения основных процедур структурного моделирования. Очевидна простота набора задачи на АВМ, где не приходится иметь дело с программированием и не требуется умение интегрировать дифференциальные уравнения. Для качественной интерпретации результатов вполне достаточно получения решения способом II.

Примеры. Задача 1. Известно, что дифференциальное уравнение второго порядка $\alpha x'' + \beta x' + \gamma x = B(t)$ описывает колебательный процесс, причем разные значения коэффициентов α, β, γ приводят к возникновению колебаний различного вида: апериодических, гармонических, затухающих, возрастающих. Составить структурную модель и исследовать поведение системы.

Задача 2. Уравнение $x'' + \omega^2 x = 0$ описывает гармонический осциллятор. Составить схему и настроить осциллятор на заданную частоту.

Одним из способов (I, II) составляются структурные схемы (рис. 2,а для задачи 1; рис. 2,б для задачи 2) и проводятся исследования на модели. Затем по готовой программе эти же уравнения решаются на



ЦВМ (способ III) для сопоставления и оценки возможностей каждой вычислительной машины.

В настоящее время проводятся работы по обновлению и усложнению задач, а также по созданию учебной аналого-цифровой вычислительной системы, которую предполагается использовать для моделирования в реальном времени задач, относящихся к разделу «обратные связи». Рассматриваются вопросы количественной оценки эффективности применения ВТ в различных формах занятий по курсу ТОР.

Адрес для справок: 420008, Казань, ул. Ленина, 18, КГУ, физфак, Житкову Г. Н.

Ю. МИХЕЕВ

учитель, г. Новосибирск

Программное сопровождение курса планиметрии

Внедрение ВТ в школу ставит перед системой народного образования задачу по созданию программного обеспечения, позволяющего обогатить процесс обучения новыми формами, связанными с использованием ЭВМ на уроках. Ясно, что характер программного обеспечения для различных предметов должен быть разным. Если для гуманитарных в первую очередь нужны программы, содержащие большие объемы информации, то для естественнонаучных — программы, способствующие развитию логических и аналитических способностей. Именно такая цель и была поставлена в Новосибирском государственном университете и физико-математической школе при НГУ в связи с компьютеризацией школьного курса математики.

В настоящее время в НГУ создана система программ, сопровождающих курс планиметрии. Завершены шесть программ для VII класса, восемь программ для VIII и шесть программ для IX. В разработанных

программах ученик достаточно свободен в выборе каждого шага, разница лишь в степени свободы этого выбора, которая определяется тематикой. Созданные программы можно разделить по степени свободы на четыре типа.

Первый тип программ близок к распространенным программам, в которых нужно отвечать на вопросы по принципу «да—нет». Он состоит из одной программы «Геометрия на клетчатой бумаге» для проведения первого компьютерного урока в VII классе. Программа посвящена абстрактным понятиям взаимного расположения точек и прямых, причем моделью точек служат узлы клетчатой бумаги, а прямые задаются двумя точками из числа узлов. Программа содержит пять видов задач: в одних следует определить, лежит ли данная точка на прямой или нет, во вторых — указать точку на прямой, параллельной данной, в третьих — выяснить, пересекает ли прямая отрезок и если пересекает, то поставить точку пересе-

чения; в задачах четвертого вида — какой из лучей лежит между двумя другими; в задачах пятого — пересекает ли прямая треугольник. Все задачи подобраны таким образом, чтобы с первого взгляда было достаточно трудно правильно ответить на поставленный вопрос. Элементы пассивной работы, сводящейся к выбору одного из двух ответов «да—нет», сочетаются с активными действиями (учащийся сам должен поставить некоторую точку). Обратим внимание еще на один момент. Как известно, при решении геометрических задач важную роль играет интуитивное мышление. В начале VII класса изучаются абстрактные понятия, в то время как свойства геометрических фигур еще неизвестны. Поэтому использование свойств клетчатой бумаги, интуитивно ясных, представляется важным, так как в этом случае работа с программой не сводится к угадыванию, а подкрепляется реальными известными свойствами.

Второй тип программ связан с перечислением объектов, удовлетворяющих определенным свойствам. Это «Вписанные углы» и «Равные треугольники» для VII класса. Первая из них содержит четыре задачи, в которых, используя свойства вписанных в окружность углов, следует перечислить углы, равные указанному. Задачи подобраны так, чтобы равных углов было достаточно много, следовательно, их решение требует и внимания, и аккуратности. Для некоторых углов сразу трудно найти причины, по которым они равны данному, и в таких случаях учащиеся имеют возможность проводить эксперименты, пытаясь называть «подозрительные» углы. Заинтересованный учитель может после работы с этой программой уделить немного времени включенным в нее задачам и разобрать их уже с полными доказательствами. На похожих принципах основана и программа «Равные треугольники». В ней предлагается два вида задач: в одном из них следует указать треугольники, равные указанному, а во втором — указать пары равных треугольников.

Третий тип программ содержит наборы задач (в каждой программе по определенной теме), для решения которых необходимо отыскать некоторую последовательность элементарных действий. На каждом очередном этапе предлагается список, содержащий практически все действия, которые возможно пытаться применять в возникшей ситуации. Ученик свободен в выборе любого из них и при каждой попытке получает определенное сообщение компьютера, анализирующее только предпринятый шаг, но ничего не говорящее о том, приближает ли данное действие учащегося к ответу или нет. Как

правило, процесс решения включенных в программы задач достаточно длинный (особенно в задачах для IX класса). Но он не является утомительным, потому что компьютер берет на себя всю трудоемкую часть работы. Это позволяет учащимся сосредоточиться на самом главном: нахождении пути решения задачи. Таким образом, программы данного типа учитывают индивидуальные особенности учащихся и позволяют реализовать различные способы решения задач. В программах достаточно просто организовать контроль. Когда задача решена, в программах появляется крупная заставка со словами: «Молодец! Задача решена правильно».

По такому принципу сделаны программы «Углы» для VII класса, «Четырехугольники», «Теорема Пифагора» и «Отношение отрезков» для VIII класса, «Векторы», «Задачи на применение теорем синусов и косинусов», «Задачи на вычисление площадей треугольников», «Площади многоугольников» и «Вписанные четырехугольники» для IX класса.

Программа «Вписанные многоугольники» дает возможность применять теорему синусов (а тем самым и тригонометрические функции углов) в решении задач, связанных с окружностями. Программа «Площади многоугольников» посвящена решению задач, в которых площадь одной фигуры считается известной, а площадь другой, связанной с ней геометрически, требуется найти. Ясно, что в такой программе на первый план выходит вычисление суммы, разности площадей, части площадей многоугольников. В программе «Теорема Пифагора» содержится пять задач, при решении которых наряду с теоремой Пифагора приходится использовать тригонометрические функции углов. Следовательно, уже в начале изучения тригонометрических функций обращается внимание на важность их применения при решении геометрических задач на вычисления.

Четвертый тип программ состоит пока из одной программы «Задачи на геометрическое место точек» для IX класса. В ней приводится восемь задач на нахождение множества точек плоскости, удовлетворяющих определенным геометрическим свойствам. В отличие от остальных эта программа не предоставляет средства для окончательного решения задач, а только помогает на первом этапе: с помощью программы ученик может выработать гипотезу относительно вида множества, удовлетворяющего данному свойству. Может показаться, что при работе с подобной программой учащимся приходится играть пассивную роль (за исключением выбора начальных данных). Однако это не совсем

так, потому что программа моделирует важный момент решения подобных задач, а именно в не совсем понятной ситуации учащимся приходится экспериментировать, выделять частные случаи и т. д. После работы с программой задачи предстают перед учащимися в другом качестве: становится ясным, в каком направлении следует проводить доказательство. Работа с программой должна быть организована так, чтобы имела возможность провести доказательство в традиционной форме обучения.

Для работы учащихся с учебными программами необходимы лишь элементарные сведения о клавиатуре, которые содержатся во встроеной краткой инструкции: управление осуществляется клавишами со стрелками. Вся дальнейшая работа пользователя связана с набором чисел, обозначений отрезков, треугольников, углов, вводом набранной информации с помощью клавиши ввода. В большинстве программ в краткой форме содержатся теоретические сведения, к которым можно обращаться в процессе решения задач. Практически в любой момент можно отказаться от предпринятой попытки, что очень важно для темпераментных учеников, любящих делать поспешные выводы.

Учителям, желающим проводить уроки по данным программам, необязательно уметь программировать. Для успешной работы им достаточно знать назначение основных клавиш (клавиш со стрелками и ввода) и стандартную команду рассылки программы по сети. Для подготовки к уроку достаточно ознакомиться с краткой инструкцией и предлагаемыми задачами. В ходе урока учащимся могут потребоваться консультации, но уже

не по работе с компьютером, а конкретно по решаемым задачам.

Ввиду значительной разветвленности основной части алгоритма программы занимают большой объем памяти, поэтому в большинстве программ предлагается четыре или пять задач. Однако работа основного блока программ не зависит от конкретных задач, поэтому человек, имеющий небольшой опыт написания программ на языке Бейсик и знакомый с графическими особенностями системы MSX-2, способен заменить предложенные задачи на другие. Для этого необходимо изменить координаты точек, задать те величины, которые считаются известными, по-другому нарисовать чертеж и написать условие задачи, внести изменения в блок проверки наличия ответа. Тем самым заинтересованный учитель, имея эталонные программы, способен создать для себя весьма обширный компьютерный задачник. По желанию и предложению заинтересовавшихся учителей автор может помочь им в этом вопросе.

В заключение несколько слов о включенных в программы задачах. Абсолютное большинство задач не входит в школьный учебник по геометрии, многие задачи оригинальны, их мотивы почерпнуты из множества конкурсных задач в ведущие вузы страны, некоторые задачи трудны. Но, как показывают проведенные эксперименты, то, что является трудным при традиционном подходе, оказывается вполне доступным при решении задач с помощью компьютера.

Адрес для справок: 630090, г. Новосибирск-90, ул. Пирогова, 2, НГУ, заведующей сектором школьной информатики Календаревой Нине Евгеньевне.

Л. БРУДМАН, В. РАСПОПОВ
канд. физ.-мат. наук, Киев

Задачи для ПМК

При организации вычислительного практикума важная роль принадлежит системе задач и упражнений, активизирующих теоретические знания учащихся об особенностях команд ПМК. Рассмотренные ниже задачи и упражнения закрепляют знания учащихся о правилах выполнения тех или иных команд. В основе предлагаемых заданий лежит функциональная избыточность выбора команд языка программирования ПМК. Это означает, что некоторые из команд могут быть исключены из употребления, а их функции возьмут на себя соответствующие под-

программы, составленные из оставшихся команд. Например, команду «+» можно заменить эквивалентной последовательностью двух команд: «/—/» и «—».

Упражнения

Испортилась одна из клавиш ПМК, указанная ниже. Составьте программу, позволяющую использовать ПМК в режиме непосредственных вычислений (в режиме АВТ), применяя вместо испорченной клавиши клавишу С/П:

- | | | |
|-------------------|------------|-----------|
| 1) /—/ | 2) + | 3) — |
| 4) \times | 5) \div | 6) $1/X$ |
| 7) $\sqrt{\quad}$ | 8) X^2 | 9) π |
| 10) sin | 11) cos | 12) tg |
| 13) arcsin | 14) arccos | 15) arctg |
| 16) 10^x | 17) e^x | 18) lg |
| 19) ln | 20) X^y | |

При составлении представленных ниже программ учащимися были использованы тригонометрические и алгебраические тождества. Выражение математических операций и функций через другие — полезное упражнение, как правило остающееся за пределами возможностей весьма насыщенной учебной программы по математике (за исключением некоторых формул тригонометрии). Работа над такими упражнениями позволяет актуализировать знания школьников, служит укреплению межпредметных связей.

Например, приведенные программы, в которых используются математические тождества $\sin \alpha = \sqrt{1 - \cos^2 \alpha}$ и $\cos \alpha = \sqrt{1 - \sin^2 \alpha}$, верны не при любых значениях аргумента α . Можно обратить на этот факт внимание учащихся и предложить им составить для ПМК универсальные программы*.

1. 00	↑	0E	6. 00	↑	0E
01	0	00	01	1	01
02	XУ	14	02	XУ	14
03	—	11	03	÷	13
04	C/П	50	04	C/П	50
05	БП	51	05	БП	51
06	00	00	06	00	00
10. 00	F cos	1Г	13. 00	FX ²	22
01	FX ²	22	01	1	01
02	1	01	02	XУ	14
03	XУ	14	03	—	11
04	—	11	04	F $\sqrt{\quad}$	21
05	F $\sqrt{\quad}$	21	05	F arccos	1—
06	C/П	50	06	C/П	50
07	БП	51	07	БП	51
08	00	00	08	00	00
16. 00	↑	0E	20. 00	F ln	18
01	1	01	01	\times	12
02	0	00	02	Fe ^x	16
03	FX ^y	24	03	C/П	50
04	C/П	50	04	БП	51
05	БП	51	05	00	00
06	00	00			

Теперь рассмотрим задачи повышенной сложности, в условии которых оговорено, что учащемуся при программировании не доступна адресуемая память. Их решение будет по силам тем учащимся, которые в

* Ниже приводятся решения для упражнений 1, 6, 10, 13, 16, 20. Остальные предлагаем читателю решить самостоятельно.

совершенстве овладели командами переписи информации в стеке ПМК.

Задача 1. Составить программу для автоматического вычисления значения $n!$ на неисправном микрокалькуляторе, у которого отказали команды обращения к регистрам адресуемой памяти.

Задача 2. Составить программу для автоматического вычисления значений элементов последовательности $a_n = \left(1 + \frac{1}{n}\right)^n$, $n = 1, 2, \dots$, не используя регистры адресуемой памяти ПМК.

Решение первой из этих задач представлено в таблице, с помощью которой можно проиллюстрировать, как промежуточные результаты счета перемещаются по регистрам стека. При решении этой задачи учащиеся применили остроумный способ хранения промежуточных результатов счета в регистрах стека X1, X, Y, Z.

Значение n набирается на клавиатуре, затем следует набрать клавиши В/О и С/П. На индикатор выводится результат — значение $n!$.

81

Программа вычисления значения $n!$							
Ад-рес	Ком-ман-да	Код	Содержание стековых регистров				
			PX1	PX	PY	PZ	PT
			0	n	0	0	0
00	↑	0E	0	n	n	0	0
01	↑	0E	0	n	n	n (p)	0
02	1	01	0	1	n	n (p)	0
03	—	11	1	n-1	n (p)	0	0
04	F $x \neq 0$	57	1	n-1	n (p)	0	0
05	10	10	1	n-1	n (p)	0	0
0,6	\times	12	n-1	p	0	0	0
07	FVx	0	n-1	n-1	p	0	0
08	БП	51	n-1	n-1	p	0	0
09	01	01	n-1	n-1	p	0	0
10	XУ	14	0	p=nl	0	0	0
11	C/П	50	0	p=nl	0	0	0
12	БП	51	0	иное	0	0	0
				n			
13	00	00	0	»	0	0	0

Задания, в условии которых специально оговорено, что в решении нельзя использовать те или иные операции или клавиши, методически целесообразно представить в виде серии однотипных задач, предлагаемых учащимся в порядке возрастания числа вводимых ограничений. Например, в качестве исходной учитель может предложить следующую задачу.

Задача. Составить программу для взаимной замены содержимого регистров «1» и «2».

Ее решение не составит труда для учащихся. Инструкция: нажать В/О, С/П.

00	ИП 1	61
01	П 3	43
02	ИП 2	62
03	П 1	41
04	ИП 3	63
05	П 2	42
06	С/П	50

«Теперь усложним задачу», — предлагает учитель. «Как изменится программа, если в условие задачи ввести ограничение: нельзя пользоваться другими адресуемыми регистрами?»

Учет этого ограничения не вызывает у учащихся затруднений; они используют для хранения промежуточных результатов регистры стека X и Y. Программа, составленная учащимися, может иметь следующий вид. Инструкция: нажать В/О, С/П.

00	ИП 1	61
01	ИП 2	62
02	П 1	41
03	XY	14
04	П 2	42
05	П 2	42
05	С/П	50

Следующим ограничением может быть запрет использовать команду XY. Инструкция: нажать В/О, С/П.

00	ИП 1	61
01	ИП 2	62
02	П 1	41
03	F O	25
04	П 2	42
05	С/П	50

«Введем еще одно ограничение в условие задачи, запретив при составлении программы пользоваться и командой F O. Попробуйте для перемещения информации в стеке применить команду F Vx», — рекомендует учитель.

Решение этой задачи имеет вид. Инструкция: нажать В/О, С/П.

00	1	01
01	ИП 1	61

02	X	12
03	ИП 2	62
04	П 1	41
05	F Vx	0
06	П 2	42
07	С/П	50

Если в число ограничений ввести и запрет на использование команды F Vx, то задача существенно усложняется и ее решение требует определенной смекалки. Целесообразно, разобрав на уроке все предыдущие задачи данной серии, эту задачу задать на дом.

Один из вариантов решения последней задачи может быть таким. Инструкция: нажать В/О, С/П.

00	ИП 1	61
01	ИП 2	62
02	—	11
03	П 1	41
04	ИП 2	62
05	+	10
06	П 2	42
07	ИП 1	61
08	—	11
09	П 2	42
10	С/П	50

Более сложные задачи подобного типа рекомендуется рассмотреть на занятиях кружка или факультатива. Приведем примеры таких задач.

Задача 3. Составить программу для взаимной замены значений элементов массива P [1:4] и Q [1:4].

Задача 4. Составить программу, вычисляющую сумму положительных элементов массива P [1:N], $1 \leq N \leq 13$, не используя (в качестве вспомогательного регистра) для накопления суммы регистры адресуемой памяти.

Задача 5. В регистрах адресуемой памяти $1 \div D$ размещены числа 0 или 1. Составить программу, изменяющую содержимое каждого регистра на противоположное, т. е. с 1 на 0 и наоборот; пользоваться командами условного перехода не разрешается.

И. ГОЛИЦЫНА, И. НАРЫКОВА

Компьютер на уроке физики

В настоящее время одним из главных показателей эффективности обучения становится сформированность у учащихся способности самостоятельно приобретать новые знания в процессе учебной деятельности [1].

В статье мы опишем два типа педагогических программных средств (ППС) по физике, использование которых позволяет орга-

низовать самостоятельную познавательную деятельность учащихся. ППС реализованы на языке «Бейсик-MSX», для КУВТ «Ямаха».

Программными средствами первого типа являются моделирующие ППС, в которых реализованы имитационные модели различных физических явлений или технических устройств. Программные средства второго

типа относятся к диалоговым конструкторским ППС творческого характера.

Основные различия между ними определяются типом управления деятельностью учащихся и характером учебного материала, заложенных в ППС.

При работе с моделирующей программой учащемуся (или преподавателю) предоставлена возможность выбора одного из трех режимов: демонстрация опыта, обучение проведению «дисплейного» эксперимента и исследования, в котором моделируются физический объект (математическая модель) и форма проведения эксперимента (имитационная модель). При этом пользователь строго не регламентирован.

В ППС второго типа осуществляется жесткое управление учебной деятельностью учащихся. Они ориентированы на решение учебных творческих задач. При этом учащийся, отвечая на вопросы и получая необходимую помощь, выступает как партнер ЭВМ по решению задачи.

Для реализации каждого из типов ППС необходим отбор учебного материала различного характера: в моделирующих, как правило, имитируются физические явления или процессы, которые описываются известными математическими зависимостями; для диалоговых характерно обращение к задачам прикладного характера.

1. Моделирующие ППС исследовательского характера (МППС)

МППС предусматривают три режима работы, описанные выше, и имеют общие программные блоки: блок математической модели, блок визуализации физического процесса и процесса измерений, блок данных, блок служебной информации.

Алгоритм деятельности при организации самостоятельной работы [2] учащихся с применением МППС описывается следующим образом.

1. Постановка цели обычно осуществляется преподавателем, так как разработанные нами МППС обладают многоцелевым назначением и охватывают, как правило, не только одно физическое явление, но и целый спектр примыкающих к нему.

2. Наблюдение фактов и явлений производится в режиме демонстрации. При этом возможности ЭВМ позволяют визуализировать в схематичной форме даже те процессы, которые не доступны восприятию человека при обычном проведении эксперимента.

3. Изучение закономерностей, связанных с исследуемым явлением,— центральный этап при работе с МППС. Задавая различные воздействия с клавиатуры, учащийся может

наблюдать за реакцией системы, причем за адекватность таких реакций отвечает математическая модель данного процесса, составляющая ядро МППС.

4. Объяснение явлений и практические выводы, на наш взгляд, учащиеся могут сделать самостоятельно, как без использования ЭВМ, так и с помощью некоторых теоретических и прикладных сведений, заложенных в ЭВМ либо в виде справочной информации, либо в виде демонстрации технических устройств, основанных на изученном явлении.

Разработаны шесть МППС по физике, охватывающие темы «Изотермы идеальных и реальных газов», «Диаграммы растяжения металлов», «Исследование зависимости сопротивления металлов и сплавов от температуры», «Изучение отражения и преломления света», «Изучение полупроводникового диода», «Изучение транзистора».

II. Диалоговые конструкторские ППС творческого характера

Работа учащегося с ППС творческого характера организуется так, чтобы она содержала необходимые компоненты творческой деятельности, перечисленные ниже.

1. Осознание проблемы и возникновение интереса к ней. На это направлена постановка творческой задачи в ППС.

2. Актуализация знаний и их мыслительная обработка учащимися. Этой цели служат содержащиеся в ППС справочник с необходимой для принятия решения информацией и система психологически обоснованной помощи.

3. На каждом шаге работы с программой создается проблемная ситуация, которая побуждает учащегося к поиску ответа в процессе самостоятельной мыслительной деятельности.

4. Обобщение и осмысление полученных при решении задачи знаний. Для этого существует информационный блок, в котором обобщаются полученные в ходе решения задачи результаты.

Мы предлагаем конструкторские диалоговые ППС «Построение принципиальной схемы выпрямителя» и «Построение принципиальной схемы усилителя».

Выполнение программ начинается с постановки задачи. На экран по кадрам выводится текст, содержащий название задачи, общее определение выпрямителя (усилителя), предложение построить принципиальную схему выпрямителя (усилителя). Затем на экран выводится общая схема выпрямителя (усилителя) с краткой характеристикой принципа его действия.

После постановки задачи начинается диалог с компьютером. Ответы на вопросы ЭВМ набираются учащимся с пульта дисплея буквами русского алфавита. Анализ ответа проводится по корневым сегментам всех возможных названий элементов электрических цепей.

В случае неправильного ответа помощь организуется в зависимости от сложности вопроса двумя или тремя шагами. Первые два шага не содержат подсказки, что очень важно для поддержания творческого уровня задачи. Третий шаг помощи содержит подсказку или вывод правильного ответа, т. е. переводит задачу на данном этапе решения из творческой в нетворческую. Ввод неправильного ответа сопровождается комментарием: «К сожалению, Вы ответили неправильно». Правильный ответ на вопрос сопровождается выводом полного ответа с поощрительным комментарием, после чего происходит графическое достроение выведенной на экран схемы; все элементы электрических цепей обозначаются общепринятыми символами.

ППС заканчиваются информационным блоком, в котором обобщаются получен-

ные учащимся знания: выводятся схемы основных типов выпрямителей (усилителей) и краткая информация об их работе.

Экспертная оценка ППС преподавателями физики школ и СПТУ и опыт работы с учащимися показали, что ППС подобного типа могут способствовать организации продуктивной самостоятельной деятельности учащихся, развитию у них навыков творческой работы.

Читателей, заинтересованных в более подробном знакомстве с описанными ППС или в их получении, просим обращаться по адресу:

420063, г. Казань, ул. Исаева, 12, НИИ ПТП АПН СССР.

Л и т е р а т у р а

1. Голин Г. М. Вопросы методологии физики в курсе средней школы. М.: Просвещение, 1987.
2. Буряк В. К. Формирование исследовательских умений и навыков // Среднее специальное образование. 1988, № 11.
3. Пидкасистый П. И. Самостоятельная познавательная активность школьников в обучении. М.: Педагогика, 1980.

М. ЕПИНА

Об организации взаимодействия вуза и школы

Эффективность обучения школьников компьютерной грамотности существенно повышается, если имеется возможность закрепить полученные теоретические знания по информатике навыками работы на вычислительной технике. Однако на сегодняшний день большинство школ обречено на преподавание безмашинного варианта курса ОИВТ. Это приводит к тому, что школьник быстро теряет интерес к теоретическому обучению.

Красноярский институт цветных металлов (КИЦМ) предлагает школам различные организационные формы взаимодействия по вопросам компьютеризации учебного процесса. Здесь ставятся различные цели: от обучения работе с готовыми программами без навыков программирования до подготовки ученика к самостоятельному написанию и отладке программ на ЭВМ.

В 1988 г. был образован Красноярский центр непрерывного образования инженерных кадров (КрасЦНОИК) на базе КИЦМ, в состав которого вошло и несколько средних школ. Одним из направлений работы в звене «школа — вуз» является совершенствование учебного процесса,

в том числе его компьютеризация.

Преподаватели и сотрудники кафедр высшей математики, физики, химии, иностранных языков КИЦМа предложили школам специальные прикладные учебные программы для ЭВМ (моделирующие, контролируемые, тренажеры, вычислительные).

Программы разрабатываются преподавателями и сотрудниками общеобразовательных кафедр и сотрудниками ВЦ института. Они же проводят со школьниками практические занятия на ЭВМ. По плану осуществляется обучение работе на ЭВМ учителей-предметников школ, входящих в КрасЦНОИК. С помощью института в СШ № 63 Кировского района оборудован компьютерный класс, оснащенный ПЭВМ «Электроника БК-0010Ш». Уроки по информатике, а также занятия компьютерного клуба проводят сотрудники кафедры высшей математики. К началу учебного года подготовлены прикладные программы по математике, физике и химии, реализуемые на БК-0010Ш, поэтому отдельные темы по этим предметам учащиеся смогут изучать с использованием ЭВМ.

В 1988/89 учебном году группа учащихся восьмых классов СШ № 15 приступила к занятиям по программированию при кафедре высшей математики института. Занятия проводятся один раз в неделю в часы, отведенные для работы в УПК. В течение трех лет эти учащиеся научатся не только пользоваться готовыми программами, но и овладеют навыками практического программирования. Вместе с аттестатом они получат и специальность лаборанта-программиста. В течение первого года обучения школьники изучили разделы, связанные с алгоритмами, устройством ЭВМ, программированием на языке Бейсик. Особый интерес у школьников вызывали графические возможности программирования.

На теоретических занятиях кроме нового учебного материала, который дает преподаватель, проводится контроль пройденного материала, разбираются составленные учащимися несложные алгоритмы и программы в форме свободного обсуждения. Отладка программы производится ими во время практической части занятий, которая проходит в кабинете информатики кафедры высшей математики, оснащенного ПЭВМ «Электроника БК-0010Ш». Систематическая работа с ЭВМ способствует приобретению навыков самоконтроля, самостоятельному определению и исправлению ошибок, расширению и углублению знаний.

В апреле 1989 г. при кафедре высшей математики начал свою работу кружок «Юный программист» для учащихся V—VII классов ближайших школ. Для первых занятий использовался клавиатурный тренажер, предназначенный для изучения и овладения клавиатурой ПЭВМ «Электроника БК-0010Ш», члены этого кружка получили возможность более раннего выхода на ЭВМ. Целью этих занятий является развитие навыков работы с ПЭВМ, подготовка к освоению курса ОИВТ. Продолжительность занятий 1,5 ч в неделю. В течение первого года обучения дети знакомятся с возможностями машинной графики, графические средства программирования значительно облегчают усвоение учениками сложных структур. Кроме этого, они получают основные представления об алгоритмах, их составлении.

В отличие от городских большинство сельских школ лишены возможности работы даже с программируемым микрокалькулятором. Знакомство с вычислительной техникой у них происходит с помощью схем и рисунков. Поэтому институт предложил еще одну форму работы, теперь уже с сельскими школьниками, а именно: учебно-тематические экскурсии «Геология, компьютеризация, туризм». Экскурсии организуются на основе

договора между Бюро международного молодежного туризма «Спутник» Красноярского КК ВЛКСМ и НТТМ комитета комсомола КИЦМ. «Спутник» взял на себя организацию туристической части, питания и проживания. Учебная и профориентационная части обеспечиваются преподавателями и сотрудниками КИЦМ. Занятия по информатике включают в себя шестичасовые лекционную и практическую части. Программа экспресс-курса, составленная с учетом имеющихся у учащихся теоретических знаний, предполагает краткое повторение алгоритмов, их структур и способов представления, изучение языка программирования Бейсик, реализованного на ДВК-3. В краткой форме дается ознакомительный материал об основных устройствах ЭВМ, их назначении, программном принципе работы ЭВМ. Закрепление знаний по алгоритмам, их структурам происходит во время коллективной работы по составлению конкретных, знакомых каждому школьнику алгоритмов (например, составлялись алгоритм варки картофеля, алгоритм определения кислотности жидкой среды с помощью лакмусовой бумажки и т. п.). При их написании используются только блок-схемы как наиболее наглядный способ записи алгоритмов.

Основная часть лекционных занятий отводится для подготовки учащихся к выходу на ЭВМ. Для этого даются сведения об организации программ, простейших конструкциях языка Бейсик, его основных операторах, а также инструкция по работе на ЭВМ. Совместно с учащимися составляются алгоритмы решения задач из школьных курсов математики, физики; затем эти алгоритмы переводятся в соответствующие программы, объясняется ход исполнения программ и назначение отдельных операторов. Во время лекции были составлены коллективно программы, начиная с простейших («Знакомство») и кончая более сложными вычислительными линейными (определение площади треугольника по формуле Герона), циклическими (табулирование функции) и разветвляющимися (решение квадратного уравнения) программами по математике, с которыми школьники во время практических занятий работают в ВЦ института на ПЭВМ ДВК-3.

Этот экспресс-курс, состоящий из лекционной и практической частей, позволяет школьникам получить минимальные навыки работы на ЭВМ.

В апреле — мае 1989 г. в экскурсиях «Геология, компьютеризация, туризм» приняли участие две группы (60 человек) учеников IX—X классов школ Новоселовского района Красноярского края.

Е. КЕЛЬМАН, В. ЛИТВИНСКИЙ, С. МЕЛЬНИК

Опыт внеклассной работы на СМ-4

Мы расскажем о занятиях со школьниками VII—X классов в одном из кружков (при отделе математических методов автоматизированного проектирования) научно-исследовательского Вычислительного центра при Латвийском государственном университете.

Систематическая работа со школьниками началась в отделе в 1982 г., с 1985 г. школьникам предоставлен дисплейный зал на базе ЭВМ СМ-4. С этого времени занятия ведутся по разработанной на четыре года программе курсов с условным названием «Элементы информатики и применения ЭВМ». Естественно, на содержании курса не могли не сказаться исследования, проводимые в отделе (они связаны в основном с математическим моделированием) и вычислительном центре.

Основные цели занятий — пробудить и развить любознательность и математическую культуру учащихся, расширить их кругозор, ознакомить школьников с возможностями и областями применения ВТ; выработать простейшие навыки алгоритмизации, постановки задач и разработки программного продукта, помочь выбрать будущую профессию. Не ставится целью ни подготовка программистов, ни, тем более, подготовка системных программистов, хотя препятствий для совершенствования в этих областях нет. Подготовку учащихся можно было бы назвать введением в специальности «Математика», «Прикладная математика» и «Математическое обеспечение ЭВМ».

Занятия проводятся 1—2 раза в неделю в течение всего учебного года, каждое продолжается 2—3 ч и состоит из теоретической и практической частей. На теоретических занятиях читаются лекции по новому материалу; решаются задачи, даются индивидуальные задания, устраивается коллектив-

ный разбор программ, составленных на заданную тему всей группой или по индивидуальному заданию кем-либо из школьников. Некоторые занятия проводятся в форме семинаров, которые ведут сами школьники.

Практические занятия проходят в дисплейном зале ЭВМ СМ-4 и состоят на начальном этапе в выполнении упражнений на ЭВМ, затем — в отладке и тестировании программ, индивидуальной работе с учащимися. Руководители занятий помогают выбрать и уточнить постановку задачи, контролируют и направляют разработку программы, активно участвуют в ее тестировании.

Посещение занятий добровольное. Обычно работают 2—3 группы: младшая (VII—VIII классы), старшая (IX—X классы) и временная — группа новичков. Основной набор новых учащихся производится в августе — сентябре на основе традиционных связей со школами Риги. Как правило, несколько новичков появляются в течение года. Если в это время нет временной группы, к ним индивидуально прикрепляются опытные школьники, которые под контролем преподавателя и с его помощью помогают новичкам освоить необходимый минимум.

Занятия проводятся сотрудниками вычислительного центра и подготовленными для этого студентами физико-математического факультета университета. Количество участников кружка (обычно регулярно занимаются 35—40 человек) ограничивается возможностями преподавателей и количеством мест в дисплейном зале. Все занятия проводятся на общественных началах.

С 1985 г. ежегодно в августе организуется летняя школа юных программистов. Занятия продолжаются 4—5 нед по 5—6 раз в неделю и проводятся утром (в той же форме,

что в течение года) и после обеда, когда предоставляется дополнительное машинное время или устраиваются экскурсии. На них школьники знакомятся с историей ВТ (по материалам музея вычислительного центра), различными ЭВМ и системами программирования. Не все, даже из постоянных «кружковцев», участвуют в летних школах. Тем не менее польза и эффективность последних очень высоки — за время летней школы удается сделать больше, чем в течение всего учебного года. Школьники, начавшие заниматься в кружке летом, как правило, усваивают материал лучше своих сверстников, пришедших в учебном году, реже уходят из кружка.

В приложении приведена примерная программа теоретических занятий, рассчитанная на четыре учебных года и четыре летние школы. Каждый учебный год и летняя школа начинаются с изучения (или повторения) правил техники безопасности и внутреннего распорядка вычислительного центра.

Основное содержание курса разбито на темы, обращение к которым на новом уровне происходит несколько раз: понятие о методах, алгоритмах и программах, технология программирования (структурное кодирование, проектирование программ «сверху вниз», выбор алгоритма, построение системы тестов, документирование, стиль написания программ и др.), элементы математической логики (математический язык, высказывания, логические операции, таблицы истинности, запись утверждений с помощью предикатов и др.), решение уравнений и систем уравнений (методы Гаусса, Зейделя, простой итерации, секущих, касательных), простейшие методы оптимизации (половинного деления, «золотого сечения», покоординатного спуска), машинная арифметика (представление целых и действительных чисел, ошибки округления), устройство ЭВМ (архитектура, принципы адресации). Изучаются также системы счисления, последовательности и пределы, комбинаторика, элементы теории вероятностей, теории информации, аналитической геометрии, теории графов и др. Большой интерес у школьников вызывают головоломки, разработанные сотрудником вычислительного центра А. А. Ванасом, их программирование и исследование с помощью ЭВМ.

Для проведения практических занятий разработан задачник [1, 2], который может быть использован в течение всего времени обучения в кружке. Он состоит из 16 разделов, содержит задачи по вышеперечисленным темам, а также по обработке данных (таких, как массивы, слова, алфавиты, геометрическая информация), статистическому моделированию, разработке системных, игровых, обучающих программ и др. Задачи составля-

ны таким образом, что применение ЭВМ для их решения является необходимым, демонстрирует важные особенности и преимущества ЭВМ. Школьникам предоставляется большая свобода в выборе и уточнении постановок задач.

К основным принципам, осуществляемым в ходе занятий, относятся учет возрастных особенностей школьников и их подготовки; изучение тех математических задач и тех элементов их теории, в которых целесообразно применение ЭВМ (при этом ЭВМ отводится роль инструмента, а не объекта исследований); периодический возврат и углубление основных тем, их глубокое изучение; разумный уровень формализации; выявление и демонстрация межпредметных связей; разнообразие задач как по объектам и назначению программ, так и по подходам и методам решения. Теоретический материал разбит на относительно независимые разделы по 10—12 ч, что позволяет, в частности, подключать к занятиям новичков в середине года.

Примерное распределение материала: первый год обучения — разделы 1—3, 4.1—4.3, 6.1, 8.1—8.3, 10.1, 10.2, 12.1—12.3; второй год — 4.3—4.5, 5.1—5.4, 6.2, 6.3, 7.1, 7.2, 10.3, 13.1, 13.2, 13.5, 16.1—16.3, 16.6; третий и четвертый годы — 4.6, 5.5—5.7, 7.3—7.5, 8.4—8.7, 9, 10.3, 10.4, 11, 12.4—12.6, 13.3—13.6, 14, 15, 16.4, 16.5, 17.

Важный момент начального периода занятий — дать правильное представление об ЭВМ. Наилучший способ, на наш взгляд, — начать с редактирования текстов. Это позволяет сразу включить в занятия практикум на ЭВМ, имеет практическую ценность на протяжении всех четырех лет. На СМ-4 при работе в многогерминальном режиме удобен текстовый редактор TED [3]. Простота, удобство и содержательность TEDa, наглядность его работы позволяют уже на начальном этапе обучения легко и естественно ввести и глубоко усвоить такие понятия, как алгоритм, команда, файл, диалог с ЭВМ и т. п., показать богатство идей и применений вычислительной техники, рассеять распространенные заблуждения (например, что ЭВМ — это только очень большой и быстродействующий калькулятор). Учащиеся изучают редактирование текстов в экранном и командном режимах, основные команды и др. Разработаны упражнения, помогающие освоить TED, закрепить и проконтролировать знания.

Известно, насколько важен выбор первого языка программирования. Однако реально использовать на СМ-4 и в качестве первого, и в качестве основного языка программирования возможно только Бейсик. Проведенные эксперименты показали, что ни Паскаль, ни Пролог, ни Фортран, ни Си не позволяют

одновременно работать 12—14 пользователям. Тем самым преимущества этих языков сводятся на нет.

Несколько компенсировать недостатки Бейсика помогает, во-первых, изучение TEda; во-вторых, школьники сразу начинают изучать программирование на Бейсике как структурное, усваивая набор правил [4] для представления основных структур (следование, развилка, цикл) с помощью операторов языка в удобном для восприятия виде. Структурное кодирование особенно полезно для начинающих: заставляя их выделять в тексте программ основные структуры, оно приучает их к дисциплине программирования, облегчает разработку и отладку программ, упрощает контроль за обучающимися.

Естественно, что школьники с удовольствием играют в компьютерные игры. На наш взгляд, мнение о необходимости их применения на этапе знакомства с ЭВМ для снятия якобы имеющегося психологического барьера является предрассудком. Использование видеоигр для мотивации учебной деятельности чревато потерей естественного интереса к ней. Дополнительное время, проведенное за экраном дисплея, приводит только к возбуждению и утомлению учащихся. Нам не известны обоснованные рекомендации по применению компьютерных игр для обучения, хотя мы не исключаем использование элементов игры, игровых обучающих программ и программирования игр. Сама возможность овладеть компьютером дает школьникам сильнейший стимул к учебе.

Эти выводы сделаны в последнее время и, возможно, вследствие того, что в Риге появилось много доступных игровых компьютерных залов.

Вышеописанная работа дала следующие результаты: прошла проверку программа работы со школьниками (особенно программа первых двух лет обучения), найдены и проверены методические приемы, упражнения, разработаны методические пособия [1—6]. Заинтересованы и подготовлены к проведению занятий со школьниками студенты физико-математического факультета Латвийского государственного университета, более 50 школьников окончили кружок, многие из них выбрали профессии, связанные с применением ЭВМ. Некоторые программы, разработанные школьниками, используются для проведения занятий кружка и экскурсий на ЭВМ.

Прежде чем назвать несколько программ школьников, которых создано за последние три года более 300, отметим, что их разработка — не самоцель, а закономерное завершение определенных этапов обучения.

Программа STE предназначена для редак-

тирования небольших текстов (3 строки по 8 символов) и печати их в виде заголовков большими буквами. Программа DIRECTORY разработана в помощь руководителю занятий. Она позволяет просматривать и при необходимости удалять файлы заданного директория. Используется при чистке дисков, защищена от несанкционированного доступа.

Программа PI может быть использована при знакомстве с теорией вероятностей. Она вычисляет число π , определяя статистически (с помощью генератора псевдослучайных чисел) площадь круга единичного радиуса. Программа TEN переводит числа из двоичного или шестнадцатеричного машинного представления в десятичное и обратно. Ее удобно использовать при изучении соответствующих тем. Программа-шутка QASIBAS имитирует при подключении к ней выполнение директив Бейсика. По словам ее авторов, «она приводит в недоумение, а потом в ярость малоопытного, но что-то знающего программиста на Бейсике».

PEPIOSSTM — программа-справочник по периодической системе элементов Менделеева, представляет собой своеобразную мини-СУБД. Она позволяет создать у школьников начальные представления о базах данных.

Назовем также программы, показывающие позиции в играх «15», «PE-XE-SO», «Города», «Жизнь» и др., программы, решающие некоторые классы (нелинейных, диофантовых и др.) уравнений, программы-календари и часы, программы, демонстрирующие свойства чисел, слов, графические возможности алфавитно-цифрового дисплея и др. Отметим, что неплохие программы получаются даже у семиклассников (PI, TEN, поиск делителей числа и др.).

К сожалению, пока не реализованы коллективная (возможно, со студентами) разработка программ, разработка пакетов программ, освоение других языков программирования. Не вполне ясны методы борьбы с детским «хакерством» и компьютерным хулиганством. Требуется дополнительной проверки и коррекции программа третьего и четвертого годов обучения.

Авторы благодарят руководство вычислительного центра за создание условий и помощь при проведении занятий со школьниками.

Литература:

1. Мельник С. Задачи по программированию. Ч. 1. Рига: Изд-во ЛГУ им. П. Стучки, 1987.
2. Мельник С. Задачи по программированию. Ч. 2. Рига: Изд-во ЛГУ им. П. Стучки, 1987.
3. Кельман Е. С., Мельник С. И., Семушин В. И. Краткое описание текстового редактора TED. Рига: Изд-во ЛГУ им. П. Стучки, 1987.

4. Кельман Е., Мельник С. Структурное кодирование на Бейсике. Рига: Изд-во ЛГУ им. П. Стучки, 1986.

5. Кельман Е., Литвинский В., Мельник С. Примеры программ школьников. Рига: Изд-во ЛГУ им. П. Стучки, 1987.

6. Абелиня А., Икауниекс Э., Кельман Е., Семушин В. Краткое описание языка Бейсик RSX. Рига: Изд-во ЛГУ им. П. Стучки, 1986.

Приложение

Программа занятий

1. Введение.

1.1. Знакомство с ВЦ ЛГУ. Правила внутреннего распорядка. Правила техники безопасности. Первая встреча с ЭВМ СМ-4.

1.2. Организация памяти ЭВМ, хранение информации и доступ к ней.

1.3. Экскурсия в музей ВЦ.

1.4. Знакомство с другими типами вычислительной техники.

Литература:

Гельфгат А. Ю., Калейс А. Я., Кочубей И. М. и др. Работа на мини-машинах под управлением ОС-РВ. Рига: Изд-во ЛГУ им. П. Стучки, 1986.

Извозчиков В. А., Марков И. В. Школьники изучают ЭВМ // Физика в школе. 1984. № 4.

Розов Н. Х., Сендов Бл. Х. Предыстория рождения компьютера // Квант. 1985. № 9.

Розов Н. Х., Сендов Бл. Х. История рождения компьютера // Квант. 1985. № 10.

Шмелев А. Детская болезнь компьютерного всеобуча // Информатика и образование. 1987. № 1.

2. Диалог с вычислительной машиной (на примере текстового редактора TED).

2.1. Знакомство с ЭВМ СМ-4 и программой TED.

2.2. Экранный режим TEDa.

2.3. Командный режим TEDa.

Литература:

Ивашин С. Л. TED — краткая инструкция. Новосибирск: Изд. Ин-та автоматизации и электротехники СО АН СССР. 1980. (Распространяется на магнитном носителе.)

Кельман Е. С., Мельник С. И., Семушин В. И. Краткое описание текстового редактора TED. Рига: Изд-во ЛГУ им. П. Стучки, 1987.

3. Введение в язык программирования Бейсик.

3.1. Синтаксис. Разбор элементарных программ.

3.2. Операторы ветвления и цикла и их применение.

3.3. Полный набор операторов языка; встроенные функции.

3.4. Разбор примеров программ.

Литература:

Абелиня А. В., Икауниекс Э. А., Кельман Е. С., Семушин В. И. Краткое описание языка Бейсик RSX. Рига: Изд-во ЛГУ им. П. Стучки, 1986.

Икауниекс Э. Основы языка Бейсик // Информатика и образование. 1986. № 3; 1987. № 1.

Исаков В. Исполнение алгоритмов // Информатика и образование. 1987. № 2.

Кельман Е. С., Мельник С. И. Структурное кодирование на Бейсике. Рига: Изд-во ЛГУ им. П. Стучки, 1986.

Маргузан Б. Я. Как Петя Бейсиков Тоню Соображалкину программировать учил. Рига: Зинатне, 1987.

4. Проектирование и разработка программ.

4.1. Блок-схемы. Программа на псевдокоде.

4.2. Основные структуры: последовательность, ветвление, цикл; их реализация на Бейсике.

4.3. Стиль программирования. Комментарии, выбор имен переменных, оформление текста программы, запись выражений. Принцип дружелюбности.

4.4. Построение системы текстов. Отладка и тестирование программ. Самоконтроль. Защищающее программирование.

4.5. Технология программирования «сверху вниз».

4.6. Документирование и сопровождение программ.

Литература:

Ван Тассел Д. Стиль, разработка, эффективность, отладка и испытание программ. М.: Мир, 1985.

Виллемс А., Пейал Я. Советы по созданию дружелюбного программного обеспечения // Информатика и образование. 1986. № 1.

Григас Г. Начала программирования. М.: Просвещение, 1987. Гл. 5.

Гудман С., Хидетниemi С. Введение в разработку и анализ алгоритмов. М.: Мир, 1981.

Йодан Э. Структурное проектирование и конструирование программ. М.: Мир, 1979.

Каймин В. А. Основы доказательного программирования. М.: Изд. МИЭМ, 1987.

Кельман Е. С., Мельник С. И. Структурное кодирование на Бейсике. Рига: Изд-во ЛГУ им. П. Стучки, 1986.

Майерс Г. Искусство тестирования программ. М.: Финансы и статистика, 1982.

Новичков В., Пылькин А. Рекомендации по оценке качества прикладных программ // Информатика и образование. 1987. № 6.

Рождественский В. В., Хлебутин С. Г. Структурный подход и язык программирования Бейсик // Квант. 1988. № 3.

Штернберг Л. Ф. Циклы, циклы, циклы... // Квант. 1987. № 1.

Лингер Р., Миллс Х., Уитт Б. Теория и практика структурного программирования. М.: Мир, 1982.

Брудно А. Л., Каплан Л. И. Олимпиады по программированию для школьников. М.: Наука, 1985.

5. Стандартные ситуации в программировании.

5.1. Переходы по сложным логическим условиям.

5.2. Обработка рекуррентных формул.

5.3. Обработка массивов на месте.

5.4. Процесс построения диалога. Дружелюбность диалога.

5.5. Оценка эффективности программ. Распределение ресурсов в зависимости от назначения программы.

5.6. Пути повышения эффективности: вычисление констант, арифметические и логические операции, индексация, оптимизация циклов, ввод — вывод.

5.7. Графические возможности алфавитно-цифрового дисплея.

Л и т е р а т у р а:

Буланова Н., Волков В., Широков П. Графика на алфавитном дисплее // Информатика и образование. 1987. № 3.

Ван Тассел Д. Стиль, разработка, эффективность, отладка и испытание программ. М.: Мир, 1985.

Виллемс А., Пейал Я. Советы по созданию дружественного программного обеспечения // Информатика и образование. 1986. № 1.

Каймин В. А. Построение диалоговых алгоритмов // Квант. 1986. № 11.

Карпов В., Карпова О., Новичков В. Графическая информация на алфавитно-цифровом дисплее // Информатика и образование. 1988. № 2.

Косневски Ч. Занимательная математика и персональный компьютер М.: Мир, 1987.

Циман Л. Л. Логические задачи и алгебра высказываний // Квант. 1971. № 4.

Штернберг Л. Ф. Стандартные приемы программирования. Урок 1 // Квант. 1982. № 10.

Штернберг Л. Ф. Стандартные приемы программирования. Урок 3 // Квант. 1982. № 12.

90

6. Численные методы. Машинная арифметика.

6.1. Арифметика в десятичных системах счисления.

6.2. Машинное представление целых и действительных чисел.

6.3. Особенности машинной арифметики.

Л и т е р а т у р а:

Гарднер М. Математические досуги. М.: Мир, 1972. Гл. 36.

Гилл Ф., Мюррей У., Райт М. Практическая оптимизация. М.: Мир, 1985. П. 2.1.

Гутер Р. С. Вычислительные машины и системы счисления // Квант. 1971. № 9.

Демидович Н. Б. Вычисления, ошибки, контроль // Квант. 1973. № 2.

Фомин С. В. Системы счисления. М.: Наука, 1975.

Штернберг Л. Ф. Стандартные приемы программирования. Урок 5 // Квант. 1983. № 2.

7. Численные методы. Функции одного переменного.

7.1. Методы решения уравнений с одним неизвестным: половинного деления, секущих, касательных, итераций.

7.2. Оптимизация функций одного переменного: методы половинного деления и золотого сечения; квадратичная и кубическая интерполяция.

7.3. Интерполяционный многочлен Лагранжа.

7.4. Численное интегрирование: формулы прямоугольников; формулы трапеций и Симпсона; формулы Гаусса. Правило Рунге. Формулы Ромберга.

7.5. Принципы построения программ численного интегрирования с автоматическим выбором шага.

Л и т е р а т у р а:

Банди Б. Методы оптимизации: Вводный курс. М.: Радио и связь, 1988.

Бахвалов Н. С. Численные методы. М.: Наука, 1975.

Болтянский В. Г. Методы итерации // Квант. 1983. № 3.

Вагутен В. Н. Числа С, многочлены, последовательности // Квант. 1973. № 2.

Егоров А. Уравнения и пределы // Квант. 1977. № 10.

8. Численные методы. Решение систем линейных алгебраических уравнений (СЛАУ).

8.1. Совместные и несовместные СЛАУ. Особенности машинной арифметики.

8.2. Метод Гаусса. Выбор главного элемента. Плохо обусловленные системы. Модификации метода Гаусса.

8.3. Матричная арифметика. Матричная форма записи СЛАУ.

8.4. Метод простой итерации для решения СЛАУ. Метод простой итерации для решения систем нелинейных уравнений 2-го порядка и нелинейных уравнений.

8.5. Метод Зейделя и его модификации.

8.6. Метод квадратного корня.

8.7. Трехдиагональная прогонка: решение и применение систем с трехдиагональной матрицей.

Л и т е р а т у р а:

Коллатц Л. Функциональный анализ и вычислительная математика. М.: Мир, 1969. № 13.

Мак-Кракен Д., Дорн У. Численные методы и программирование на Фортране. М.: Мир, 1977.

Маргулис Б. Е. Системы линейных уравнений. М.: Наука, 1960.

9. Численные методы. Методы оптимизации.

9.1. Максимум и минимум функции нескольких переменных. Линии уровня, графическое нахождение минимума функции двух переменных. Функция Розенброка.

9.2. Задача линейного программирования: случай двух переменных, общий случай. Симплекс-алгоритм.

9.3. Задача нелинейного программирования. Методы покоординатного спуска, методы Хука—Дживса, Нелдера—Мида. Метод оврагов. Градиентные методы.

Л и т е р а т у р а:

Алейников Б., Бузыцкий П., Дубсон М. Симплекс-метод // Квант. 1976. № 7.

Банди Б. Методы оптимизации: Вводный курс. М.: Радио и связь, 1988.

Барсов А. С. Что такое линейное программирование. М.: Наука, 1960.

Васильев Н. Б., Гутенмахер В. Л. Прямые и кривые. М.: Наука, 1970. § 7.

Гилл Ф., Мюррей У., Райт М. Практическая оптимизация. М.: Мир, 1985.

Тьмеладзе З. Нелинейное программирование // Квант. 1976. № 1.

Каток А. Б. Экономика и линейные неравенства // Квант. 1971. № 3, 4.

10. Алгоритмы и программы.

10.1. Представление об алгоритмах. Примеры алгоритмов.

10.2. Алгоритмы Евклида и бинарный алгоритм. Решение линейных диофантовых уравнений с двумя неизвестными.

10.3. Алгоритмы игр: игра Баше, игра Виттхофа, «Ним» и другие игры со спичками; игры с симметричной стратегией; игры с «доминантной» стра-

тегией; «Бридж-ит», «Ханойская башня», «Рассада» и «Брюссельская капуста», «Сим» и др.

10.4 Методы, алгоритмы, программы.

Л и т е р а т у р а:

Абрамов С. А. Самый знаменитый алгоритм // Квант. 1985. № 11.

Болл У., Коксетер Г. Математические эссе и развлеченья. М.: Мир, 1986. Гл. 1.

Вагутен В. Н. Алгоритм Евклида и основная теорема арифметики // Квант. 1972. № 6.

Вирт Н. Структуры данных и программы // Современный компьютер. М.: Мир, 1986.

Гальперин Г. А., Корлюков А. В. Бинарный алгоритм // Квант. 1986. № 12.

Гарднер М. Математические головоломки и развлеченья. М.: Мир, 1971. Гл. 6, 14.

Гарднер М. Математические досуги. М.: Мир, 1972. Гл. 5, 30.

Гарднер М. Математические новеллы. М.: Мир, 1974. Гл. 18, 33, 23.

Ершов А. П. Компьютер — алгоритм — алгоритмический язык // Квант. 1985. № 9.

Матулис А. Ю., Савукина А. Ю. «Ферзя в угол», «Цзяньшицзы» и числа Фибоначчи // Квант. 1984. № 7.

Успенский В. А., Семенов А. Л. Решимые и нерешимые алгоритмические проблемы // Квант. 1985. № 7.

Яглом И. М. Две игры со спичками // Квант. 1971. № 2.

11. Методы построения алгоритмов.

11.1. Методы подъема и отработывания назад.

11.2. Построение эвристических методов.

11.3. Динамическое программирование. «Разделяй и властвуй».

11.4. Способы организации перебора. Метод ветвей и границ.

Л и т е р а т у р а:

Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. М.: Мир, 1979. Гл. 2.

Гудман С., Хидетниemi С. Введение в разработку и анализ алгоритмов. М.: Мир, 1981. Гл. 3.

Рейтман М. И. Динамическое программирование // Квант. 1972. № 3.

Бурков В. Н., Рубинштейн М. И. Комбинаторное программирование. М.: Знание, 1977.

12. Элементы математической логики. Логические основы вычислительных машин.

12.1. Логические операции. Таблицы истинности. Операции над множествами. Диаграммы Венна. Преобразования логических выражений.

12.2. Запись утверждений и решение задач с помощью предикатов. Кванторы.

12.3. Строение математического языка. Имена, высказывания, операторы, форматоры.

12.4. Теорема Гёделя. Алгоритмы Маркова, тезис Черча.

12.5. Машина Поста. Машина Тьюринга.

12.6. Архитектура современных ЭВМ. Принцип явной и неявной адресации.

Л и т е р а т у р а:

Гарднер М. Математические новеллы. М.: Мир. 1974. Гл. 31.

Касаткин В. Н. Семь задач по кибернетике. Киев: Вища школа, 1975.

Кудрявская Н., Ломакина И., Приз С. Машина Поста // Квант. 1972. № 5.

Кэрролл Л. Символическая логика // История с узелками. М.: Мир, 1973.

Макаренко Ю. Алгоритмы на словах // Квант. 1977. № 2.

Мендельсон Э. Введение в математическую логику. М.: Наука, 1976. § 5.1, 5.2.

Салтовский А. Н., Первин Ю. А. Как работает ЭВМ. М.: Просвещение, 1986.

Смаллиан Р. М. Как же называется эта книга? М.: Мир, 1981. П. 16.

Смаллиан Р. М. Принцесса или тигр? М.: Мир, 1985. Ч. 4.

Королев Л. Н. Операционные системы. М.: Знание, 1977.

Успенский В. А. Машина Поста. М.: Наука 1988.

13. Элементы аналитической геометрии.

13.1. Декартовы и полярные координаты. Декартовы, цилиндрические и сферические координаты.

13.2. Построение графиков в полярных координатах.

13.3. Векторы, углы, расстояния, площади, объемы.

13.4. Прямые на плоскости. Плоскости и прямые в пространстве. Нормали. Взаимное расположение точки и прямой на плоскости; точек, прямых и плоскостей в пространстве.

13.5. Окружность (сфера), касательная, нормаль, секущая. Площади (объемы) элементов круга (шара).

13.6. Перемещения в плоскости и пространстве.

Л и т е р а т у р а:

Гельфанд И. М., Глаголева Е. Г., Кириллов А. А. Метод координат. М.: Наука, 1973.

Клейн Ф. Элементарная математика с точки зрения высшей. Т. 2. Геометрия. М.: Наука, 1987.

Косневски Ч. Занимательная математика и персональный компьютер. М.: Мир, 1987.

Смогоржевский А. С. Метод координат. М.: Наука, 1957.

Назаретов А. Плоскость в пространстве // Квант. 1982. № 6.

14. Введение в теорию информации.

14.1. Энтропия, условная энтропия, информация.

14.2. Решение задач с помощью понятия информации. Применение к алгоритмам сортировки.

14.3. Элементы теории кодирования. Коды Фано, Шеннона. Оптимальный код Хаффмана.

14.4. Информация и избыточность письменной и устной речи.

14.5. Передача сообщений по каналу связи с помехами.

Л и т е р а т у р а:

Бауэр Ф. Л., Гооз Г. Информатика: Вводный курс. М.: Мир, 1978.

Реньи А. Трилогия о математике. Дневник. Записки студента по теории информации. М.: Мир, 1980.

Садовский Л., Аршинов М. Двоичное кодирование // Квант. 1979. № 7.

Слоэн Н. Дж. А. Коды, исправляющие ошибки,

и криптография // Математический цветник. М.: Мир, 1983.

Шестопал Г. Как обнаружить фальшивую монету // Квант. 1979. № 10.

Яглом А. М., Яглом И. М. Вероятность и информация. М.: Наука, 1973.

Ефимов А. Н. Информация: ценность, старение, рассеяние. М.: Знание, 1978.

Леонтьев В. К. Теория кодирования. М.: Знание, 1977.

15. Элементы теории графов.

15.1. Основные понятия теории графов. Примеры графов и применение их к решению задач. Способы представления графов.

15.2. Поиск расстояния между вершинами: алгоритм Форда—Беллмана, алгоритм Дейкстры, алгоритм Уоршелла, волновой алгоритм, матричные алгоритмы.

15.3. Связность и связность. Определение связанных элементов: алгоритм Уоршелла и матричный алгоритм.

15.4. Построение минимального стягивающего дерева — алгоритм замены. Фундаментальная система циклов.

15.5. Графы и программы. Алгоритмы Реджеевского.

15.6. Эйлеровы циклы. Гамильтоновы циклы. Перебор с возвратом. Задача коммивояжера — метод ветвей и границ.

Литература:

Басакер Р., Саати Т. Конечные графы и сети. М.: Наука, 1974. Гл. 6.

Габович Е. Задача коммивояжера // Квант. 1978. № 6.

Евстигнеев В. Графы и программы // Квант. 1981. № 3.

Кристофидес Н. Теория графов: алгоритмический подход. М.: Мир, 1978.

Липский В. Комбинаторика для программистов. М.: Мир, 1988.

Нивергельт Ю., Фаррар Дж., Рейнголд Э. Машинный подход к решению математических задач. М.: Мир, 1977.

Шейнцвит Р. П. Ветви и границы // Квант. 1972. № 7.

16. Комбинаторика и вероятность.

16.1. Перестановки, сочетания, размещения, перестановки с повторением. Числовые лотереи.

16.2. Треугольник Паскаля. Бином Ньютона, свойства биномиальных коэффициентов. Полиномиальная теорема.

16.3. Метод математической индукции. Фибоначчиевы последовательности. Явный вид рекуррентных последовательностей для линейных соотношений с постоянными коэффициентами.

16.4. Основные понятия теории вероятностей. Решение задач. Назначение вероятностей. Элементы теории матричных игр.

16.5. Сортировка и поиск. Задачи информационного поиска.

16.6. Четные и нечетные перестановки. Игра «15» и подобные ей.

Литература:

Абрамов С. А. Поиск в упорядоченной последовательности и упорядочение // Квант. 1986. № 1.

Беренгат Д. Задача об одной игре в «классы», имеющей случайный характер, или Как добиться, чтобы Джонни больше читал // Математический цветник. М.: Мир, 1983.

Бизам Д., Герцег Я. Игра и логика. М.: Мир, 1975. Ч. 2.

Васильев Н. Б., Гутенмахер В. Л. Комбинаторика — многочлены — вероятность // Квант. 1986. № 1.

Виленкин Н. Я. Комбинаторика. М.: Наука, 1969.

Воробьев Н. Н. Числа Фибоначчи. М.: Наука, 1978.

Гарднер М. Математические головоломки и развлечения. М.: Мир, 1971. Гл. 5.

Гарднер М. Математические досуги. М.: Мир, 1972. Гл. 33.

Гарднер М. Математические новеллы. М.: Мир, 1974. Гл. 17, 26.

Кордемский Б. А. Математика изучает случайности. М.: Просвещение, 1975.

Кемени Дж., Снелл Дж., Томпсон Дж. Введение в конечную математику. М.: Изд-во иностр. лит-ры, 1963. Гл. 3, 5, 6.

Реньи А. Трилогия о математике. Письма о вероятности. М.: Мир, 1980.

Соминский И. С. Метод математической индукции. М.: Наука, 1974.

Тьмеладзе З. Теория игр // Квант. 1977. № 8.

Успенский В. А. Треугольник Паскаля. М.: Наука, 1979.

Фукс Л. Алгоритмы поиска и сортировки // Информатика и образование. 1988. № 2.

Цаленко М. Комбинаторные задачи информационного поиска // Квант. 1979. № 12.

Долгов О. Т. Игра в 15 // Квант. 1974 № 2.

Воробьев Н. Н. Теория игр. М.: Знание, 1976.

Вентцель Е. С. Теория вероятностей (первые шаги). М.: Знание, 1977.

Гнеденко Б. В. Математика и контроль качества продукции. М.: Знание, 1978.

17. Последовательности и пределы.

17.1. Определение и способы задания последовательностей.

17.2. Предел последовательности. Сходящиеся и расходящиеся последовательности. Признаки сходимости. Ряды. Вычисление пределов.

17.3. Предел функции. Непрерывность функции в точке. Непрерывность функции на отрезке, теорема Больцано—Коши и ее применение.

Литература:

Дорофеев Г. В. Пределы последовательностей // Квант. 1974. № 11.

Ефремович В. А. Что такое непрерывность // Квант. 1988. № 6.

Кириллов А. А. Пределы. М.: Наука, 1973.

Перевалов Г. Что значит «для любого ϵ »? // Квант. 1978. № 10.

Программирование логических игр

В курсе информатики рассматривается логическая игра Баше, суть которой состоит в следующем. Два игрока — ЭВМ и человек — ходят поочередно. В начале партии перед играющими находится группа из S предметов. Делая ход, можно взять из группы от 1 до P предметов. Тот, у кого нет возможности сделать очередной ход (так как после хода соперника предметов в группе не осталось, т. е. $S=0$), проигрывает.

Усложняя правила выбора очередного хода, учитель может на факультативных занятиях предложить учащимся для анализа и программирования целый класс логических игр, берущих свое начало от игры Баше. Одновременно учитель имеет возможность познакомить учащихся с современными математическими объектами — графами и их изображением на плоскости.

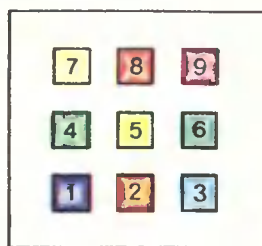
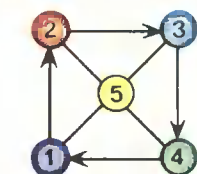
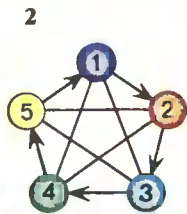
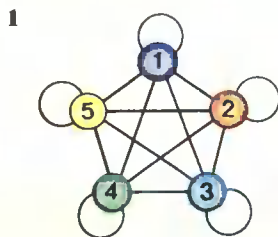
— Рассмотрим геометрическую модель игры Баше в виде взвешенного графа, — предлагает учитель классу, указывая на рис. 1, соответствующий случаю $P=5$. — Этот граф будет использоваться нами в качестве игрового поля, причем каждому ходу соперников соответствует передвижение фишки по дуге в одну из смежных вершин графа, — поясняет он.

Выигрышный алгоритм для этого представления игры Баше заключается в следующем:

1. Вычислить остаток от деления S на $P+1$ и установить фишку на вершину графа с этой меткой. Если S кратно $P+1$, то уступить первый ход сопернику.

2. На любой ход соперника отвечать перемещением фишки по дуге в ту вершину, которая помечена разностью числа $P+1$ и хода соперника.

Задача поиска выигрышного алгоритма будет усложняться по мере того, как учитель будет предлагать учащимся рассмотреть в качестве игрового поля графы с неполным числом дуг или с ориентированными дугами. Примеры таких игровых полей изображены на рис. 2, 3. Их число без труда может быть увеличено учителем.



3

4

— Построить все множество выигрышных алгоритмов (выигрышных стратегий) нам поможет метод полного анализа вариантов, — поясняет учитель. — Этот анализ выполняется с конца партии.

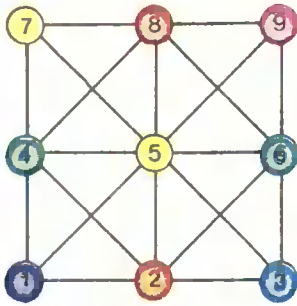
Продемонстрируем суть метода на примере полного анализа логической игры, правила которой сформулировал известный венгерский специалист в области вычислительной техники Антал Чакань. Вот как он описывает ее в книге «Что умеет карманная ЭВМ» (М.: Радио и связь, 1982): «Запишите в вычислительную машину целое положительное число, целесообразно от 50 до 150. Затем каждый из игроков поочередно вычитает из него целое положительное однозначное число (кроме нуля, естественно). При этом выигрывает тот, кто первым получает отрицательное число». Отличая от игры Баше эту игру следующие дополнительные ограничения.

93

А. Чакань предлагает при вычитании учитывать предыдущий ход соперника и расположение цифровых клавиш на клавиатуре микрокалькулятора: «...В карманных ЭВМ цифровые клавиши обычно сгруппированы в три ряда по три клавиши (рис. 4). Если один из игроков выбирает какую-нибудь клавишу, например 4, то соперник может выбрать только ту клавишу, которая находится с ней в непосредственном соседстве, т. е. 1, 2, 5, 8, 7. Если выбрана клавиша 5, то соперник может выбрать любую клавишу, кроме этой. Игра, особенно в конце, требует от игроков большого внимания и комбинационных способностей».

Интуитивное предположение автора этой игры о сложности ее анализа и трудоемкости поиска беспроигрышной стратегии подтверждается и структурой соответствующего графа (см. рис. 5). Существенная неполнота числа дуг этого графа усложняет поиск выигрышного алгоритма в сравнении, например, с нахождением выигрышного алгоритма для игры Баше.

— Будем последовательные результаты



анализа построчно фиксировать в таблице,— предлагает учитель.— В ее первом столбце указывается текущее значение общей для обоих игроков суммы S . Во втором перечисляются выигрышные варианты хода. В третьем — возможные положения фишки на графе, блокирующие для противника всю систему выигрышных ходов, соответствующую текущему значению суммы S .

94 Порядок заполнения строк таблицы следующий.

Строка $S=0$. Любой ход из этого состояния игры ведет к проигрышу. Поэтому столбец «Выигрышные позиции» в этой строке остается пустым. В столбце «Блокировка выигрышных позиций» записываются все цифры от 1 до 9, так как ни одно из положений фишки на графе не дает возможности противнику сделать выигрышный ход (любой ход ведет к $S < 0$).

Строка $S=1$. Будем трактовать любой наш выигрышный ход как ход, блокирующий возможность выигрыша соперника. В данном случае, при $S=1$, такой возможный выигрышный ход единственный — на вершину графа,

помеченную цифрой 1. Этот ход приводит игру к текущему значению суммы $S=0$, а значит, к проигрышу противника. Поэтому в средний столбец таблицы в данной строке записывается цифра 1. В правом столбце перечисляются вершины графа 1, 3, 6, 7, 8, 9, из которых невозможно за один ход попасть в выигрышную позицию 1.

Строка $S=2$. В столбце «Выигрышные позиции» перечисляются выигрышные ходы 1 и 2. Ход 1, переводя игру в состояние $S=1$, тем самым блокирует возможность выигрышного хода 1 сопернику (см. столбец «Блокировка выигрышных позиций», строка $S=1$). Ход 2, переводя игру в состояние $S=0$, блокирует выигрышный ход 2 (см. перечень заблокированных ходов в столбце «Блокировка выигрышных позиций», строка $S=0$).

В столбце «Блокировка выигрышных позиций» в этой строке записываются цифры 7, 8, 9, так как при этих положениях фишки противник не сможет за один ход перейти ни в одну из выигрышных вершин графа — ни в вершину 1, ни в вершину 2.

— И так далее, строка за строкой, последовательно заполняется таблица,— подытоживает объяснение правил учитель.

Затем он предлагает учащимся продолжить заполнение таблицы самостоятельно. Продолжая эту работу и для $S > 66$, учащиеся убеждаются, что выигрышные позиции 1, 3, 7, 9 в дальнейшем повторяются. Следовательно, результаты анализа игры А. Чакая, представленные в таблице, являются полными для любого натурального S .

S	Выигрышные позиции	Блокировка выигрышных позиций	S	Выигрышные позиции	Блокировка выигрышных позиций	S	Выигрышные позиции	Блокировка выигрышных позиций
0		123456789	24	1 3 789	1 3	48	1 3	1 3 789
1	1	1 3 6789	25	1 3 789	1 3	49	1 3 7	1 3 7 9
2	12	789	26	1 3 789	1 3	50	1 3 7	1 3 7 9
3	3	1 34 789	27	1 3 7 9	1 3 7 9	51	1 3 7 9	1 3 7 9
4	1 34	3 9	28	1 3 7 9	1 3 7 9	52	1 3 7 9	1 3 7 9
5	5	5	29	1 3 9	1 3 7 9	53	1 3 7 9	1 3 7 9
6	3 6	1 4 7	30	1 3 9	1 3 7 9	54	1 3 7 9	1 3 7 9
7	1 34 67		31	1 3	1 3 789	55	1 3 7 9	1 3 7 9
8	78	123	32	1 3	1 3 789	56	1 3 789	1 3
9	1 789	1 3	33	1 3	1 3 789	57	1 3 7 9	1 3 7 9
10	12 45 789		34	1 3 7	1 3 7 9	58	1 3 7 9	1 3 7 9
11	3 89	1 3	35	1 3 7	1 3 7 9	59	1 3 7 9	1 3 7 9
12	1 3 9	1 3 7 9	36	1 3 7 9	1 3 7 9	60	1 3 7 9	1 3 7 9
13	1 7 9	1 3 7 9	37	1 3 7 9	1 3 7 9	61	1 3 7 9	1 3 7 9
14	1 3	1 3 789	38	1 3 7 9	1 3 7 9	62	1 3 7 9	1 3 7 9
15	1 3	1 3 789	39	1 3 789	1 3	63	1 3 9	1 3 7 9
16	1 3	1 3 789	40	1 3 789	1 3	64	1 3 7 9	1 3 7 9
17	1 3	1 3 789	41	1 3 789	1 3	65	1 3 7	1 3 7 9
18	1 3	1 3 789	42	1 3 7 9	1 3 7 9	66	1 3 7 9	1 3 7 9
19	1 3 7	1 3 7 9	43	1 3 7 9	1 3 7 9	67	1 3 7 9	1 3 7 9
20	1 3 7	1 3 7 9	44	1 3 7 9	1 3 7 9	.	.	.
21	1 3 7 9	1 3 7 9	45	1 3 7 9	1 3 7 9	.	.	.
22	1 3 789	1 3	46	1 3 9	1 3 7 9	.	.	.
23	1 3 789	1 3	47	1 3 9	1 3 7 9	150	1 3 7 9	1 3 7 9

Любопытно, что заполнение столбца «Выигрышные позиции» выполняется совершенно формально на основании той информации, которая уже содержится в предыдущих строках столбца «Блокировка выигрышных позиций». А именно: в очередную строку столбца «Выигрышные позиции» заносятся те цифры, которые указывают количество строк, отделяющих данную строку от той из предыдущих, в столбце «Блокировка выигрышных позиций» которой встречается соответствующая цифра. В справедливости этого утверждения можно убедиться, проанализировав структуру таблицы. Поэтому можно создать программу для ЭВМ, которая будет сначала самостоятельно по графическому описанию правил игры выполнять полный анализ всех выигрышных стратегий (т. е. будет конструировать таблицу игры), а затем с помощью программного генератора случайных чисел сможет разнообразить свою игру с соперником, переключаясь в ходе партии с одной выигрышной стратегии на другую.

— Последующее внимательное изучение строк таблицы показывает, что выигрышный ход 1 встречается для всех значений S, больших 11, а выигрышный ход 3 встречается для всех S, больших 13,— обращает внимание учащихся на эти и еще некоторые закономерности данной таблицы учитель.— Используя эти особенности информации, представленной в таблице, можно построить такой вариант выигрышного алгоритма, который запишется в виде программы даже в весьма ограниченную память ПМК «Электроника БЗ-34».

Приводимая программа для ПМК «Электроника БЗ-34» разработана учащимися IX класса на факультативных занятиях по информатике. Поясним, как играть с «Электроникой».

Инициатива выбора начального значения общей суммы S принадлежит человеку. Допустим, соперник «Электронике» выбрал начальное значение S=99. Оно набирается с клавиатуры, после чего нужно нажать клавиши В/О и С/П.

— Сейчас «Электроника» подумает над своим первым ходом и сообщит его нам,— объясняет учитель кружковцам.— В данном случае на индикаторе появится сообщение: «300096», т. е. «Электроника» выбрала клавишу 3, тем самым уменьшив первоначальное значение до 96.

Свой ответный ход (в данном примере это одна из клавиш, соседних с клавишей 3,— клавиши 2, 5 или 6) человек вводит в калькулятор. Например, выбрав ход 6, он вводит его, нажимая на клавиатуре две клавиши: 6 и С/П. Игра продолжается.

— В конце партии ПМК обязательно сообщит вам о своей победе выводом на индикатор отрицательного числа,— завершает объяснение правил использования этой игровой программы учитель.

Следует обратить внимание учащихся на то обстоятельство, что из-за ограниченного объема памяти ПМК в данной программе не удалось предусмотреть автоматический контроль корректности хода соперника.

Аналогично построена программа для ПЭВМ. Если при программировании на Бейсике воспользоваться средствами описания табличных величин, то можно реализовать любой из беспроигрышных алгоритмов (или их комбинацию) на основании полной информации о выигрышных стратегиях, представленной в таблице.

Игровая программа на Бейсике

```

10 PRINT "С каким числом начнем игру";
20 INPUT S
30 K=0
40 PRINT "Я размышляю над своим ходом"
50 IF S<>11 THEN 110
60 IF K=2 THEN 90
70 K=8
80 GOTO 160
90 K=3
100 GOTO 160
110 D=(S-2)*(S-3)*(S-5)*(S-6)*(S-8)
120 IF D=0 THEN 150
130 K=1
140 GOTO 160
150 K=S
160 S=S-K
170 PRINT "Веду число";K
180 PRINT "Осталось число";S
190 IF S=0 THEN 240
200 PRINT"Каким будет Ваш ход";
210 INPUT K
220 S=S-K
230 IF S>0 THEN 40
240 PRINT "Вам не повезло. На этот
раз победа за мной!"
250 GOTO 10
260 END

```

Игровая программа для ПМК "Электроника БЗ-34"

Включите ПМК и нажмите клавиши F ПРГ. Введите программу.

00. П†	01. Сх	02. †	03. ИП†	04. ХУ
05. -	06. Fx≥0	07. 53	08. П†	09. 1
10. 4	11. -	12. Fx≥0	13. 26	14. 3
15. ИП†	16. ХУ	17. -	18. П†	19. FВх
20. ВП	21. 5	22. +	23. С/П	24. ВП
25. 02	26. 1	27. +	28. Fx#0	29. 45
30. 3	31. +	32. Fx#0	33. 45	34. 1
35. +	36. Fx#0	37. 45	38. 7	39. +
40. Fx#0	41. 45	42. КИП†	43. ВП	44. 15
45. ХУ	46. 2	47. -	48. Fx#0	49. 42
50. 1	51. ВП	52. 15	53. С/П	54. Сх
55. ВП	56. 53			

Нажмите клавиши F АВТ В/О и введите числа: 1 П1 П4; 2 П2; 3 П3 П6 П7 ПВ ПС; 5 П5; 6 П8; 7 П9 ПА ПД; Сх.

Моделирование и визуализация в программах учебной ориентации

Приобретая компьютерную грамотность, учащиеся сначала осваивают ЭВМ как устройство, взаимодействие с которым, в отличие от других машин, осуществляется в языковой форме. На этом, начальном, этапе рекомендуется решать задачи по известным алгоритмам, закрепляя навык уверенного использования языка программирования.

Опыт работы Малой академии наук «Искатель» позволяет предложить для этого случая следующие классы учебных задач:

96

арифметика обыкновенных дробей;
арифметика многоразрядных чисел;
генерирование замечательных числовых последовательностей (числа Ферма, Мерсенна, Паскаля, Бернулли, Каталана, Лейбница, фигурные числа, числа Фибоначчи и др.);
решение уравнений, в том числе методом сведения к вычислению соответствующей обратной функции;

геометрические задачи, решаемые с использованием метода координат;
задачи символической обработки;
задачи математической логики;

клеточное моделирование, в том числе моделирование игровых полей (шахматы, игра Хейна, тригекс и др.).

Решая задачи предложенной тематики, обучающиеся получают хорошую ориентацию как в возможностях персонального компьютера, так и в приемах и методах алгоритмизации. У них возникает естественная потребность разработать что-то значительное, важное.

Поэтому на следующем этапе у преподавателя появляется возможность сделать серьезный шаг в деле воспитания подлинной алгоритмической культуры. Предлагаемые задачи должны уже требовать разработки модели в соответствующей предметной области, а создаваемые алгоритмы должны в подходящей форме обосновываться, хорошо иллюстрироваться, убедительно разъясняться, доказываться. В сознание учащегося должна быть внедрена мысль о том, что реализация алгоритма требует тщательной разработки, включающей этапы отладки, тестирования и документирования. Учащийся постепенно должен понять, что существует технология программирования.

Одна из первых целей преподавателя — добиться осознания учащимися роли модели, смысла переноса задачи из одной предметной

области в другую. Здесь, как показывает опыт, особую роль играют алгебраические модели геометрических задач, использующие, наряду с декартовой, недекларативные системы координат. Полезным оказывается и рассмотрение задач на использование в качестве моделей десятичных систем счисления, в том числе систем с целым отрицательным основанием, систем с симметричным основанием, систем остаточных классов и смешанных систем.

По форме модельные программы, имеющие учебную ориентацию, должны быть визуальными. Визуализация используемых моделей в интерактивном режиме позволяет составителю программы освоить графические средства и средства символической обработки, а преподавателю — получить программное средство, используемое в учебном процессе.

Под модельной визуальной программой учебной ориентации будем понимать интерактивную программу решения определенного класса задач, которая с помощью графических средств раскрывает сущность метода решения, а также отображает динамику используемых информационных структур.

Покажем, как удачно подобранная числовая модель может быть использована для решения систем линейных диофантовых уравнений (ЛДУ). Визуализация этой простой модели и эффективно работающий на ее основе алгоритм оставляют в сознании учащихся глубокий след.

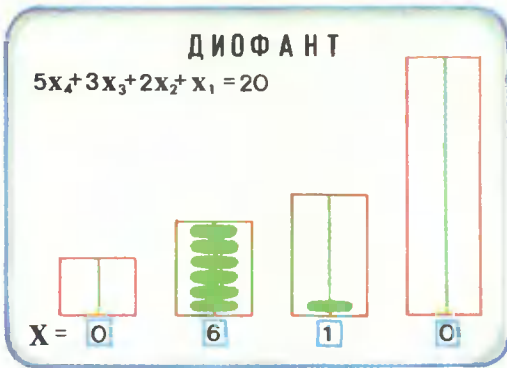
Программа находит все решения ЛДУ $a_n x_n + a_{n-1} x_{n-1} + \dots + a_1 x_1 = b$, $a_i, x_i, b \in \mathbb{Z}^+$, причем каждый шаг процесса отображается на экране. Каждое решение ЛДУ трактуется как число в смешанной системе счисления с переменной верхней границей цифры каждого разряда. «Число», представляющее решение ЛДУ,

$$(x_n, x_{n-1}, \dots, x_1)$$

набирается на своеобразном сумматоре, в котором размер регистра, представляющего «цифру» x_i , динамически изменяется. Можно интерпретировать модель как n -рядные счеты с переменным числом косточек в каждом ряду.

Состояние счет, соответствующее одному из решений уравнения

$$5x_4 + 3x_3 + 2x_2 + x_1 = 20,$$



показано на рис. 1. Эффективность модели обеспечивается тем, что очередное решение ЛДУ получается сложением текущего решения с единицей в выбранной системе счисления. Перенос при сложении выполняется по правилам, очень похожим на обычное сложение с единицей на счетах. Если единица прибавляется к i -му разряду и этот разряд содержит максимальное значение (оно переменное), то i -й разряд устанавливается в нуль, а в разряд $i + 1$ переносится 1. Так как значение x_1 всякого предыдущего решения ЛДУ равно верхней границе первого разряда чисел, то при сложении с 1 x_1 обращается в 0, а число

$$S = a_n x_n + \dots + a_2 x_2$$

после сложения с 1 не превышает b . Тогда $\text{INT}((B-S)/A(1))$ — новая верхняя граница x_1 относительно числа S , представленного всеми разрядами, кроме первого. Если $(B-S)/A(1) = (B-S) \setminus A(1)$, т. е. если $B-S$ нацело делится на $A(1)$, то, положив

$$x_1 = (b-s) / a_1,$$

получим очередное решение ЛДУ.

Например, решение (1, 0, 0, 15), следующее за приведенным на рис. 1, получим в соответствии с таблицей.

x_4	x_3	x_2	x_1	
0	6	1	0	$\text{ВГ}(x_1) = 0$
+			1	
1	0	0	0	
+			15	
1	0	0	15	$\text{ВГ}(x_1) = 15$

Алгоритм решения ЛДУ легко обобщается до алгоритма решения системы ЛДУ. Модельная визуальная программа решения систем ЛДУ строится на основе алгоритма, приведенного на распечатке 1. Сложение с единицей на специальном сумматоре моделируется в строках 140—200.

```

10 REM СИСТЕМА ДИОФАНТОВЫХ УРАВНЕНИЙ
20 INPUT "K-ВО УРАВНЕНИЙ; K=BO
      НЕИЗВЕСТНЫХ ";M,N
30 DIM X(N),B(M),A(M,N),S(M),T(M)
40 FOR I=1 TO M : S(I)=0 : FOR J=1 TO N
50 PRINT "A(";I;" ";J;" "):
      INPUT A(I,J) : NEXT J
60 PRINT "B(";I;" "): INPUT B(I) :
      NEXT I
80 FOR P=1 TO M : IF A(P,1)<>0 THEN 100
90 NEXT
95 PRINT "НЕВЕРНО ЗАДАНА МАТРИЦА
      КОЭФФИЦИЕНТОВ" : END
100 X(1)=INT((B(P)-S(P))/A(P,1))
110 FOR K=1 TO M :
      IF A(K,1)*X(1)<>B(K)-S(K) THEN 140
120 NEXT
130 FOR K=1 TO N : PRINT X(K) :
      NEXT : PRINT : R=R+1
140 J=2
150 FOR I=1 TO M : T(I)=S(I)+A(I,J) :
      IF T(I)>B(I) THEN 190
160 NEXT
170 X(J)=X(J)+1
180 FOR I=1 TO M : S(I)=T(I) : NEXT :
      GOTO 100
190 FOR K=1 TO M :
      S(K)=S(K)-A(K,J)*X(J) : NEXT :
      X(J)=0 : J=J+1
200 IF J<=N THEN 150
210 PRINT "R=";R
220 END

```

Отметим попутно, что, используя изложенный выше метод и соответствующий ему базовый алгоритм, достаточно просто перейти к решению задачи линейного целочисленного программирования (ЛЦП). Совокупность решаемых этим методом задач можно свести к следующей логической цепочке: решение ЛДУ, решение системы ЛДУ, решение линейного диофантова неравенства (ЛДНР), решение системы ЛДНР и, наконец, решение задачи ЛЦП (распечатка 2).

```

10 REM ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ
20 REM A(I,J),B(I) - КОЭФФ. СИСТЕМЫ
30 REM C(I) - КОЭФФ. ФУНКЦИИ ЦЕЛИ
40 INPUT "K-ВО НЕРАВЕНСТВ; K=BO
      ПЕРЕМЕННЫХ ";M,N
50 DIM X(N),B(M),A(M,N),S(M),T(M),C(N)
60 FOR I=1 TO N : PRINT "C(";I;" "):
      INPUT C(I) : NEXT
70 FOR I=1 TO M : S(I)=0
80 FOR J=1 TO N
90 PRINT "A(";I;" ";J;" "):
      INPUT A(I,J)
100 NEXT J
110 PRINT "B(";I;" "): INPUT B(I)
120 NEXT I
130 INPUT "K-ВО НЕРАВЕНСТВ ВН.
      ПОВЕРХНОСТИ";L
140 Y=0
150 FOR K=L+1 TO M : IF S(K)>B(K)
      THEN 200
160 NEXT
170 R=R+1
180 Z=0 : FOR K=1 TO N : Z=Z+C(K)*X(K) :
      NEXT
190 IF Z>Y THEN Y=Z : FOR K=1 TO N :
      V(K)=X(K) : NEXT

```

```

200 J=1
210 FOR K=1 TO L : IF S(K)+A(K,J)>B(K)
                               THEN 250
220 NEXT
230 X(J)=X(J)+1
240 FOR I=1 TO M : S(I)=S(I)+A(I,J) :
                               NEXT : GOTO 150
250 FOR K=1 TO M :
    S(K)=S(K)-A(K,J)*X(J) : NEXT :
    X(J)=0 : J=J+1
260 IF J<=N THEN 210
270 PRINT "R=";R
280 IF R>0 THEN FOR K=1 TO N :
    PRINT V(K); : NEXT : PRINT :
    PRINT "MAX=";Y
300 END

```

Еще одним характерным примером является создание визуальной программы численного интегрирования. После исследования эффективности методов прямоугольников, трапеций и парабол (Симпсона) поиск более совершенных алгоритмов численного интегрирования приводит к идее неравномерного разбиения отрезка интегрирования в зависимости от поведения функции. Процесс оптимизации такого разбиения хорошо поддается визуализации. Демонстрируется график подынтегральной функции, динамика разбиения отрезка интегрирования на части, последовательность закрашивания частичных площадей, для которых достигнута требуемая точность. Кроме того, в программе иллюстрируется использование классических структур данных в прикладных (в данном случае вычислительных) задачах. Процесс половинного деления частичных отрезков сопровождается построением на экране соответствующего, в общем случае несбалансированного, бинарного дерева. Управление последовательностью обработки частичных отрезков сопровождается сменой на экране состояний стека, хранящего их границы.

В визуальной программе численного интегрирования используется оконная технология. Состояния графического, стекового окон и окна, содержащего дерево разбиения,

синхронно меняются и сопровождаются текстовыми ремарками (рис. 2).

В качестве примера на распечатке 3 приведена подпрограмма визуализации бинарного дерева, соответствующего разбиению отрезка интегрирования.

```

630 "-----ИЗОБРАЖЕНИЕ ДЕРЕВА-----"
640 IF SIZE<=1 THEN RETURN
650 SUCCZ=SIZE\2 : ISUCC=IT+GAP :
    JL=JT-SUCCZ : JR=JT+SUCCZ
660 FOR K=JL TO JR STEP SIZE
670 LINE (JT,IT)-(K,ISUCC),3 :
    CIRCLE (K,ISUCC),2,3
680 NEXT K
690 IT=ISUCC : JT=JL : SIZE=SUCCZ
700 II(P)=IT : JJ(P)=JR : ST(P)=SIZE :
    RETURN

```

Здесь GAP — расстояние между уровнями дерева; SIZE — расстояние между вершинами одного уровня; IT — у-координата вершины данного уровня; JL, JR — x-координаты корневых вершин соответственно левого и правого поддерева; RAD — радиус изображения вершины; C — цвет изображения; SUCCZ, ISUCC — рабочие переменные.

В строке 700 координаты IT и JR и величина SIZE сохраняются в стеке (P — его указатель). Обращение к подпрограмме построения очередных ветвей дерева выполняется, если на соответствующем вершине дерева частичном отрезке точность не достигается и необходимо дальнейшее разбиение пополам. Если же точность достигнута, то в основной программе выполняется операция выталкивания из стека

IT = II(P) : JT = JJ(P) : SIZE = ST(P),
соответствующая переходу на более высокий уровень дерева. Перед записью в стек выполняется операция P=P+1, после считывания операция P=P-1.

В рассмотренных выше задачах использование подходящих моделей позволило построить эффективные алгоритмы решения.

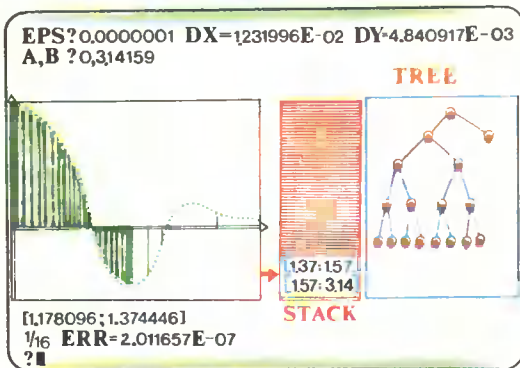
Можно, однако, привести и такие примеры, когда реализация той или иной модели чересчур сложна и не может предлагаться в качестве учебной задачи.

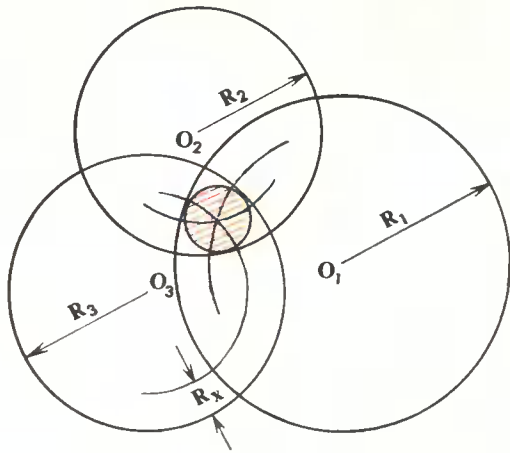
Весьма поучительным в этом смысле является решение известной задачи Аполлония: провести окружность, касающуюся трех данных.

Существует классический метод решения этой задачи, основанный на использовании инверсии. Однако алгоритм, основанный на инверсии, оказывается слишком громоздким. Трудоемкой оказывается реализация алгебраической модели. На рис. 3 показано, что центр искомой окружности (см. заштрихованный круг) лежит на пересечении трех вспомогательных окружностей. Иначе говоря, координаты центров искомых окружностей и величин их радиусов являются корнями

98

2





системы уравнений которую требуется решить в общем виде:

$$(x-x_i)^2 + (y-y_i)^2 = (R_i \pm R_x)^2, \quad i=1, 2, 3.$$

В подобных случаях можно прибегнуть к использованию конструктивных геометрических моделей, т. е. реализовать режим непосредственного управления пользователем поиском решения на экране ЭВМ. Идея состоит в использовании клавиатуры и экранных средств для подбора радиусов вспомогательных окружностей при фиксированных центрах, заданных в программе. Подбор радиусов ведется шаг за шагом и имеет целью нахождение точек пересечения трех вспомогательных окружностей. Как

только такая точка будет получена на экране, необходимо получить ее координаты. Для этого пользователь вручную перемещает маркер, совмещая его с точкой пересечения. Координаты маркера вычисляются в программе, что автоматически позволяет найти центры искоемых окружностей с точностью до пиксела.

На распечатке 4 — фрагмент программы управления построением вспомогательных окружностей.

```

265 '-----УПРАВЛЕНИЕ-----
266 '---ВСПОМОГАТЕЛЬНЫМИ ОКРУЖНОСТЯМИ---
270 A$=INKEY$ : IF A$="" THEN 270
275 '---RH - ТЕКУЩИЕ РАДИУСЫ,
    '---RR - ИСХОДНЫЕ РАДИУСЫ,
    '---DEV - ПРИРАЩЕНИЯ RR;
    '---72, 80 - КОДЫ КЛАВИШ
280 IF LEN(A$)=2 AND ASC(RIGHT(A$,1))=72
    THEN RH1=RR1 : RH2=RR2 : RH3=RR3 :
      RR1=RR1+DEV1 : RR2=RR2+DEV2 :
      RR3=RR3+DEV3 : GOTO 300
290 IF LEN(A$)=2 AND ASC(RIGHT(A$,1))=80
    THEN RH1=RR1 : RH2=RR2 : RH3=RR3 :
      RR1=RR1-DEV1 : RR2=RR2-DEV2 :
      RR3=RR3-DEV3 : GOTO 320
300 '---ВЫХОД ИЗ БЛОКА УПРАВЛЕНИЯ-----

```

Кроме модельных визуальных программ «Диофант», «Видеоинтегратор» и «Аполлоний», назначение которых описано в этой статье, авторами разработан и ряд других программ учебной ориентации.

Для вас, пользователи ЭВМ УКНЦ

Центр НТТМ «Поиск» г. Ангарска

может принять вас на абонентное обслуживание и предоставить вам следующую документацию и программное обеспечение:

1. Руководство по ОС ФОДОС/RT11.
2. Документацию по дополнительному программному обеспечению.
3. Новые версии системного программного обеспечения.
4. Пользовательские программы — курсы машинописи, экзамены по правилам дорожного движения, обучающие, игровые и др.

Для занесения вас в картотеку Центра необходимо выслать на наш адрес:

665806, г. Ангарск-6, а/я 5933, Центр НТТМ «Поиск»
 письмо с вашими данными и конверт с обратным адресом для высылки вам каталога программного обеспечения и литературы.

Авторы признательны всем, кто откликнулся на публикацию статьи «База данных школы» (Информатика и образование. 1988 № 5). Из полученной почты видно, что в техникумах и вузах, в компьютерных клубах и центрах, в ВЦ базовых предприятий школ есть реальная потребность в решении задачи информационного обслуживания учебного заведения. Привлечение учащихся к такой работе дает двойной эффект: позволяет автоматизировать обработку информации с помощью школьных ЭВМ и привить им практические навыки решения реальных информационных задач.

Н. ПАХОМОВА

учитель программирования СШ № 444, Москва

Юношеское научное общество школьной информатики

100 Как организовать работу учащихся? Что можно поручить школьникам? Какое программное обеспечение для этого использовать? На эти вопросы, пришедшие в наш адрес, мы и постараемся ответить в статье. Мы используем ЭВМ типа ЕС и реляционную СУБД УНИСОН (Универсальная информационная система оперативного назначения).

Юношеское научное общество школьной информатики (ЮНОШКИН) школы № 444 (далее просто ЮНО или Общество) зарегистрировано во Дворце пионеров и школьников Первомайского района Москвы в феврале 1987 г. Неформально оно образовалось намного раньше. Школьники, которые решали первые задачи информационного обслуживания школы, на пороге своего выпуска задумались о преемниках. Родилась идея создания разновозрастного общества учащихся, объединенных общим делом — информационным обслуживанием школы.

С первых дней сентября нового учебного года ЮНО начинает процедуру организационного возрождения. Организаторские функции, ведущую роль в решении информационных задач примеряют на себя вчерашние девятиклассники, а ныне десятиклассники. В IX и VIII классах на классных собраниях, через стенную газету они рассказывают о работе Общества, решенных и нерешенных задачах, планах на будущее, об уставе и организационной структуре. Для новичков проводится начальный курс обучения силами одиннадцатиклассников, которые еще какое-то время помогают и консультируют. Десятиклассники распределяют между собой направления деятельности. Таким образом, первый этап каждого нового учебного года заключается в смене действующих лиц, передаче организационных и управляющих функций, а самое главное, в передаче эксплуата-

ции базы данных школы (БДШ), задач информационного обслуживания школы. Завершается он общим собранием, на котором переизбираются президент Общества, вносятся добавления или исправления в устав, утверждается список вновь вступивших, для которых оформляются членские книжечки.

Такая книжечка, точнее, членский билет члена ЮНО содержит информацию о ее владельце (фамилия, имя, секция, дата вступления) и о его делах, т. е. о творческой деятельности, организаторских функциях, преподавательской и кураторской работе. На последней странице приведены выдержки из устава ЮНО как памятка о правах и обязанностях его членов.

Не последнее значение для ребят имеет символика: значок, девиз, эмблема — все это изобретается членами Общества. Сколько выдумки и талантов при этом открывается! Символика Общества отражается и в стенной газете «Экран ЮНО», и в экслибрисе на выдаваемой заказчикам продукции и в значке — символе единения в общем деле.

Творчество школьников проявляется не только в информатике и программировании, постановке и реализации задач информационного обслуживания школы, но и в организационной структуре совета Общества и всего Общества в целом. Распределение руководящих функций в совете, т. е. введение должностей и распределение обязанностей осуществляют сами ребята, учитывая специфику работы Общества, его потребности и способности ребят, входящих в совет. Принципы свободы самореализации и демократизма позволяют школьникам раскрывать и проявлять самые разнообразные способности. Надо заметить, что никто не препятствует принятию в совет учащихся IX и VIII класса, достаточно компетентных,

уважаемых товарищами по ЮНО и предлагающих себя для выполнения каких-либо функций, необходимых для всего Общества. В основном же членами совета являются десятиклассники.

Состав и обязанности членов совета ЮНО

Президент — официальное лицо, выступающее от имени совета перед членами Общества или от имени Общества перед представителями других ученических обществ, организаций, администрации школы или внешкольных учреждений.

Ученый секретарь — оформляет протоколы собраний Общества, заседаний совета и членские билеты. Заносит в членские билеты записи о выполнении общественных обязанностей и поручений, о решенных научно-технических задачах, творческой работе, о выступлении на ученических научных конференциях с докладами, об участии в каких-либо научно-технических конкурсах, а также о кураторской работе для решения задачи преемственности в Обществе.

Администратор базы данных школы отвечает за правильное ведение БДШ, т. е. поддерживает информацию, хранящуюся в базе в адекватном состоянии, контролирует доступ к информации (санкционированный доступ), предоставляет всю необходимую членам ЮНО информацию о состоянии БДШ или ее части.

Организатор-диспетчер осуществляет сбор исходной информации, необходимой для решения задач информационного обслуживания школы, путем непосредственной и регулярной работы с информационными бюро классов. Это информация об изменении списочного состава класса, текущей успеваемости учащихся, пропуске учебных дней,

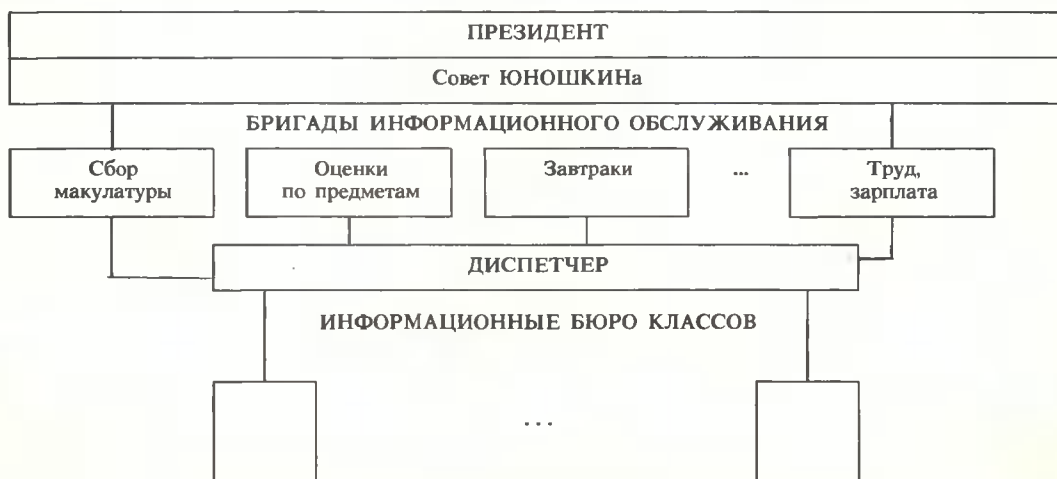
количестве сданной макулатуры, об оценках состояния учебников и т. д.

Главный редактор бумажных информационных изданий информирует других членов ЮНО о лекциях, собраниях, заседаниях совета. Для этого он составляет и вывешивает на стенде соответствующие объявления. В его ведении находится также выпускаемый один раз в месяц информационный орган — стенная газета «Экран ЮНО», предоставляющий в доступной даже неспециалистам форме информацию о деятельности Общества, решаемых задачах, повседневных проблемах и достижениях. Проспекты, инструкции для различных категорий пользователей программной продукции, выпускаемой ЮНО, конспекты и учебные брошюры создаются при самом активном участии главного редактора и его помощников.

Бригадир курирует группу новичков. Он помогает восьми-девятиклассникам освоить клавиатуру, обучает простейшим операторским действиям, необходимым для подготовки данных, контролирует их практическую работу. В его обязанности входит контроль за явкой во время работы и обучающей лекции, аккуратностью и полнотой ведения конспектов лекций, выполнением обучающих упражнений. Одним словом, бригадиры решают важную задачу подготовки преемников.

Экономист-учетчик ведет учет объема и качества выполненных работ, определяет коэффициент трудового участия каждого члена Общества. При подведении итогов работы такой скрупулезный учет поможет мотивировать как моральные, так и материальные поощрения.

Консультант по СУБД УНИСОН — член ЮНО, проявивший большую активность в изучении принципов эксплуатации возмож-



ностей языка общения СУБД УНИСОН, способный не только понять сам, но и доходчиво объяснять другим все, что необходимо для практического ее использования при решении задач. Важность его функций становится понятной, если иметь в виду, что никаких учебных пособий или популярной, доступной всем школьникам литературы по данной тематике пока нет. Нами используется комплект документации для взрослых программистов-пользователей. Возможны личные консультации с авторами-разработчиками.

Консультант по СВМ ОС ЕС ЭВМ активно решает задачи создания необходимых для Общества процедур, которыми пополняются ученические виртуальные машины СВМ ОС ЕС ЭВМ, помогающие членам ЮНО освоить использование этих процедур. Надо заметить, что в поставляемом пользователям варианте СВМ такие процедуры не предусмотрены. Они необходимы для упрощения доступа и эксплуатации СУБД УНИСОН и базы данных.

В настоящее время у школьного ЮНО имеется самое главное — крепкое ядро, опыт работы по решению задач информационного обслуживания школы, жаждающее работы молодое пополнение и четкая организационная структура.

За каждым ребячим делом, как правило, стоит компетентный наставник-взрослый. Организующую и координирующую, направляющую и контролирующую роль играет учитель информатики. Но учитель не последняя инстанция в решении всех дел, а советчик и опора в решении тех вопросов, которые ставят ребята. Учитель — организатор научной работы и грамотного исполь-

зования вычислительной техники.

Кроме учителя в работе ЮНО принимают участие и другие взрослые. Это консультанты-программисты базового предприятия.

Таким образом, коллектив ЮНО, осуществляющий информационное обслуживание школы, имеет разновозрастную структуру. Найденная форма работы с учащимися позволяет освободить их личность от формализма, заорганизованности.

Общее дело — ядро, центр притяжения ребят по интересу, то, что создает коллектив, наполняет его жизнь конкретным содержанием. Для разновозрастного коллектива такое дело должно, как правило, требовать не разового участия, а систематического выполнения определенной работы, приводящей к конкретному результату.

Педагогической целью работы разновозрастного коллектива является развитие творческой личности учащегося, его трудолюбия, стремления к социальной направленности труда (т. е. приложению сил для решения задач, выдвигаемых обществом) и, что очень важно, воспитание коллективизма. В разновозрастном коллективе решение таких задач, как выдвижение неформального лидера, преемственность в работе и передача знаний от старших младшим, ролевые взаимоотношения при единой цели деятельности, посильное участие каждого и ощущение значимости в общем деле, имеет свою специфику и преимуществва по сравнению с коллективом сверстников.

Общества, подобные описанным выше, реально организовать в современной школе, в которую пришли компьютер и новые информационные технологии.

Кооператив «Компьютерные игры»

ПРЕДЛАГАЕТ

программное обеспечение для пользователей компьютеров БК-0010, БК-0010.01, ДВК-2М, ДВК-3М и классов информатики КУВТ-86.

Банк кооператива насчитывает более 1500 системных, учебных и игровых программ.

Поставка возможна на ГМД и на компакт-кассетах.

Расценки значительно ниже государственных.

Списки программ высылаются бесплатно.

Заявки направлять по адресу: 199034, Ленинград, а/я 437.

Диагноз ставит компьютер

Желая проверить искренность человека, мы часто говорим: «Посмотри мне в глаза». В них можно прочесть радость и гнев, испуг, тревогу, грусть. Верно подмечено: «Глаза — зеркало души». А вот, что глаза — точнейший «индикатор» здоровья, известно далеко не каждому. Между тем еще три тысячи лет назад древние лекари Востока судили по радужной оболочке — ирису — о состоянии организма пациента.

...В комнате полумрак. Пациент сидит в удобном кресле перед телевизионной камерой. Освещено только его лицо. Впечатление такое, будто в студии идет запись телепередачи. Но это не так. Дело происходит в клинике медицинского факультета Университета дружбы народов им. П. Лумумбы. Мы на сеансе иридодиагностики.

Врач набирает на клавиатуре персонального компьютера задание, и на экране монитора появляется увеличенное изображение радужки. Стучит принтер. Через несколько минут готова распечатка. На бумажной ленте текст — оценка проанализированного компьютером состояния здоровья больного.

Беседую с человеком, благодаря которому иридодиагностика вошла в практику отечественного здравоохранения, профессором Е. С. Вельховером.

— *Иридодиагностика. Что это — погоня за модой на необычное или метод, действительно необходимый в медицинской практике?*

— Долгое время считалось, что радужка выполняет лишь механическую работу: сужает и расширяет зрачок, осуществляет переток внутриглазной жидкости. Однако недавние исследования доказали, что ирис — часть центральной нервной системы, полномочный представитель мозга. Радужка осуществляет двустороннюю связь между раздражителями внешней среды и внутренней сферой организма.

Наша радужка осуществляет постоянный взаимообмен информацией со всем организмом. Если какой-то орган заболевает, он начинает посылать в ирис сигнал бед-

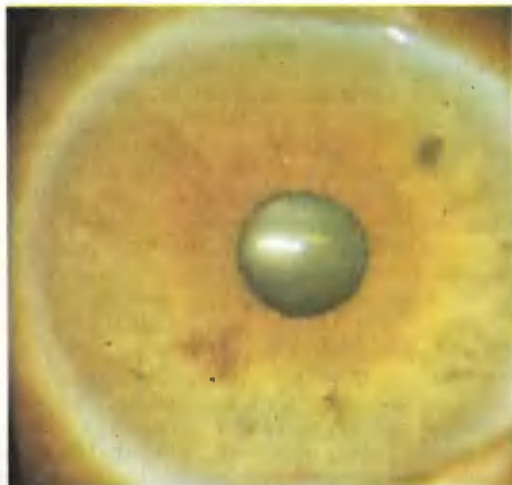
ствия, и те оставляют на радужке отметины. Наш организм как бы спроецирован на внешнее полотно радужки. У ребенка, если он не болен с раннего возраста, ирис относительно чистый, незамутненный. С годами на нем в виде точек, пятнышек, насечек будет проступать портрет нездоровья.

Иридолог обращает внимание на плотность, цвет, четкость рисунка, чистоту радужки. По этим признакам можно судить об особенностях здоровья человека, его тонусе, о сопротивляемости организма в случае заболевания, интоксикации, травмах, стрессах.

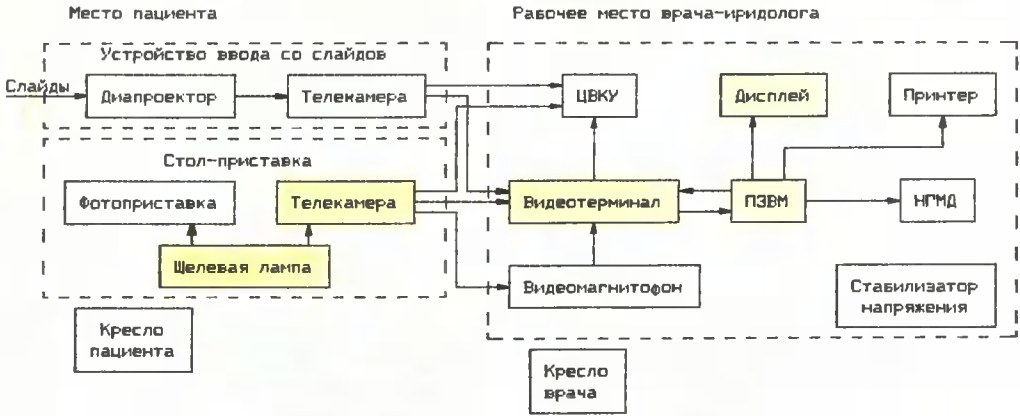
— *Как же правильно прочесть эту необычную историю болезни?*

— Условно круг радужки можно разделить на секторы. Каждый орган представляет один из них. Вверху — головной мозг, внизу — ноги и т. д. О состоянии желудка и кишечника «рассказывает» кольцо, опоясывающее зрачок. Однако и сегодня в диагностике по ирису иридолог должен во многом полагаться на свой опыт, ведь малейший сдвиг точки на радужке может запутать врачей.

— *Значит, рассчитывать на то, что этот метод станет массовым, трудно?*



Структурная схема



— Еще недавно это было так. Теперь на помощь врачам пришли компьютеры. Сотрудниками нашей лаборатории вместе с инженерами научно-производственного объединения «Комета» создан компьютерный комплекс «Ирис» — первый в мире опыт применения компьютеров для ириодиагностики.

О принципах работы комплекса рассказывает заместитель главного конструктора объединения «Комета» В. Бондур.

— Цветная телекамера увеличивает и передает изображение радужки на монитор для визуального наблюдения. Цвет радужки при этом сохраняется. Одновременно изображение в виде кодированных цифровых сигналов поступает в специальный процессор. В этом электронном устройстве выявляются границы естественных зон радужки: зрачковая кайма, цилиарный пояс и др. Анализируются интегральные иридологические признаки радужки. Тип ее, который определяет генетическую конституцию организма, плотность, форма и так далее. Здесь же выявляются аномалии радужки. Их 22 главные, наиболее часто встречающиеся. Детально анализируется каждая. Исследуются яркость, цвет, место расположения, геометрические характеристики и многие, многие другие. Всего процессор исследует 43 информативных признака. Затем данные передаются в управляющий компьютер. Он проводит их статистический анализ. Что это значит? Поясню. При визуальной диагностике врачи полагаются на свой опыт, условное деление ириса на сектора. А если аномалия находится на стыке секторов, куда ее отнести? Отсюда вероятны ошибки, и большие. Мы предложили создать банк изображений с наиболее вероятным изображением органов на схеме и с помощью методов математической статистики определять самый достоверный

вариант. Иными словами, признаки той или иной болезни. Для этого компьютеру необходимо проанализировать более 15 тыс. вариантов.

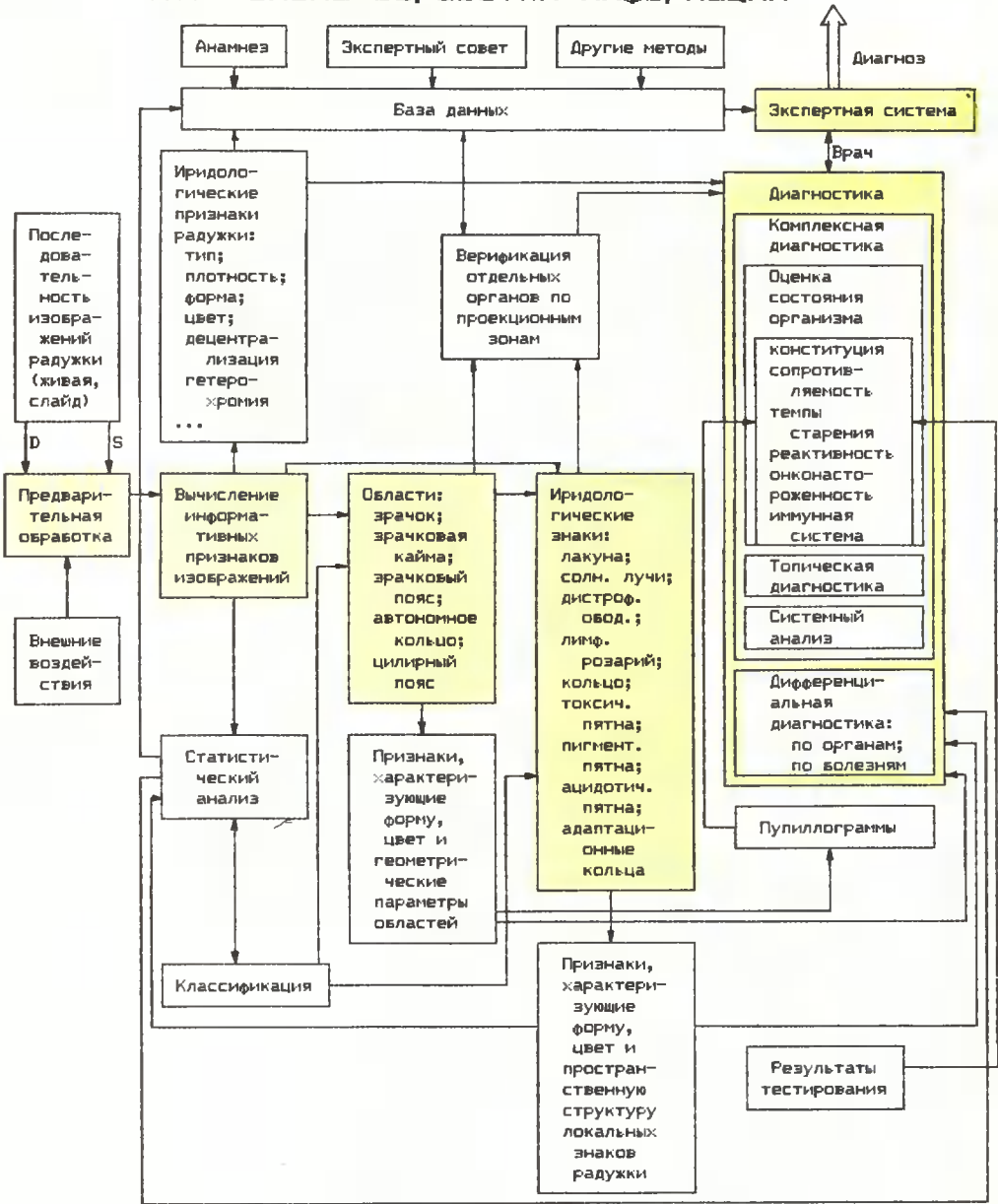
— Пока речь шла о наиболее часто встречающихся болезнях. А как быть в случае, если недуг редкий?

— Наш комплекс готов ответить и на этот вопрос. Для этой цели мы используем метод диагностики, основанный на применении теории нечетных множеств. Принадлежность выявленных аномалий к признакам, свойственным данной болезни, определяется по специальному правилу с помощью опытных специалистов.

Результаты диагностики «Ирис» выдает в виде заключения о клиническом состоянии здоровья пациента. Компьютерный комплекс может работать в режиме экспресс-диагностики — это важно для массовых обследований населения. Скрининг-диагностика — выборочное, более углубленное исследование, скажем, сердца, почек. И наконец, возможна полная диагностика всего организма человека. В первых двух случаях ответ готов через пять мин, в третьем — через 15—30. Хотел бы отметить, что компьютерная ириодиагностика позволяет определить не только болезни человека, но и его наследственный статус, предрасположенность к недугам, темпы старения, скорость реакции организма, что очень важно для космонавтов, летчиков, спортсменов. Комплекс «Ирис» исследует также психоэмоциональную и иммунную системы организма, способность его сопротивляться вредным воздействиям окружающей среды. Комплекс может выдавать эти сведения раздельно и вместе для общей оценки здоровья пациента. Для этого в него заложена специальная экспертная система.

— Евгений Сергеевич, болезни как битвы

Блок-схема обработки информации



оставляют «шрамы» на «теле» радужки. Это я понял. Но как можно определить психо-эмоциональное состояние человека? Оно ведь постоянно меняется.

— И наши душевные невзгоды проявляются на радужке в виде различных знаков, которые можно учесть. Скажем, способность зрачка адаптироваться к свету. Исследуется также количество и расположение так называемых колец сокращения — стрессовых колец. Это «складки» на радужке.

Они образуются в период эмоциональных сдвигов.

— И последний вопрос, профессор. Что ж за недуг оказался у пациента, на обследовании которого я присутствовал?

— У него заболевание щитовидной железы. Причем в ранней стадии. Думаю целенаправленное лечение в сравнительно небольшой срок даст положительный эффект.

Методички из США

Каждый месяц в десятках школ появляются компьютерные классы. Учителя вместе со старшеклассниками осваивают их, а на переменах в дверь с завистью заглядывают младшие школьники. Со временем у учителей появляется желание начать преподавание информатики пораньше, но по каким программам это делать? Мне не раз доводилось отвечать на подобные вопросы, основываясь на собственном опыте, а теперь появилась возможность адресовать учителей к зарубежному. Книга Б. Хантера* отражает опыт, накопленный в процессе коллективных усилий по внедрению компьютеров в учебно-воспитательный процесс в школах США. Нам сейчас приходится решать те же проблемы, и хотя компьютеров типа «Apple» в наших школах нет, методический опыт учителей Америки во многом будет полезен всем, кто взялся за трудное дело компьютеризации нашей многострадальной школы.

С первых строчек чувствуется рука опытного методиста, умеющего ясно и лаконично излагать свои мысли. Прочитав начало первой главы: «Для кого написана эта книга?»:

«Книга «Мои ученики работают на компьютере» написана с целью оказать помощь каждому, кто причастен к использованию компьютеров в процессе школьного обучения. Она будет полезной административным работникам системы школьного образования, учителям, другим членам коллектива школы, местным органам управления, методистам и учащимся при решении следующих вопросов:

что необходимо знать детям о компьютерах и обработке информации;

как новые цели сочетаются с программами по математике, естествознанию, родному языку и общественным дисциплинам;

как использовать компьютеры в классе;

какое материальное обеспечение необходимо и откуда его можно получить.»

Обоснованию важности компьютерной грамотности посвящена вторая глава книги «Следует ли детям использовать компьютеры в школе?».

Принципиальное содержание книги изложено в третьей главе «Обзор курса ком-

пьютерного обучения». На вопрос «Чего мы хотим добиться?» авторы отвечают: «Наша задача заключается в достижении всеобщей компьютерной грамотности». И следом дают весьма своеобразное определение компьютерной грамотности: «Это все то, что необходимо знать каждому человеку, чтобы уметь общаться с компьютером и иметь о нем представление, для полноценной жизни в информационном обществе». Сформулировав затем цели обучения, авторы излагают организацию программы курса. Здесь выделяются шесть направлений:

1. Использование и разработка алгоритмов.

2. Работа с готовыми компьютерными программами.

3. Фундаментальные понятия о компьютерах.

4. Области применения компьютеров.

5. Воздействие компьютеров на общество.

6. Программирование.

При этом выделяются следующие группы учащихся: с подготовительного по II класс; III — IV классы; V — VI классы; VII — VIII классы. Для каждой из групп ставятся свои конкретные цели и задачи, в соответствии с которыми в главах 4—7 дается около 90 описаний урочных планов, достаточно подробных, каждый из которых может быть использован при подготовке к уроку.

Все планы построены по единому методическому замыслу:

цели;

предварительная подготовка учащихся; оборудование;

продолжительность занятий;

формы организации учебной деятельности; методические рекомендации учителю;

задание к уроку;

дополнительные упражнения.

Объектами исследования на занятиях становятся задачи из различных школьных дисциплин, предметы и явления окружающего мира. Начальное знакомство с программированием осуществляется на основе языка Лого. Для некоторых занятий приводится текст обучающих программ на языке Бейсик.

В «Послесловии к советскому изданию» приведены практические рекомендации по адаптации материала книги к нашим условиям, высказываются некоторые критические замечания.

* Хантер Б. Мои ученики работают на компьютерах: Книга для учителя / Пер. с англ. — М.: Просвещение, 1989.

Редко кто из учителей пользуется чужими методичками, как правило, предпочитают разрабатывать свои. Но 90 лаконичных урочных планов вполне могут послужить основой, которую каждый учитель сумеет «адаптировать» к своей манере изложения, имеющей-ся технике и программному обеспечению.

А отсутствующие программы могут стать объектом разработки на факультативах.

Почитайте эту книгу с карандашом в руках!

О. КОЗЛОВ,
г. Серпухов

Алгебра на дисплее

Компьютерный всеобуч предполагает изучение математики и физики (а в перспективе и гуманитарных наук) с помощью миниатюрного электронного помощника — персональной ЭВМ. Советоваться с ней нужно будет не от случая к случаю, а постоянно, в том числе и готовя домашнее задание. Иными словами, компьютер органично вписывается в ход самого учебного процесса, становясь той самой «антропогенной средой», без которой не мыслится жизнь современного человека. Первыми это поняли математики. Они стали интенсивно внедрять в свои курсы элементы программирования, а то и составлять настоящие «электронные учебники», которые пользователь «перелистывает» простым нажатием клавиши. Появились компьютерные практикумы по алгебре — древнейшей ветви математической науки. Но переход от вычислений на бумаге к цифрам на дисплее таит в себе ряд тонкостей, неучет которых подчас ставит начинающего пользователя в трудное положение.

Обобщению опыта работы при обучении «алгебре на компьютере» посвящен недавно вышедший «Практикум»¹. Его авторы — сотрудники лаборатории вычислительных методов механико-математического факультета МГУ.

Практикум родился не на пустом месте: уже в течение пяти лет студенты мехмата имели возможность использовать целую систему компьютерных практикумов. Основная их цель — повысить эффективность практических занятий и семинаров за счет роста знаний, получаемых в единицу времени, и исключения непроизводительных затрат, неизбежных при работе на классной доске с мелом. Опыт показал, что именно на этих занятиях студенты, по существу, впервые сталкиваются с особенностями машинной арифметики и часто оказываются беспомощными при выборе оптимальной вычислительной схемы расчета того или иного элемен-

тарного алгебраического выражения. Возьмем хотя бы случай решения квадратных уравнений на ЭВМ (с. 16 «Практикума»). Решение задается известной школьной формулой. Кажется, следует ввести с клавиатуры коэффициенты и все остальное — только дело машинной техники. Однако здесь-то и кроется загадка «размытости» высвечиваемой на экране искомой функции. Неучет «машинного нуля», забвение об ограничениях используемого компьютера (конечность физической памяти, отнюдь не мгновенное быстроедействие и пр.) приводят к неприятным сюрпризам. Авторы разбирают несколько поучительных примеров, дают ценные рекомендации.

Кстати, книга, обильно насыщенная учебными примерами, акцентирует внимание читателя и на иного рода «подводных камнях». Скажем, какой алгоритм следует предпочесть при решении вполне конкретной задачи? Абстрактная «теория сложности алгоритмов», усиленно развиваемая на стыке таких дисциплин, как логика, теория чисел и дискретная математика, не всегда может дать верный ответ. Причина: эта теория, как правило, недостаточно учитывает специфику работы конкретной ЭВМ с действительными числами, так называемую «концепцию действительного числа», заложенную в архитектуру современной ЭВМ. Поэтому авторы на простейших примерах поясняют самые общие принципы выбора алгоритмов и построения конкретных вычислительных схем для алгебраических выражений. Анализируются небольшие фрагменты компьютерных программ (на Фортране).

Авторы поясняют, почему для многих ЭВМ быстроедействие программы можно сильно повысить, убрав из цикла условные операции и даже лишь изменив порядок хранения данных в памяти. При этом может оказаться, что алгоритм, требующий больше арифметических операций, в действительности работает значительно быстрее. Вот поэтому операции типа условного перехода часто называют операциями, понижающими результирующую производительность ЭВМ.

¹ Размыслов Ю. П., Ищенко С. Я. Практикум по вычислительным методам алгебры: Учебное пособие. М.: Изд-во МГУ. 1989.

В книге сделан небольшой экскурс в компьютерную технику завтрашнего дня — в так называемые «векторные процессоры» и «распределенные системы», что отражает стремительный бег НТР в области аппаратного обеспечения.

Практикум, несомненно, заинтересует широкие круги читателей, и прежде всего преподавателей и учащуюся молодежь, стремящуюся освоить новые, нетрадиционные подходы к изучению таких «вечных наук», как математика. К числу неоспоримых до-

стоинств книги следует отнести высокий теоретический уровень изложения основ компьютерной алгебры и удачные примеры, взятые из опыта программирования. Следует пожелать авторам при переиздании «Практикума» — необходимость которого продиктована самой жизнью — дать в приложении листинги полных программ по типовым задачам линейной алгебры.

В. ФРОЛОВ,
канд. техн. наук

Электронный мозг

108 Нередко компьютер сравнивают с мозгом. Причем речь, как правило, идет о ставших уже обычными цифровых компьютерах, которым для решения поставленной задачи нужна жесткая программа. Но с полным правом это сравнение можно применить лишь к нейрокомпьютерам, разрабатываемым в одной из лабораторий НАСА (США).

Архитектура нейрокомпьютера напоминает высшую нервную систему человека с ее многочисленными перекрестными связями между нервными клетками и моделирует миллиард нейронов и триллион синапсов человеческого мозга. Причем каждый нейрон представлен простым процессором, позволяющим за микросекунду решить дифференциальное уравнение, описывающее поведение одного нейрона.

В чипе размером 9×9 мм расположено более 4 тыс. подобных процессоров, работающих одновременно. В нейрокомпьютере чипы объединяются в сложную параллельную структуру. Предусмотрена возможность использования нейрочипа и в качестве сопроцессора в обычных цифровых компьютерах.

Главное достоинство нейрокомпьютера — способность решать неформализуемые задачи — основано на таких его замечательных качествах, как способность к самообучению и адаптации. Это означает, что нейрокомпьютер не требует строгой программы для решения задачи, достаточно лишь сформулировать цель и предоставить ему все

имеющиеся исходные данные. Создать нейрокомпьютер, способный самообучаться не только на программном уровне, но и на аппаратном уровне, — следующая главная цель сотрудников лаборатории.

До начала работ над созданием нейрокомпьютера он был смоделирован программным путем. Работа с моделью показала значительное скоростное преимущество нейрокомпьютера в решении многих задач. Например, проектирование чипа осуществляется нейрокомпьютером в 100—1000 раз быстрее, чем это происходит с использованием обычных средств. Аналогичный выигрыш в производительности был достигнут и при нахождении оптимальных маршрутов перемещения армий во время боевых действий по космическим снимкам и при решении задачи распознавания зрительных образов. При этом вероятность правильной идентификации изображения достигла 87 %, тогда как обычные цифровые системы узнают показанный объект не более чем в 65 случаях.

Высокая скорость работы и возможность адаптации и самообучения позволяют использовать нейрокомпьютеры в сложных автономных космических системах не только для управления аппара-

том, но и для принятия решений по поведению аппарата в условиях постоянно меняющейся обстановки, для ремонта неисправных систем и даже для выдачи рекомендаций в центр управления полетом.

Поиграть? В планетарий!

Оборудованные современными техническими средствами планетарии все больше из сугубо научно-просветительских учреждений превращаются в центры отдыха. Напичканные компьютерами, видеосистемами и сложной механикой планетарии преподносят астрономическую информацию в столь привлекательном виде, что иначе, как игрой, эти контакты с космосом теперь и не назовешь.

В одном из планетариев Северной Ирландии, например, помимо того что общее управление движением звезд под куполом осуществляется с помощью компьютера, его посетителям предоставляется возможность почувствовать себя пилотами межпланетного корабля. Если кто-то не решается взять управление кораблем на себя, видеокomпьютерная система проведет его одной из звездных дорог сама. Если же вы хотите проследовать своим маршрутом, берите в руки штурвал и путешествуйте по любимым местам Солнечной системы. Компьютер подберет нужный видеодиск с очередным звездным пейзажем или видом планеты, обеспечит оптимальный режим полета и быстрое возвращение на родную Землю.

ЧТО?
МОЖЕТ
ЭВМ

Неделя информатики

Пишет вам учитель математики и информатики Шуляк Игорь Владимирович. Живу я и работаю в селе. У нас в школе есть микрокалькуляторы МКШ-2. А о персональных компьютерах приходится только мечтать.

Но хотя в нашей школе и нет ЭВМ, ребятам предмет очень нравится. С помощью старшеклассников мы обучаем в кружке элементарным понятиям по информатике учеников средних классов, оборудовали кабинет информатики, составили раздаточный материал практически на каждый урок, собрали много книг по предмету, больше 30 папок с вырезками.

Но все же этого, оказывается, мало, и мы решили проводить ежегодно в школе неделю информатики. Подготовку к ней начинаем за 1,5—2 месяца. За это время ребята готовятся к конкурсу КВН, помогают составлять викторину по информатике, сочиняют стихи. За две недели до открытия вывешивается план недели.

Расскажу, как мы проводили неделю информатики в прошлом году, в конце II четверти.

Понедельник

Открытие недели информатики: на линейке перед учениками школы появляются Его величество король Алгоритм и Ее высочество королева Программа. Сопровождают их верные слуги «нули» и «единицы» (ученики старших классов). Первый глашатай королевства Байт зачитывает Указ Его величества об открытии недели и сообщает, что самые активные участники недели станут почетными членами королевской Академии наук и будут награждены.

В этот же день «нули» и «единицы» рассказывают учащимся V — IX классов об ЭВМ и ЭВТ, их применении в народном хозяйстве, медицине, гуманитарных науках, демонстрируют работу микрокалькуляторов.

Вторник

На второй день проводится конкурс газет, в котором принимают участие все желающие классы. Газеты могут иметь различные названия, например «Алгоритм», «Дисплей», «Компьютер», «Компьютерный калейдоскоп», «Информатика для всех» и т. д. Кроме информации об ЭВМ и ЭВТ, ученики помещают в газетах фельетоны, рисунки. В этот же день учащиеся знакомятся с заданиями викторины по информатике, которая включает вопросы по истории развития информатики, ЭВТ, ребусы, кроссворды.

Среда

В этот день проводится КВН между командами X и XI классов. Ребята любят подобные соревнования и поэтому старательно готовятся к ним. В программе КВН — семь конкурсов, среди которых и такие традиционные, как «Приветствие», «Разминка», «Домашнее задание», «Конкурс капитанов». От участников требуются не только юмор и находчивость, эрудиция и литературные способности, но и знания по информатике.

Четверг

В этот день можно провести тур карикатур. В конкурсе принимают участие все желающие классы. Каждый представляет несколько рисунков, а компетентное жюри определяет победителей. В этот же день проводится и радиогазета. Ученики XI класса рассказывают о применении роботов в народном хозяйстве.

Пятница

В пятницу проводится выставка ученических работ, в частности макетов роботов, выполненных учениками из различных мате-

риалов. В этот же день проводится олимпиада. Ее цель — отобрать учеников на районную олимпиаду по информатике.

Суббота

Заканчивается неделя вечером информатики. Учащиеся XI класса рассказывают о

поколениях ЭВМ, путях развития ЭВТ. Проводится конкурс на лучшего чтеца, ученики читают свои стихи на тему об информатике. Награждаются самые активные участники недели. А затем король Алгоритм закрывает неделю информатики.

И. ШУЛЯК,
с. Злазно,

Костопольский р-н, Ровенская обл.

ПОЧТОВЫЙ ЯЩИК

В пятом номере за прошлый год нашим читателям была предложена анкета. С ее анализа мы и начинаем сегодняшний обзор писем.

Небольшой объем выборки не позволяет делать глубокие обобщения, однако некоторый, пусть и локальный, статистический срез представляет интерес. Почти половина ответов пришла от учителей, по 14 % — от школьников, студентов и инженеров; возраст читателей, как и можно было ожидать, самый разный — от 16 до 50 лет.

Средняя оценка, выставленная журналу, — «четыре с минусом». Мы не считаем ее низкой; она отражает, по-видимому, не столько компоновку и качество материалов, сколько разнообразие интересов и потребностей аудитории, полностью удовлетворить которые невозможно в рамках небольшого по объему и периодичности журнала с малым штатом сотрудников. Кроме того, часть наших читателей, не избалованных обилием компьютерных изданий, забывают, что в названии журнала есть слово «образование», определяющее основное направление нашей деятельности.

Рубрики «Методика», «Педагогический опыт» и другие предназначены в основном для читателей, занимающихся педагогическими проблемами использования вычислительной техники в образовании: учителей, методистов, преподавателей, научных сотрудников. Этим объясняется появление в 1989 г. объемных публикаций методических материалов к новым учебникам. Уделяя много внимания «Роботландии» — курсу информатики для младших школьников — и открывая раздел «НИТ в дошкольном образовании», мы стремимся отразить на страницах журнала процессы, происходящие на разных этапах образования. Отвечая на анкету, учителя положительно оценивают публикацию этих материалов. К авторам статей идут многочисленные запросы. В то же время в письмах отмечается большая потребность в задачах, и непременно с решениями, в освещении опыта творческих учителей-

практиков. К сожалению, сотрудники научных учреждений и вузов присылают материалы гораздо чаще, чем учителя.

Большой интерес читателей вызывает зарубежный опыт компьютеризации образования.

Многие корреспонденты отрицательно отзываются о содержании рубрики «Общие вопросы». Мы постараемся учесть высказанные замечания и наполнить ее более конкретными и практически полезными материалами.

Практически все читатели одобряют публикацию рекламных объявлений. Очень многие просят публиковать материалы по программированию на ассемблере, об операционных системах, архитектурах ЭВМ, советы по доработке компьютеров и т. п. Журнал постарается поддерживать эти направления. Со следующего номера мы начнем печатать цикл статей по ассемблеру БК-0010, вскоре возродим «Справочный листок»; идут поиски авторов, которым можно было бы заказать статьи по ОС; на подходе публикации по архитектуре и конструкции БК-0010, в частности его схема, которой не хватает столь многим.

Часто встречаются просьбы публиковать листинги программ — игровых, прикладных, педагогических, системных; напечатать курс Паскаля и повторить курс Бейсика; организовать клубы УКНЦ, «Агата», «Ямахи». К сожалению, такие вещи, как правило, лежат за гранью возможностей редакции. Листинг серьезной программы на алгоритмическом языке займет 20—30 с. журнала; программы в кодах компактней, но и менее читабельны, повышается вероятность ошибки при попытке их ввода в ПЭВМ. Но главное даже не это. Опубликовать программу — значит сделать ее общедоступной. Авторы же серьезных программ, как правило, профессионалы, обоснованно желающие получить деньги за свою работу. Журнального же гонорара в большинстве случаев недостаточно даже для оплаты одного экземпляра программы, не говоря уже о полном ее выкупе в подарок читателям.

Описания алгоритмических языков, экзо-

тических операционных систем типа ПРОС, схема компьютера «Атари» и т. п. вряд ли появятся в журнале. По языкам издано уже довольно много книг, и они продолжают выходить (впрочем, об этой теме речь пойдет ниже); уделить внимание каждой ОС и каждой марке ПЭВМ при шести номерах в год и особой способности советских компьютеростроителей разнообразить ассортимент ПЭВМ не представляется возможным.

Клубы «Агата», УКНЦ и т. п.— это большая организационная и текущая работа, которую можно проделать лишь совместно с энтузиастами — пользователями этих машин. От них мы и ждем предложений.

Многих читателей интересуют зарубежные новинки, перспективы вычислительной техники, компьютерные вирусы; мы будем рассказывать об этом.

А теперь перейдем к письмам.

«Хватит уже «экспериментально» преподавать по учебному пособию под ред. А. П. Ершова. Пора перейти к делу. В большинстве школ, хоть и в малом количестве, есть ПЭВМ, и уже можно параллельно изучать алгоритмический язык и программирование. Я думаю, это дело пойдет, но где соответствующая литература? Прошу вас помочь, выслать нужные книги наложенным платежом» (Н. И. Конохова, пос. Красный Ключ).

Подобные письма приходят часто. Чем тут можно помочь? Конечно, высылать книги мы не в состоянии. Постараемся информировать об интересных книгах из перспективных планов издательства, чтобы вы могли заказать их в своих магазинах или выписать по «Книга — почтой»; будем знакомить с отрывками из рукописей, готовящихся к изданию; а пока пользуйтесь библиотеками и межбиблиотечным абонементом.

Некоторые поступления в редакцию вызывают тревогу. Они регистрируются как письма, потому что состоят из двух-трех исписанных от руки листочков; при чтении, однако, выясняется, что это статьи. Работать с рукописными текстами трудно, но здесь речь не об этом. Просто именно так зачастую выглядят материалы, в которых учителя делятся своим опытом. Впрочем, они могут быть оформлены по всем правилам, перепечатаны на машинке, но их содержание заставляет задуматься о степени успешности информатизации образования. Вот отрывок из одной из таких статей.

«...Причем при составлении программы можно сделать так, чтобы машина вела диалог с учеником. Машина задает вопрос и сразу же реагирует на ответ. При правильном ответе на дисплее ЭВМ высвечивается «молодец», при неправильном — «подумай», «посмотри учебник»»

А вот часть программы — один из вопросов и реакция на него.

```
2000 PRINT "УПРОСТИТЬ ВЫРАЖЕНИЕ"
2010 PRINT "45*X+55*X=?"
2020 PRINT "НАБЕРИТЕ ОТВЕТ И НАЖМИТЕ
                                КНОПКУ ВК"
2030 INPUT ОТВЕТ$
2040 IF ОТВЕТ$="100*X" THEN 2060
2050 PRINT "ПЛОХО":GOTO 2000
2060 PRINT "МОЛОДЕЦ"
```

Здесь озадачивают и уровень организации диалога, и способ анализа ответа. Представим, что ученик, например, ответил не $100 \cdot X$, а $100 \cdot x$, или $100X$, или $X \cdot 100$, или $100 \cdot X$ (последний вариант кажется верным? Нет, X русское)... Программа не предоставляет справочной информации, может довести до белого каления повторением одного вопроса, не воспринимает как правильные 90 % правильных ответов... Создается впечатление, что публикуемые в журнале статьи о том, каким должно быть педагогическое программное средство, не читаются. Или они непонятны? Или их мало?

Вот еще один пример того, как не надо писать ПСС.

«Для проведения тестированного контроля знаний вам поможет специально созданная программа «Программированный опрос». Она написана на языке «Бейсик-БК» и может быть легко адаптирована на любые другие языки и КУВТы. Главное ее преимущество в том, что содержание подготовленных для опроса билетов может быть любым и совершенно не зависит от того, по какому предмету проводится опрос.»

Чрезмерная универсализация, как и следовало ожидать, оборачивается примитивностью функций. Вот образец заготовки билета:

```
100 PRINT "БИЛЕТ N"
110 PRINT "ВОПРОС N"
120 PRINT 'ТЕКСТ ВОПРОСА
130 PRINT "ВАРИАНТЫ ОТВЕТОВ"
140 PRINT "1)..."
150 PRINT "2)..."
160 PRINT "3)..."
170 PRINT "УКАЖИТЕ НОМЕР ПРАВИЛЬНОГО
                                ОТВЕТА"
```

Таким образом, простейшую контролирующую программу с множественным выбором (выбором правильного ответа из числа предлагаемых вариантов) автор считает пригодной для использования в качестве педагогического программного средства. Но это было верно только на самой заре компьютеризации образования, когда не существовало других программ! Быть может, журналу следует опубликовать обзор используемых (и уже не используемых) при построении ППС приемов?

В заключение, как обычно, письма без комментариев и доска объявлений.

«На мой взгляд, было бы неплохо открыть рубрику «Советы начинающим» и не замыкаться на БК-0010. Ведь уже есть в продаже «Партнер», «Криста» и другие, доступные по цене. Очень не хватает консультации. Пусть она будет платной, но быстрой и исполнительной.»

А. В. Иванов, Петропавловск-Камчатский.
«Хотелось бы прочитать о возможностях использования микро-ЭВМ в качестве программируемого устройства управления.»

П. В. Федоряка, Полтавская обл.

«С нашим КУВТ УКНЦ та же история: четыре машины давно в ремонте, еще две вышли из строя, но на ремонтные предприятия пока не попали. Ставропольский участок Георгиевского завода по техническому обеспечению и ремонту ВТ «Союзсчеттехника» сделал ремонт не может из-за отсутствия запчастей. А как же гарантийные обязательства завода-поставщика? Наш «родной» РЦ ВТИ в г. Майкопе вообще отказался от обслуживания КУВТ. И что же теперь делать? Может быть, завод «Квант» пришлет «гарантированный» ЗИП непосредственно в училище? Это было бы лучше, чем ничего.»

Ю. С. Сергеев, Майкоп, Адыгейское педучилище.

«Выпускаемые нашей промышленностью КУВТы непригодны для работы в условиях морского климата, в особенности с содержанием в воздухе малейших примесей сероводорода. Предлагаю заводам-изготовителям помещений и другие необходимые условия для организации испытательной лаборатории у нас в условиях влажного морского субтропического климата. Установите у нас КУВТ на один год, а мы будем проводить типовые испытания по заводским данным. Польза от этого будет большая.»

В. А. Селуянов, г. Сочи.

«Мы стали обладателями очень дорогой (по цене) учебно-контролирующей программы для КУВТ-86. Система «Б1», где на обложке значится «Министерство народного образования РСФСР», очень сильно характеризует состояние народного образования, мероприятия по ликвидации компьютерной безграмотности учащихся. Описание составлено безобразно, оно расходится с программами, записанными на дискетах. Там нет текстов описаний, например, README.LST V1REKL.LST. Запуск одной из программ дает ошибки, она не идет. В пакете ошибки русского текста, примитивное оформление заставок. Где здесь ОТК, где госприемка? Очень дорого — 400 рублей за 5 дискет. Автор — НПК «Энтузиаст.»

Н. А. Какинин, г. Скопин, СШ № 4.

«Известна большая стоимость ВТ, поэтому считаю, что отдача кабинета информатики должна быть большей. Содержать его ради двух часов информатики и одного часа факультатива — безнравственно.

Для решения проблемы компьютерной грамотности необходима большая внеклассная работа (естественно, оплачиваемая). К примеру, можно писать учебные программы, оказывать помощь учителям других предметов в проведении уроков в кабинете. База для этого есть. Ученики, мои помощники, подготовлены к работе — но сделать это мне удалось, лишь благодаря жесткому режиму работы в прошлом учебном году (бесплатно, во внеурочное время, в ущерб семье, причем ежедневно).»

Н. Н. Дружинин, с. Яркое, Крымская обл.

«Из 13 ПЭВМ «Агат» три были неисправны сразу; присланные из Казани дискеты с прикладными программами читаются только на НГМД 5088.02, а такие стоят только на двух ПЭВМ, на остальных — НГМД 5323.01, причем магнитофонные входы на этих машинах не работают. Кто может нам помочь, ответьте, пожалуйста! Мой адрес: 613304 Кировская обл., с. Коршик, школа, *В. В. Кочкину.*»

Получаем мы и такие письма:

«Мне бы хотелось, чтоб вы приобрели новые рубрики в журнале о «НЛЮ» и если их может замет. комп.»

П. Б.

«В нашей школе никто не собирается закупать дискеты с играми, и поэтому прошу вас выслать мне программу какой-либо игры для БК-0010, язык программирования Бейсик.»

Это просьба всего нашего класса!

Ждем ответа. 10 «Б.»

Пенза

ДОСКА ОБЪЯВЛЕНИЙ

Ищу программы для БК «Львов ПК-01», хочу переписываться с владельцами этого компьютера.

684900 Камчатская обл., Пенжинский р-н, с. Каменское, ул. Бекерова, 24, кв. 8. Валерий Борисович МОЗУР.

Всех обладателей КУВТ УКНЦ, желающих обмениваться программами, просим писать по адресу:

Ижевск, ул. Союзная, 25, СШ № 54, клуб «БЕЙС».

Ищем программы, в первую очередь педагогические, для ПЭВМ «Искра-1256».

643050 Северо-Казахстанская обл., с. Пресновка, СПТУ № 17.

Старые компоненты — новый результат

Симбиоз компьютера и телефона подарил миру телефакс. Прошло совсем немного времени, и симбиоз компьютера, правда, более мощного, и неизменного телефона породил новое детище — фотофон, который так же не похож на телефакс, как черно-белая фотография не похожа на цветную широкоформатный стереофильм.

Принципы работы фотофона ничем не отличаются от принципов работы телефакса: оцифровывание и кодирование информации, передача ее по относительно недорогому телефонному каналу, дешифрация и печать информации на другом конце провода. Отличие лишь в скорости и объемах пересылаемой информации, которые позволяют передавать трехмерное цветное изображение оригинала в реальном масштабе времени с фотографическим качеством, что практически превращает фотофон в цифровой видеотелефон.

Имея такой инструмент, вы сможете не только участвовать в цифровых телесовещаниях, но и работать над одним проектом с коллегой из страны, получать и передавать с высоким качеством большие массивы информации.

Интеллектуальная клавиатура

Никколо Паганини был настолько талантливым музыкантом, что мог на единственной струне исполнить сложную скрипичную партию. И хотя с тех пор этот инструмент почти не изменился, превзойти мастера не смог никто.

У наших современников появились новые сферы деятельности. Одна из них — вычислительная техника. Разумеется, вряд ли при

работе на современных компьютерах возникнет необходимость управляться с помощью единственной или клавиши. Но и нескольких десятков часто не хватает электронным виртуозам. Поэтому нередко каждая клавиша несет тройную или даже большую смысловую нагрузку, обозначая не только латинскую и русскую буквы, но и ключевые слова, команды и даже целые фразы какого-либо языка программирования.

Гибкая многофункциональность клавиатуры обеспечивается обычно специальными программами, такими, как Суперкей или Прокей, которые «запоминают» новые функции клавиш, обеспечивают их изменение и использование. Но у всех подобных программ есть недостаток: они потребляют столь дорогие ресурсы компьютера, как системная память и время работы центрального процессора.

Новая клавиатура, подключаемая к IBM PC — совместимым персоналкам, обеспечивает максимальный сервис без использования ресурсов компьютера. Еще бы — ведь в нее встроены специализированный микропроцессор, поддерживающий программирование 400 макроклавиш и не требующий специального программного обеспечения.

Если бы столь энергичный прогресс произошел и в области изготовления смычковых инструментов, то одно прикосновение к скрипке вызвало бы звучание целого аккорда.

Еще один миллионер

Речь идет о последней новинке американской фирмы Intel — микропроцессоре 80486. В нем 1 млн. 200 тыс. транзисторов. Тактовая частота первой модели — 25 МГц, следующей — 40 МГц. В нем немало новинок: конвейерная обработка команд (благодаря чему его быстродействие приблизится к быстродействию RISC-процессоров), операции с плавающей точкой (для предыдущих моделей эти операции выполняли сопроцессоры), встроенная кэш-память. Цена — около 950 долларов при продаже партиями по 1000 штук.

Первой компьютер на этом микропроцессоре выпустила английская фирма Argosoft; это VX FT Server. IBM пока предлагает на основе новинки только плату для усовершенствования одной из своих старых моделей ПЭВМ.

60 миллиардов операций в секунду

может выполнять компьютер NCUBE-2 фирмы NCUBE (США). По-видимому, сегодня это самый быстродействующий компьютер в мире. Его архитектура весьма специфична: 8192 64-разрядных микропроцессора, связанных в единую сеть, моделью для которой послужил многомерный куб, в вершинах которого расположены микропроцессоры, а ребрами служат линии связи. Каждый микропроцессор связан с 13 соседями и выполняет часть вычислений, внося свой вклад в общую «мозаику» задачи.

«Все мы хотим одного — избежать войны», —

говорит Г. Майер-Кресс, физик по образованию и немец по происхождению, занимающийся созданием компьютерных моделей в Лос-Аламосской национальной лаборатории, одном из известнейших военно-научных учреждений США. Его имя получило известность в связи с новым подходом к анализу возможных последствий гонки вооружений. Г. Майер-Кресс, используя в своих моделях нелинейные элементы, показал, что даже незначительные изменения начальных условий, например добавление лишь одного нового вида противоракетного оружия в моделируемую систему национальной обороны США, способны вызвать лавинообразный процесс дестабилизации международных отношений вплоть до ядерной войны. Американские военные серьезно отнеслись к его работам; их намерены учитывать и использовать в ВМС США, разведывательном управлении Министерства обороны и других ведомствах.

Стоит отметить, что ученый является пацифистом со стажем. На родине, в ФРГ, он голосовал за «зеленых» и участвовал в демонстрациях протеста против размещения американских ядерных ракет. Однако это не стало препятствием для его приглашения в Америку, где он регулярно, в годовщину бомбардировки Хиросимы, устраивал акции протеста против ядерного оружия. Но создание модели гонки вооружений стало, видимо, его самым крупным вкладом в дело мира.

Рождение сегодня множества разнообразных научно-методических советов, объединений, комиссий, подразделений при организациях и предприятиях и т. д., занимающихся вопросами внедрения компьютеров в систему образования, отражает растущий общественный интерес к этой проблеме. Многообразие задач и подходов к их решению открывают перед специалистами огромный фронт работ. Общее дело информатизации требует взаимной информированности. Редакция предлагает всем заинтересованным органам заявить о себе, прислав свое положение или иной документ, отражающий задачи, функции и проблематику проводимых работ. Это поможет объединить усилия и избежать дублирования исследований.

Пленум УМО по информатике

В феврале 1990 г. в Коломенском педагогическом институте состоялся организационный пленум учебно-методического объединения «Информатика в общеобразовательной школе». Созданное при МОПИ им. Н. К. Крупской, новое объединение ставит целью интегрировать усилия школы, средних учебных заведений всех типов и вузов по реализации принципа непрерывности образования в условиях разветвляющейся компьютеризации и информатизации общества. В принятом на пленуме положении УМО определяется как совещательный орган, призванный способствовать внедрению новых информационных технологий в преподавание школьных предметов в тесной связи с курсом основ информатики. Функции УМО включают определение приоритетных направлений и тематики работ в области информатизации, рассмотрение рукописей новых учебных пособий по применению компьютеров в школе, обобщение и пропаганду передового педагогического опыта, повышение квалификации учителей в пединститутах и ИУУ, подготовку новых кадров учителей, специалистов по информатике и компьютерному обучению и ряд других.

Два дня работы пленума были заполнены обсуждением острых проблем выбора дальнейших путей развития школьной информатики и внедрения в школьное обра-

зование новых технологий обучения, ориентированных на применение ЭВМ. Серьезную озабоченность дальнейшей судьбой курса ОИВТ выразили в своих выступлениях М. Г. Ратинский (МОИУУ), И. Е. Подчиненов (Свердловский пединститут). Решение Гособразования СССР об объединении математики и информатики в старших классах (приказ № 751) в сегодняшних условиях препятствует решению основных задач школьной информатики как средства интегрирования образования и его гуманизации на основе широких межпредметных связей и внедрения новых информационных технологий обучения. В этой связи Ю. А. Первин (ИПС АН СССР, г. Переславль-Залесский) говорил о целесообразности постановки в начальной школе нового современного компьютеризованного курса «Математика — информатика — родной язык».

Для ускорения создания и внедрения новых ППС и методик обучения, ориентированных на применение ЭВМ, УМО следует продумать способы поощрения разработчиков, подчеркнул В. В. Анисимов (НИИ школ МНО РСФСР). Пленум УМО обратился к Гособразованию СССР с просьбой рассмотреть вопросы, связанные с защитой авторских прав педагогов и программистов, создающих новые ППС.

В выступлении Е. С. Овчинникова (педагогическое училище № 1, Москва) была поставлена проблема подготовки учителей начальной школы со специализацией в области информатики. Широкою программой подготовки экспериментальных программ по информатике для старших классов разных профилей, а также средних классов общеобразовательной школы предложил в своем выступлении А. Р. Кангро (Латв. ГУ). В. П. Недошин (МОПИ. им. Н. К. Крупской) рекомендовал пленуму изыскать возможности для оформления УМО в качестве общественной организации с правами юридического лица. Это позволит наладить прочные деловые связи по самым разным направ-

лениям работ и создавать при УМО комитеты или отделения, объединяющие усилия педагогов по родственным дисциплинам.

С информацией о новой разработке — учебной персональной ЭВМ «Элекс-1» — выступил В. А. Корнев (НПК «Компьютер», Москва).

Л. Е. Самовольнова (Гособразование СССР) поставила перед УМО задачу глубокой, всесторонней проработки комплексной целевой программы компьютеризации образования и выразила надежду найти в новом объединении надежного помощника, имеющего тесные связи с практикой применения компьютеров на всех ступенях образования.

Быть ли единой системе?

115

В феврале 1990 г. в Москве состоялось всесоюзное совещание, посвященное созданию Единой государственной системы информатизации образования (ЕГСИНО). В совещании, организованном Гособразованием СССР и Московским энергетическим институтом, участвовало 160 человек из 18 городов страны.

В обзорном докладе по проблеме создания ЕГСИНО СССР лауреат Государственной премии СССР канд. техн. наук К. Победоносцев отметил, что, несмотря на значительное отставание СССР в области компьютеризации, имеются реальные предпосылки для ускорения информатизации общества через информатизацию народного образования. Система ЕГСИНО имеет большое гуманитарное значение, так как обеспечит возможность создания и наращивания информационных ресурсов народного образования и обмена ими между самыми удаленными пунктами Советского Союза. Для создания ЕГСИНО СССР необходимо объединение средств и усилий центральных и республиканских органов народного образования, а также местных Советов народных депутатов.

Об основных принципах создания ЕГСИНО рассказал доктор техн. наук С. Чернышев, об архитектуре системы — доктор техн. наук Н. Губонин. Социально-педагогическим проблемам ЕГСИНО было посвящено выступление доктора техн. наук В. Взятыхшева, а также «круглый стол», проведенный в рамках совещания.

Проблеме построения банков данных и знаний в ЕГСИНО был посвящен доклад доктора техн. наук И. Башмакова. Перед присутствующими была раскрыта перспекти-

ва оперативной передачи и получения информации (по спутниковым каналам) о наличии тех или иных программных средств, конкретные автоматизированные учебные курсы будут пока передаваться, видимо, наземными средствами. В ходе ответов на вопросы докладчик проинформировал собравшихся об имеющихся в СССР интересных, уникальных в своем роде банках данных, к которым проявится интерес и за рубежом. Таким образом, система будет ориентирована на взаимовыгодный и полезный в научном отношении межрегиональный и межгосударственный обмен. Причем банки данных будут включать и банки программ, и банки технологий. Здесь существенную помощь может оказать недавно созданный центр новых информационных технологий при НИИ ВШ.

Об особенностях построения и функционирования информационных сетей ЕГСИНО говорилось в докладе канд. техн. наук Л. Абросимова и В. Мошкина. Было отмечено, что проект системы находится на первом уровне проработки и что при условии финансирования аванпроект может быть выдан уже в 1990 г.

Количество абонентов информационных сетей в области образования по предварительной прикидке равняется 10^6 . Сюда входят пользователи от детского сада и школы до вузов и НИИ. Источниками информации будут служить уже действующие системы поиска, сбора информации, существующие в стране.

Таким образом должен быть осуществлен принцип максимального использования существующих ресурсов и приобретения перспективных средств. В полной мере бу-

дуг учитываться международные стандарты.

Лауреат Государственной премии СССР доктор техн. наук В. Кулешов в своем докладе рассмотрел принципы построения «функциональных» подсистем на примере подсистем «Вуз» и «Школа». Для участников совещания была организована демонстрация инструментальных и технических средств подсистемы «Школа» и фрагментов региональной сети «Вуз» — «Школа».

Во время «круглого стола» по экономическим и коммерческим аспектам создания ЕГСИНО обсуждались, в частности, проблемы интеллектуальной собственности разработчиков программного обеспечения. Оживленно обсуждалась и парадоксальная ситуация в советской школе, с успешным развитием которой связывают будущее об-

щества и которая тем не менее влечит нищенское существование.

В рамках совещания была организована экскурсия в Центр управления полетами космических аппаратов «Медвежьего озера», где присутствующим было показано, как сочетаются средства связи отдельного региона и общесоюзная среда связи.

По результатам совещания принят заключительный документ, в котором, в частности, Гособразованию СССР рекомендуется представить в Совет Министров СССР проект постановления о Единой государственной системе информатизации народного образования СССР.

Наш журнал будет информировать читателей о ходе утверждения и выполнения этого проекта.

Третья Ленинградская научно-методическая конференция

В декабре 1989 г. состоялась третья Ленинградская научно-методическая конференция «Проблемы образования в области информатики, вычислительной техники и автоматизации», организованная Ленинградским областным правлением Союза научных и инженерных обществ СССР и Ленинградским отделением Всесоюзного общества информатики и вычислительной техники. По представительству, широте обсуждаемых проблем и глубине проработки отдельных вопросов конференция вполне могла бы считаться всесоюзной.

Пленарное заседание конференции проходило в Выборгском Дворце культуры и техники. Открыл работу конференции профессор С. Лавров, член-корреспондент АН СССР, докладом «Концепция информатизации общества». Подробно осветив три аспекта информатизации общества: политический, социальный и экономический, — он остановился на трудностях сегодняшнего этапа, подчеркнул важность грамотного использования вычислительной техники пользователями-профессионалами. Профессор Б. Советов в докладе «Методологические основы информационной технологии» остановился на вопросах информатизации всех сторон жизни общества, особое внимание уделив информационным технологиям в образовании. Практическим аспектам информатизации образования был посвящен доклад А. Кушниренко и Г. Лебедева «Начальный курс программирования на механико-математическом факультете МГУ». Читатели журнала знакомы с их

концепцией, многие используют Е-практикум в различных модификациях. Решение типовых вычислительных задач на русском алгоритмическом языке позволяет, по мнению авторов, создать фундаментальную основу для освоения в дальнейшем любых языков программирования и сервисных средств. Профессор М. Игнатъев в докладе «Информатика в экологии, здравоохранении и охране исторических памятников» раскрыл возможности применения информационных средств в этих нетрадиционных областях, остановился на возможностях привлечения школьников к решению такого рода задач.

Затем начались заседания в трех секциях: «Информатика в средних учебных заведениях»; «Программные и технические средства обучения»; «Информатика в высшем образовании и повышении квалификации».

Весьма интенсивно работала под руководством А. Матюшкина-Герке первая секция. В докладах рассматривались такие основные проблемы: опыт преподавания основного курса; содержание и организация внеклассной работы; педагогические эксперименты; проблемы подготовки и переподготовки учителей.

Все 25 докладов секции представляли интересные направления и опыт работы авторских коллективов. Так, Р. Имедашвили рассказал об организации работы Пушкинского районного центра информатики и творчества школьников. Силами этого центра обеспечивается как проведение занятий по основному курсу во всех школах района,

так и экспериментальное преподавание, различные виды внеклассной работы. Пакет учебных программ, продемонстрированный А. Гриценко на КУВТ-86, еще раз показал богатейшие возможности, заложенные этот столь широко распространенный комплекс. С точки зрения программно-информационного обеспечения КУВТ-86 существенно сближается с КУВТ «Ямаха». Г. Фролова сделала интересное сообщение об организации индивидуальной работы с учащимися, Н. Лунина — о факультативном курсе программирования. А. Матюшкин-Герке, подводя итоги четырехлетней работы по постановке курса ОИВТ в Ленинграде и области, в качестве основной выделил проблему непрерывной подготовки учителей.

Общие итоги работы секции таковы: нужно признать, что общепризнанной концепции информатизации образования на сегодняшний день не существует, хотя и есть попытки волевым порядком утвердить ту или иную концепцию; подобно тому как в технических вузах сохраняется курс математики, что обеспечивает методологическое единство при использовании ее средств и методов в других дисциплинах, должен быть сохранен в средней школе и общий курс информатики. Попытки ликвидировать этот курс некоторые докладчики расценивали как стремление уйти от ответственности за ошибки, допущенные при его введении; существующие учебные программы и учебники не обеспечивают нормального прохождения курса; необходима фундаментальная учеба учителей информатики; проектирование, создание, тиражирование и сопровождение ППИС осуществляется стихийно, что во многом осложняет практическую работу учителей; разнообразие используемых в учебном процессе типов ЭВМ будет еще долгое время оставаться реальностью. Недопустимо ориентировать школу на применение ПЭВМ типа IBM PC, ибо в ближайшие годы этой техники подавляющее большинство школ не получит, а в более дальней перспективе такие компьютеры попросту устареют.

Вторая секция под руководством профессора В. Ефремова и доцента Е. Павловского рассматривала практические аспекты разработки программных средств и технических средств обучения, преимущественно для высших учебных заведений. Докладчики рассказывали об опыте использования вычислительной техники различных типов при постановке курсов общинженерного и общетехнического профиля, разработке учебных САПР. При этом отмечают основные преимущества внедрения компьютеров: индивидуализация обучения,

активизация обучения, повышение мотивации к овладению профессиональными знаниями.

Разработки Д. Варсонофьева, А. Дымченко, В. Мельниченко, Г. Таяновского и других докладчиков заинтересовали бы многие родственные вузы, но обмен информацией об имеющихся программных средствах у нас пока не налажен, и это приводит как к дублированию разработок, так и к потере ценного опыта.

Работа третьей секции проходила под руководством профессоров Б. Советова и И. Румянцева. Здесь наряду с опытом постановки конкретных курсов с учетом специфики вузов рассматривались и общие, концептуальные проблемы внедрения информационных технологий в учебный процесс. В выступлениях звучали самые неожиданные предложения и оценки. Так, В. Фомичев из Минска предложил передать обучение программированию профилирующим специальным кафедрам технических вузов, М. Лившиц поделился опытом обучения информатике будущих экономистов. Но общие положения в принципе совпадали: основой курса является алгоритмизация, большое значение имеет практическое решение задач за дисплеем, при этом на начальном этапе обучения целесообразно использовать русскоязычные сервисные средства. Наиболее полно эти вопросы были обобщены в докладах В. Каймина и О. Козлова. Интерес многих присутствующих вызвали доклады А. Шлеймера «Информационная модель психолого-педагогических измерений» (здесь было предложено измерять знания обучаемых с помощью компьютеров) и В. Лукашова «Игровой вариант обучающей программы для ЭВМ по курсу марксистско-ленинской философии», где было раскрыто еще одно нетрадиционное применение компьютеров.

Тезисы конференции будут опубликованы.

В завершение работы конференции была организована демонстрация ППИС как в школьных центрах, так и на базе кафедры информатики ЛГПИ им. Герцена. Подобные демонстрации всегда оживляют работу и способствуют зарождению новых деловых контактов, ведь хорошие программы привлекают внимание и понятны без лишних слов.

Заключительное пленарное заседание подвело итоги конференции. Принимая во внимание значимость обсуждаемой тематики и накопленный организационный опыт, решено впредь проводить подобные конференции ежегодно.

О. КОЗЛОВ

Всесоюзная конференция «Компьютер в школе и педвузе»

С 9 по 15 сентября 1989 г. в г. Одессе проходила Всесоюзная конференция «Компьютер в школе и педвузе». Организаторы конференции — НИИ ИВТ АПН СССР (г. Новосибирск) и ОГПИ им. К. Д. Ушинского (г. Одесса). Конференция вызвала большой интерес среди преподавателей педвузов и школ, сотрудников НИИ. Было прислано свыше 300 докладов и сообщений из различных регионов страны: Прибалтики, Белоруссии, Украины, Молдавии, РСФСР, Средней Азии.

Оргкомитет отобрал около 100 докладов, соответствующих профилю и уровню конференции.

На пленарных заседаниях было заслушано и обсуждено шесть докладов. Авторы учебников для средней школы В. Каймин и А. Кушниренко (Москва) рассказали о своих концепциях школьного курса информатики. Активно обсуждался доклад А. Козырева (Москва) «О проблемах рынка программного обеспечения в СССР, авторских и лицензионных договорах и ценах».

Принципам построения учебного плана и рабочих программ для специальности «Учитель информатики и ВТ, специалист по НИТО» был посвящен доклад Ю. Кузнецова и И. Румянцева (Ленинград).

О новых информационных технологиях в подготовке и повышении квалификации педагогических кадров рассказал М. Лапчик (г. Омск).

Перспективам информатизации школьного образования посвятил доклад В. Ефимов (Москва).

Работало пять секций:

1. Использование новых информационных технологий обучения в учебном процессе школы и вуза — рук. И. Румянцев (Ленинград) и А. Слуднов (г. Новосибирск).

2. Инструментальные средства разработки компьютерных средств обучения — рук. А. Кушниренко (Москва) и А. Слуднов (г. Новосибирск).

3. Программное и методическое обеспечение курсов информатики в школе и вузе — рук. В. Каймин (Москва) и А. Павловский (Минск).

4. Программное и методическое обеспечение естественнонаучных курсов в школе и вузе — рук. Г. Бритавский (г. Одесса) и М. Лапчик (г. Омск).

5. Программное и методическое обеспечение гуманитарных курсов в школе и ву-

зе — рук. Ю. Кузнецов (Ленинград) и Ю. Полушкин (г. Омск).

Ежедневно проводилась демонстрация программного обеспечения.

Участники конференции обсудили актуальные вопросы и тенденции развития информатики в образовании. В результате обобщения высказанных мнений и предложений была дана критическая оценка положения дел в области информатизации и сформулированы конкретные рекомендации:

1. Оснастить педагогические учебные заведения, готовящие учителей информатики и вычислительной техники, самой современной вычислительной техникой, отвечающей требованиям эргономики и обеспечивающей интеграцию советской средней и высшей школы в мировую систему образования.

2. Учитывая, что старый механизм финансирования производства и тиражирования программных средств полностью изжил себя и совершенно не работает в условиях перехода народного хозяйства на новые методы (хозрасчет, самокупаемость и т. д.), необходимо:

увеличить объем финансирования разработок программных средств до уровня затрат на приобретение средств вычислительной техники;

выделенные средства распределить по нескольким направлениям: на проведение опережающих разработок, на финансирование хорошо зарекомендовавших себя коллективов разработчиков, а основную часть средств передать для приобретения программных средств потребителям и ассоциациям потребителей.

3. Исходя из опыта ряда педвузов, считать целесообразным введение нового курса «Теоретические основы информатики» или переориентацию в этом направлении традиционного курса «Математическая логика и теория алгоритмов».

4. Рекомендовать Госкомитету по народному образованию СССР организовать выпуск ежегодного тематического сборника трудов кафедр информатики и вычислительной техники педагогических вузов страны на базе Ленинградского ГПИ им. А. И. Герцена.

5. Поручить проректору Омского ГПИ им. А. М. Горького проф. М. Лапчику организацию и подготовку целевого номера журнала «Информатика и образование» по теме «Информатика в педвузе». Просить

главного редактора журнала академика В. А. Мельникова содействовать скорейшему выпуску этого номера.

6. Обратиться в Госкомитет по народному образованию СССР с предложением сделать Всесоюзную конференцию «Компьютер

в школе и педвузе» ежегодной и проводить ее через год — в Одессе и в Новосибирске.

Материалы конференции готовятся к публикации.

Г. БРИТАВСКИЙ,
М. КОНСТАНТИНОВСКИЙ
г. Одесса

Международный конгресс «Образование и информатика»

В апреле 1989 г. в Париже в штаб-квартире ЮНЕСКО состоялся первый международный конгресс «Образование и информатика». Участники конгресса приняли декларацию, в которой подчеркнули острую необходимость повсеместно готовить квалифицированные кадры, способные использовать информационные технологии. Участники конгресса пришли к выводу, что информатика, в силу ее значимости для управления образованием, не должна стать дисциплиной, монополизированной самоназначаемыми группами, а являться частью знаний, предлагаемых в пользование всем гражданам. Это повысит ответственность систем образования и подготовки кадров, независимо от того, являются ли они формальными или неформальными, государственными или частными.

Участники конгресса, заинтересованные в будущем образования, развитии культуры, призвали компетентные национальные и международные органы содействовать тому, чтобы в каждой стране осуществлялись меры по:

всеобщему распространению информатики в области управления образованием не только на уровне администрации, но и на уровне доступа к библиографическим и документальным источникам, а также в организации и управлении процессом обучения (индивидуализация) и оценки (экзамены и конкурсы);

развитию нововведений, экспериментов и педагогических исследований в школьных учреждениях и вне их по выявлению областей, в которых информатика наиболее способна совершенствовать образование и, в первую очередь, повысить качество обучения;

повышению уровня культуры и технологической подготовки всех работников, занятых в сфере образования в каждой стра-

не, независимо от того, заняты ли там начальным образованием или переподготовкой кадров, чтобы они могли мобилизовать таланты, необходимые для оптимального использования информатики в различных областях образования, где такое использование необходимо и оправдано потребностями;

осуществлению необходимых исследований для определения методов и средств, способных обеспечить дополняемость, модульность, совместимость программного обеспечения, оборудования, систем и сетей электронной коммуникации, используемых в образовании и подготовке кадров.

Участники заявили о своем намерении сделать все возможное для расширения международного сотрудничества, начиная с активизации обмена информацией и облегчения передачи знаний из страны в страну; участия в инициативах, направленных на поиск решений в области использования компьютеров в образовании и подготовке кадров, которые были бы общими для стран или групп стран с одинаковым уровнем развития; предоставления приоритета прогрессивным формам применения информатики в образовании и подготовке кадров, которые способны содействовать охране национальных языков и внедряться в национальные культуры, не оказывая глубокого воздействия на их дух и специфические ценности.

Участники призвали правительства, отдельных лиц, коллективы предпринять новые усилия для согласования (если это необходимо) действий в этой области и поступать таким образом, чтобы международное сообщество предоставляло международным проектам и программам сотрудничества в области информатики для образования приоритет, чтобы охватить все страны, испытывающие в этом потребность.

Проект ИТЕС*: Международное лонгитюдное (длительное) исследование влияния информационных технологий на развитие детей

Начиная с 1987 г., при поддержке ЮНЕСКО, Академии медицинских наук Болгарии и других международных организаций, ученые из 18 стран мира ведут изучение возможностей использования новых информационных технологий в образовании. Цель этого исследования — оценить влияние использования компьютеров в школьном обучении на познавательное развитие детей. Концептуальной основой проводимого исследования является культурно-историческая концепция советского психолога Л. С. Выготского, рассматривающая развитие высших психических функций ребенка как результат социально-исторического опыта. В соответствии с этой концепцией предполагается, что изменение в развитии высших психических функций, которое наблюдается у детей в результате применения компьютеров в обучении, есть функция вполне определенных социальных влияний, включающих культуру, тип и часто-

* Information Technology. Education. Children

ту использования компьютеров, социальную организацию и виды деятельности, которые определяют это использование, а также своеобразие используемых программных средств.

Особый интерес представляет для исследователей оценка влияния применения компьютеров в различных культурных районах. Данный аспект важен с точки зрения возможностей изменения уровня образования в различных странах.

Согласно программе проекта в исследовании в 1991 г. примут участие дети III—IV классов (9—10 лет). Планируется включить в эксперимент по крайней мере 4 класса в каждой стране, обеспечить наблюдение за каждым ребенком в течение 3 лет (1991—1993 гг.). Проверка развития высших психических функций у детей будет проводиться по специально разработанной системе диагностического обследования до использования компьютеров в обучении и после такого использования, а также в конце каждого года обучения.

Результаты наблюдения за детьми, точка зрения участвующих в проекте учителей, а также общие методологические и теоретические выводы о содержании проводимых исследований будут опубликованы в специальном издании и станут достоянием широкого круга специалистов.

Конференция в г. Ульяновске

27—28 ноября 1989 г. в г. Ульяновске состоялась межвузовская конференция «Лингвистические и компьютерные основы оптимизации обучения иностранным языкам в педагогических институтах», в которой приняли участие преподаватели вузов страны.

На конференции работали три секции: «Общие вопросы компьютеризации преподавания иностранных языков», «Обучение различным аспектам языка с помощью обучающих лингвистических автоматов», «Лингвостатистические основы компьютеризации преподавания иностранных языков». Было заслушано 46 докладов.

В докладе Р. Пиотровского и автора этих строк «Об обучающем лингвистическом автомате» были определены операции, которые должен выполнять такой автомат в пакетном, а чаще в диалоговом режиме: готовить лингвистический первопродукт (частотные, алфавитные, обратные словники, списки машинных оборотов и т. п.); осуществ-

лять обучение родному или иностранному языку, контролировать ход обучения; производить не только пословно-пооборотный, но и лексико-грамматический (связный) перевод текстов разной тематики с одного языка на другой; аннотировать и семантически рубрицировать (индексировать) русский или иностранный текст, а также выполнять функции экспертной системы; выявлять и устранять искажения текста. Была также раскрыта стратегия построения программного, лингвистического и лингводидактического обеспечения обучающего лингвистического автомата.

Доклад П. Алексеева «Оппозиции в квантитативной лингвистике» был посвящен анализу квантитативно-лингвистических оппозиций как парадоксов инженерной лингвистики, которые используются и учитываются при построении информационных и обучающих лингвистических автоматов. Были рассмотрены три класса оппозиций: общеметодологические, общелингвистические и линг-

востатистические.

Э. Королев представил доклад «Лингвистические процессоры в автоматизированных обучающих системах», в котором он изложил основные принципы построения лингвистического процессора КАСКАД (класс автоматизированных систем комплексного анализа документации) и его подсистем, реализованного на ПЭВМ, совместимых с РС XT и использующего текстовые данные, обрабатываемые с помощью СУБД типа dBASE-3.

В докладе Л. Пономаренко «Прогнозирование эффективности занятий языком при помощи ЭВМ» был представлен разработанный автором алгоритм выявления способности абитуриента к изучению иностранных языков. Алгоритм ориентирован на АОС СОКРАТ. Он может быть применен также для индивидуализации самостоятельной работы студента и для объективной оценки работы преподавателя.

На секционных заседаниях докладчики информировали своих коллег об использовании ЭВМ при обучении различным аспектам языка.

О структуре спецкурса по основам компьютеризации для преподавателей вузов и учителей иностранных языков, направленного на ликвидацию их компьютерной безграмотности, говорилось в докладе С. Агаповой (г. Ростов-на-Дону).

Опытом использования в учебном процессе системы информационно-логической обработки документов поделились Х. Арзикулов, И. Умаров, Ш. Юсупова (г. Самарканд). О компьютеризированном самоучителе иностранного языка рассказал М. Рыбаков (Москва).

Часть докладов была посвящена применению ЭВМ в преподавании иностранных языков на неязыковых факультетах пединститутов и в технических вузах.

Большой интерес вызвали сообщения, в которых рассматривались актуальные проблемы компьютеризации уроков иностранного языка в средней школе — М. Иванова (г. Абакан), Л. Борисенкова, Л. Ньюбина (г. Смоленск), лингвометодические принципы создания педагогических программных средств по английскому языку для IV—V классов — А. Гурьяшкина, В. Диваев (г. Ульяновск).

Многие участники конференции привезли обучающие программы, которые были с успехом продемонстрированы в компьютерном классе инфака и в лаборатории ЭВМ института.

Успешное решение проблем компьютеризации преподавания иностранных языков предполагает оптимальный отбор лексических и грамматических единиц на основе статистико-вероятностного анализа лингвистического материала. С докладами на эту тему выступили: Р. Крылова, И. Мацукова, В. Скиценко (г. Ульяновск), Т. Иванова (г. Чебоксары), П. Кадырбекова (Москва) и др.

В целом на факультетах иностранных языков пединститутов страны увеличилось число компьютеров и компьютерных классов; возрос интерес преподавателей-филологов к использованию ЭВМ в аудиторной и внеаудиторной работе; повысилась их компьютерная грамотность, чему способствовала организация двухмесячных курсов по информатике; увеличилось количество научно-методических работ, направленных на создание лингвистических, лингводидактических и технико-информационных основ преподавания иностранных языков в вузе и в школе; усилилось сотрудничество программистов и преподавателей иностранных языков; повысился КПД этого сотрудничества.

Вместе с тем участники конференции отметили, что еще мало работ, в которых освещались бы психологические, педагогические, дидактические аспекты использования ЭВМ в учебном процессе, а также медицинские и другие вопросы, неизбежно возникающие при работе преподавателя и студента с обучающим автоматом. Указывалось также на слабую защищенность авторских прав разработчиков учебных программ. Говорилось и о трудоемкости составления программ, что, однако, не находит отражения в документах, регламентирующих учебную нагрузку преподавателей.

Конференция приняла ряд решений, в том числе: провести в 1992 г. на базе Ульяновских педагогического и политехнического институтов межвузовскую школу-семинар по использованию компьютеров в обучении иностранному языку в вузе, а в 1993 г. на базе этих же институтов — республиканскую конференцию по использованию компьютеров в преподавании общественно-политических и гуманитарных наук.

В свободное время участники конференции посетили ленинские места г. Ульяновска.

Проректор по научной работе
Ульяновского пединститута
В. БОГОДИСТ

Конкурс школьных химических программ на ЭВМ

В дни школьных каникул, с 8 по 10 января 1990 г., на базе Новосибирского государственного университета состоялся конкурс Сибирского отделения АН СССР на лучший комплект школьных химических программ на ЭВМ. Конкурс был объявлен в августе 1988 г., а информация об этом конкурсе была дана в газете «Наука в Сибири» и в третьем номере журнала «Химия и жизнь» за 1989 г. По условиям конкурса, заявки на участие в нем принимались из всех населенных пунктов СССР. Ограничений на тип ЭВМ не было.

Необходимость в проведении такого конкурса была вызвана целым рядом причин, главные из которых:

1. В последние годы не только в СССР, но и во всем мире наблюдается падение интереса к естественным наукам, уменьшаются конкурсы в вузы, ухудшается качество знаний абитуриентов и выпускников. Химия в этом отношении отнюдь не исключение. Углубление существующего кризиса обернется очень большими потерями для народного хозяйства. Чтобы противостоять этой тенденции, необходимы активные действия.

2. С ростом количества и улучшением качества ЭВМ внедрение их в учебный процесс будет происходить очень быстрыми темпами, и здесь необходимо заранее подготовить соответствующий программный продукт. Особенно это важно для школьного образования, когда у детей начинается формирование профессиональных интересов.

3. Программы на ЭВМ необходимы не для того, чтобы заменить школьного учителя, а, наоборот, для того, чтобы сделать общение учителя и ученика более содержательным, индивидуальным и действенным. Необходимо оказать существенную помощь школьным учителям в улучшении качества преподавания химии, показать привлекательность и глубину химической науки.

Проведение такого конкурса впервые позволило разработчикам обменяться мнениями, опытом, требованиями к программам, посмотреть, кто и на каком уровне работает. Участие учителей в конкурсной комиссии позволило разработчикам более четко представить себе тот уровень требований, который необходимо предъявлять к программному продукту. В комиссию вошли также ведущие ученые НГУ.

Что же показал конкурс? Разработки школьных химических программ ведутся во

многих регионах страны. На конкурс было заявлено 12 пакетов программ из Москвы, Минска, Новосибирска, Калининграда, Улан-Удэ, Челябинска, Краснодара, Кишинева, Белгорода, но из-за неявки двух коллективов в конкурсе участвовало десять пакетов программ. Как и предполагалось, не существует какого-либо приоритетного типа ЭВМ, на котором составляются программы. Разнообразие ЭВМ сильно ограничивает возможности передачи программ от разработчиков к пользователям.

Были представлены программы от обучающихся решению химических задач до контролирующих и игровых. Хорошее впечатление произвел пакет программ Белорусского государственного университета, посвященный обучению школьников простым количественным химическим расчетам. Следует отметить нестандартность идеи и техническое воплощение программы «Химические шашки», разработанную в том же университете. Оригинальна идея программы «Дисфор» Новосибирского университета, хотя программа нуждается в наполнении и серьезной обкатке.

Несколько слов об итогах конкурса. В результате проведенного тайного голосования места распределились так:

1. Пакет игровых и обучающих программ по неорганической и общей химии Новосибирского государственного университета, рук. Н. Сорокин, Н. Шохирев, А. Пусеп.

2. Пакет обучающих программ по органической химии Белгородского технологического института строительных материалов, авт. И. Тикунова и др.

3. Комплекс программных средств «Дисфор» Новосибирского государственного университета, рук. А. Жижин.

4. Пакет программ по органической химии Белорусского государственного университета, авт. Н. Рафальский и др.

5. Программа «Химические шашки» Белорусского государственного университета, авт. П. Гляков и Р. Варова.

6. Пакет программ по неорганической химии Белорусского государственного университета, авт. Н. Рафальский и др.

7. Пакет программ по теме «Электролитическая диссоциация. Ионные реакции» Московского государственного педагогического института им. В. И. Ленина, рук. Н. Гоголевская.

8. Комплекс программ по неорганической химии Краснодарского политехнического института, авт. В. Корпачев.

9. Комплекс обучающих программ по химии «Пушок — Пух-7» Калининского научно-производственного объединения «Центрпрограммсистем», рук. А. Диковский.

10. «Химический тренажер» (по И. Нентвингу и др.), авт. А. Шабуров.

Исходя из количества набранных баллов (из-за незначительности расхождений в оценке работы, занявшей третье место), комиссия рекомендовала отметить также третьей премией работу, занявшую четвертое место.

Комиссия отмечает, что необходимо обращать большее внимание на создание и разработку оригинальных химических идей, демонстрирующих глубину предмета и его привлекательность. Программный продукт должен обладать большей защищенностью при неправильных действиях учащихся. Все участники конкурса высказали пожелание о необходимости регулярного проведения специальных конференций и семинаров, посвященных применению ЭВМ в химическом

образовании. Разобщенность разработчиков приводит к чрезвычайно медленному внедрению наработанного программного продукта в школы. Поэтому комиссия считала бы целесообразным создание специального центра по применению ЭВМ в школьном и вузовском химическом образовании при Новосибирском государственном университете, который бы взял на себя задачу по координации работ по созданию программного обеспечения. Авторам работ, занявших призовые места, рекомендовано опубликовать материалы о своих разработках в журнале «Преподавание химии в школе».

В целом комиссия считает, что подобные конкурсы могут оказать благотворное влияние на развитие работ по применению ЭВМ в химическом образовании.

Н. БАЖИН

Семинар в Узбекистане

19 января в средней школе им. В. И. Ленина г. Хивы прошел семинар-совещание учителей начальных классов и преподавателей информатики школ города. Проводил его член-корреспондент АПН СССР В. Болтянский. В работе семинара приняли участие зав. кабинетами информатики республиканского и областного ИУУ, методисты, завучи по начальным классам школ города. Тема семинара — «Применение средств вычислительной техники в начальных классах». В числе других семинар ставил целью опровергнуть популярное среди родителей мнение о вредности вычислительной техники. На основании опыта, приобретенного в результате десятилетнего эксперимента в школах Еревана, В. Болтянский показал, насколько может быть эффективным урок с применением компьютеров и калькуляторов в на-

чальных классах, и продемонстрировал, чему и как учить малышей на компьютерах КУВТ-2 «Ямаха». К слову, компьютеры японского производства установлены у нас в мае 1989 г. Как видите, срок небольшой, а результаты просто ошеломляющие: не только старшеклассники «заболели» компьютерами, на них учатся пятиклассники из спецшахматных классов, учатся русскому и немецкому языку многочисленные посетители нашего кабинета ВТ.

В распоряжение школы, как, впрочем, и других школ города, предоставлен пакет программных средств по обучению началу геометрии детей в начальных классах.

Семинар прошел очень живо и, на наш взгляд, способствовал решению некоторых актуальных вопросов практической педагогики.

Р. САБИРОВ, учитель информатики

Куда не дотянулись провода

Сколько персональных компьютеров сможет обеспечить мощностью одна лошадиная сила? Не так уж и много — не более десяти. А вот для нового одноплатного компьютера достаточно мощности, выделяемой всего-навсего одним хомячком. Несмотря на столь скромный аппетит, эта микро-

ЭВМ имеет 16/32 битный процессор с тактовой частотой до 10 МГц, два стандартных последова-

тельных порта PS-232 и 20-разрядный параллельный порт ввода/вывода, два таймера, часы-календарь и даже четырехканальный 8-битный аналогово-цифровой преобразователь.

И хотя такой компьютер, при самом широком распространении, энергетической проблемы не решит, его появление позволит современным информационным технологиям проникнуть туда, куда еще не дотянулись провода.

ЧТО?
МОЖЕТ
ЭВМ

КУВТ на базе ПК-8000 «Сура», «Веста» — это комплекс микро-ЭВМ, предназначенный для использования в учебном процессе.

В его состав входят рабочие места учащихся, объединенные в локальную вычислительную сеть, и рабочее место преподавателя, оборудованное:

сдвоенным НГМД (5,25" общей емкостью 1,6М байта);
дисплеем, магнитофоном, принтером;
устройством контроля за работой учащихся;
цветным демонстрационным телевизором.

Это позволяет преподавателю проконтролировать работу каждого учащегося, а также обеспечивает быструю и эффективную загрузку программ в компьютеры.

В КУВТ «Сура» реализована операционная система CP/M. В ее рамках создан широкий спектр программ, включающий, в частности, трансляторы с языков Си, Паскаль, Ада, системы управления базами данных.

В индивидуальном режиме ПЭВМ «Сура» работает со встроенным интерпретатором языка Бейсик, что позволяет автономно использовать каждое рабочее место.

КУВТ «Сура» — это большой пакет прикладных программ:

программы-тренажеры для приобретения навыков работы с компьютером;

полное программное сопровождение курса информатики;

программы, обучающие использованию Бейсика;

лабораторный практикум для практического освоения Бейсика;

учебные программы по общеобразовательным предметам (физика, химия, английский язык);

игровые и демонстрационные программы.

Итак, кооператив «ПРОЛОГ» предлагает:

оборудование компьютерных классов на базе ПК-8000 «Сура», «Атари», «Спектрум»;

локальную сеть для ПЭВМ «Сура», «Веста»;

накопитель на гибких магнитных дисках для КУВТ;

пакет оригинальных обучающих программ;

широкий выбор системных программ.

Познакомиться более подробно, получить интересующую информацию Вы сможете в любое удобное для Вас время по адресу:

310024, Харьков, ул. Петровского, 30-А, кооператив «Пролог».

Тел. 43-28-28.

В. ЖАВОРОНКОВ

Барсикиана, или Новые операторы языков программирования

Пользователям ЭВМ хорошо знакома ситуация, когда периферийные устройства (дискеты, принтер, графопостроитель и т. д.) простаивают и попусту транжируют электроэнергию, дожидаясь, когда центральный блок соизволит начать с ними диалог.

Включившись в борьбу за экономию энергоресурсов, системные программисты нашей организации разработали расширенную (R) версию Бейсика, получившую название Барсик (BARSIC), с операторами, включающими и выключающими из электросети в нужный момент нужные устройства. Об этих результатах уже сообщалось в печати (Наука и жизнь. 1989. № 4), там же приводилась Барсик-программа поиска первых 10 тыс. простых чисел проверкой делимости нечетных чисел натурального ряда на ранее найденные простые числа, хранящиеся в массиве P (расп. 1). Определив эти числа, машина включает принтер, распечатывает на нем массив P, выключает принтер, включает дисковод, заносит массив P на диск, выключает дисковод. Последний оператор программы не только выключает сам компьютер, но и гасит свет в помещении.

```
'Программа поиска простых чисел
'по алгоритму DO-DO-DO (давай-давай-давай)
'на языке BAR-SIC (r - Рачительный)
K=10000: DIM P%(K): P%(1)=2: N=1: I=1
DO ' давай
  DO ' давай
    N = N + 2: S = INT(SQR(N)): J = 0
    DO ' давай
      J = J + 1: P = P%(J)
      LOOP UNTIL P >= S OR N / P=INT(N/P)
    LOOP UNTIL N / P > INT(N / P)
  I = I + 1: P%(I) = N
```

```
LOOP UNTIL I = K
' Работа с принтером и дисководом
SWITCH ON PRINTER' Включение принтера
FOR I = 1 TO K: LPRINT I, P%(I): NEXT
SWITCH OFF PRINTER' Выключение принтера
SWITCH ON DISK' Включение дисковода
OPEN "DATA" FOR OUTPUT AS #1
FOR I = 1 TO K: PRINT #1, I, P%(I): NEXT
CLOSE
SWITCH OFF DISK' Выключение дисковода
SWITCH OFF COMPUTER AND SWET V KOMPATE
END
```

Бережливость — не главная цель новых операторов Барсика, хотя их использование позволило нам последние 948 простых чисел рассчитать на сэкономленной электроэнергии. Более важно то, что Барсик позволяет включать и выключать не только свой собственный компьютер, но и другие машины, объединенные с ним в сеть. Ведь создать компьютерную сеть — это еще только полдела. Не менее важно иметь возможность включить в электросеть соседний компьютер, после чего ему можно будет передавать информацию, выключив эту машину по окончании сеанса связи.

Расскажем и о других особенностях Барсика.

Работу на ЭВМ намного облегчают так называемые функциональные клавиши, позволяющие вводить целые цепочки символов (RUN, LIST, LOAD и др.) одним-единственным нажатием. В компьютерном мире встречаются и так называемые тактильные дисплеи, чувствующие прикосновение пальца программиста к стеклу экрана, на котором могут быть, например, написаны те же слова RUN, LIST... Барсик же реализует более привычную для отечественного пользовате-

ля технологию общения с ЭВМ. В этом языке определенными функциями нагружены удары по разным частям дисплея и другие подобные действия пользователя. Пример: исходный текст программы «Календарь» содержит две ошибки — «Суббота» и «Deу», причем вторая «ломает» алгоритм за счет «проглатывания» машиной новой переменной.

Как известно, искусство программирования — это в первую очередь не умение составлять программы, а способность отыскивать в них ошибки. Как поступает мастер, уронивший в недра телевизора какую-нибудь детальку, мешающую, как и ошибка в программе, нормальной работе аппарата? Он подстилает газету и трясет над ней ремонтируемым прибором до тех пор, пока лишнее не выпадет. Таким же образом на Барсике отлаживают программы: ее дефектный участок выводят на экран, приподнимают дисплей и трясут им до тех пор, пока все ошибки не высыпятся на своеобразную газету — в специально выделенный участок программы (нижняя часть расп. 2). Удар по левой стороне дисплея (другая встроенная функция языка) «паскализует» Барсик-программу — выделяет ее структуру отступами от левого края. При ударе по правой стороне дисплея исходный неструктурированный текст восстановится, что экономит (опять рачительность!) оперативную и архивную память машины, не забывая ее лишними проблемами.

```
' Исходный текст программы "Календарь"
' на языке BAR-SIC (r - Реагирующий)
INPUT "ГГ,ММ,ДД ";Year,Month,Day;
                                Month=Month+1
IF Month<4 THEN Month=Month+12;Year=Year-1
Day=INT(365.25*Year)+INT(30.6001*Month)+Day
SELECT CASE Day - 7 * INT(Day / 7)
CASE 0:  .? "Воскресенье"
CASE 1:  ? "Понедельник"
CASE 2:  ? "Вторник"
CASE 3:  ? "Среда"
CASE 4:  ? "Четверг"
CASE 5:  ? "Пятница"
CASE ELSE: ? "Суббота" ' случай 6
END SELECT
```

```
' Текст программы
' после удара по левой стенке дисплея
' и потряхивания дисплея с текстом программы
INPUT "ГГ,ММ,ДД ";Year,Month,Day
Month = Month + 1
IF Month < 4 THEN
    Month = Month + 12; Year = Year - 1
END IF
Day=INT(365.25*Year)+INT(30.6001*Month)+Day
SELECT CASE Day - 7 * INT(Day / 7)
CASE 0
PRINT "Воскресенье"
```

```
CASE 1
PRINT "Понедельник"
CASE 2
PRINT "Вторник"
CASE 3
PRINT "Среда"
CASE 4
PRINT "Четверг"
CASE 5
PRINT "Пятница"
CASE ELSE
PRINT "Суббота" ' CASE 6
END SELECT
NEWSPAPER' Подстелили газету для
                                сбора ошибок
Deу (Day) Субота (Суббота)
'Ошибка (Исправлено)
END NEWSPAPER
```

Функциональные нагрузки несут и остальные части дисплея. Так, удары по крышке позволяют постранично пролистывать текст программы; для вызова HELP-функции нужно постучать по стеклу дисплея и спросить «Что там?» и т. д.

Все, когда-либо писавшие программы с развитым диалогом «человек — компьютер», хорошо знают, что подбор «обоймы» нужных реплик машин — дело довольно трудное и ответственное. Часто основные силы уходят не на саму программу, а на ее украшательство меткими фразами. В выигрыше здесь оказываются программисты, про которых говорят, что они «за словом в карман не лезут». Барсику еще далеко до таких людей, но...

Программа на расп. 3 реализует подобную технологию ответов машины на действия человека. В цикле с выходом из середины (делай до тех пор, пока не LOOPнет чье-то терпение) ЭВМ запрашивает у пользователя пароль (здесь число 17), знание которого позволяет пользователю войти, например, в какую-то программную среду.

```
' Программа "Пароль" на языке
' BAR-SIC (r - сообразительный)
DO ' цикл запроса пароля
PRINT "В А Ш П А Р О Л Ь ": A$ = ""
FOR I = 1 TO 2' Запрос двух байт
B$ = ""
DO : B$=INKEY$ : LOOP WHILE B$= ""
A$ = A$ + B$
NEXT
IF A$ = "17" THEN EXIT DO
GET WORD FROM POCKET
' полез в карман за репликой
PRINT WORD ' печать реплики
LOOP
POCKET' карман компьютера
"Рад познакомиться.",
"А мы вас тут давно ждали.",
"Ты опять здесь, подлый трус.",
"А у Вас вся спина белая",
```


"ТОВАРИЩИ! УБЕРИТЕ СКОРЕЕ
ЭТОГО ЧЕЛОВЕКА.",

"Не жми на эту клавишу,
Балда, жми на другую",
"Ой больно!", "А ты кто такой?",

.....
END POCKET

Ошибки ввода машина комментирует своими репликами, «вынимаемыми из кармана». Программисту при этом достаточно только наполнить этот карман фразами из классической и новой литературы, а также услышанными в очередях и на транспорте. Такой карман заносится в библиотеку Барсика и при необходимости «пришивается» к вновь создаваемой программе, обогащая заложенный в ней диалог. После этого машина сама будет подбирать необходимую реплику в зависимости от конкретной ситуации, степени обидчивости человека, наличия у него чувства юмора и т. д. и т. п.

О новых необычных возможностях Барсика можно рассказывать довольно долго. Мы ограничимся еще одной.

Работая по Барсик-программе на рис. 3, машина, вытащив нужную реплику из кармана, выводит ее на экран дисплея. Но компьютер, имеющий звукогенератор и подружившийся с Барсиком, может литературные, булевские, числовые константы, переменные и выражения не только печатать на экране дисплея, но и проговаривать. Так, работая по Барсик-программе на рис. 4, машина отсортирует числовой массив и назовет его элементы в порядке возрастания. Оператор SPEAK использован здесь без параметров, что по умолчанию означает, что слова будут выговариваться по-английски, женским голосом, в среднем темпе. Подставляя в оператор SPEAK нужные параметры, можно менять и язык, и голос. Догадливый читатель уже сообразил, что благодаря этому проблема машинного перевода на Барсике решена полностью.

Программа на расп. 4, можно сказать,



Программа "Сортировка" на языке
BASIC (r - Разговорчивый)

```
PLAY "o3"  
DATA 5,3,-6,-69,-89,-5  
READ N:DIM A(N)  
FOR I=1 TO N: READ A(I): NEXT: FLAG=1  
PLAY "c1"  
PLAY "d1"  
WHILE FLAG = 1  
PLAY "c2"  
FLAG = 0  
PLAY "c2"  
PLAY "g3"  
FOR I = 2 TO N  
PLAY "e4"  
IF A(I) < A(I - 1) THEN  
PLAY "c5"  
SWAP A(I),A(I-1) : FLAG=1  
PLAY "c5"  
END IF  
PLAY "e4"  
NEXT  
PLAY "g3"  
WEND  
PLAY "d1": PLAY "c1"  
FOR I=1 TO N:SPEAK A(I):NEXT  
'выговариваем массив  
PLAY "c1"  
PLAY "c0"
```

127

пронизана звуком. Операторами PLAY отмечены края структурных блоков так, что при прогонке программы слышен алгоритм решения поставленной задачи. Особенно ценна такая особенность Барсика в программах, склонных к закликиванию, прогонка которых подобна проигрыванию заезженных грампластинок, где спираль звуковой дорожки превратилась в замкнутый круг. Разрывают такой круг легким ударом по проигрывателю. На Барсике также предусмотрена подобная функция-макрос: программа выходит из цикла после легкого удара по корпусу центрального блока машины (не все же должно доставаться одному дисплею).

От редакции. Хотя статья и шутовская, она может стать основой для серьезной работы. Написание сочинений на тему «Каких команд мне не хватает в языке программирования» и их разбор на занятиях в школе или вузе («Но почему же разработчики не сделали это?») могут стать полезным уроком, углубить понимание причин реализации именно таких языков программирования, операционных систем, прикладных программ, аппаратных средств, архитектур ЭВМ...

Ждем от читателей продолжения «Барсикианы». Очередной выпуск с наиболее интересными командами и абрисами путей их реализации планируется через год.

Учебный дисплейный класс

«ХТИБО»

Учебный дисплейный класс (УДК) «ХТИБО» разработан отраслевой научно-исследовательской лабораторией таймерных разрядно-аналоговых вычислительных систем Министерства промышленности средств связи СССР при Хмельницком технологическом институте совместно с Институтом проблем моделирования в энергетике АН УССР.

В УДК «ХТИБО» реализованы принципы построения распределенных вычислительных систем таймерного типа, в которых операционные взаимодействия и потоки информации представлены в виде интервалов времени определенной длительности. Это позволяет существенно упростить линии связи между вычислительными устройствами системы, сведя их к двухпроводной линии. Существенно упрощается также и аппаратурная часть в приемниках и передатчиках информации. Применение принципов разрядно-аналогового представления таймерных операндов позволило при сохранении двухпроводной линии связи получить очень высокую скорость передачи данных. Так, например, информация с пульта ввода дисплея при нажатии соответствующей клавиши обрабатывается и передается в линию в течение нескольких микросекунд. При условии применения микросхем с частотами переключения более 100 МГц это время уменьшается до сотых долей микросекунды.

С центральным процессором по двухпроводным линиям связи, исключая контроллер шины, связаны пульта ввода символьной информации таймерного типа. Благодаря таймерным операционным взаимодействиям принципиальная электрическая схема пульта ввода чрезвычайно упрощена (4 корпуса микросхем малой степени интеграции по сравнению с примерно 40 корпусами микросхем средней и большой степени интеграции в пульте ввода информации серийных дисплеев).

Связь пультов ввода с клавиатурой ПЭВМ осуществляется при помощи приемника таймерных операндов, к которому подключен дешифратор. Выходы дешифратора символов подключены к соответствующим контактам клавиатуры ввода.

Основная функция пульта ввода обучаемого — обеспечение ввода алфавитно-цифровой информации в ПЭВМ. Вводимая информация индицируется на дисплее ПЭВМ и на экранах мониторов на каждом рабочем месте.

Управление процессом обучения и контроля осуществляется с коммутатора включения рабочих мест и клавиатуры ввода ПЭВМ. Предусмотрена возможность одновременного включения в линию связи всех пультов ввода обучаемых.

Годовой экономический эффект от использования одного УДК в расчете на 12 рабочих мест в зависимости от комплектности составляет около 105 тыс. руб

Первые серийные образцы УДК эксплуатируются в учебном процессе в Хмельницком технологическом институте, Тернопольском и Винницком медицинских институтах, в средних школах Хмельницкой области, в Хмельницкой морской школе ДОСААФ, в Институте проблем моделирования и энергетики АН УССР и внедряются в настоящее время в других отраслях народного хозяйства.

На базе ПЭВМ «Нейрон-И9.66» готовится серийное производство УК «Нейрон» (учебный комплекс «Нейрон») в ПО им. Королева (Киев).

Учебный дисплейный класс «ХТИБО»

**ориентирован на использование в учебном процессе
высших учебных заведений любого профиля,
техникумов, средних общеобразовательных школ
и профессионально-технических училищ**

Конструктивно УДК представляет собой распределенную в пространстве вычислительную систему на базе центрального диалогового вычислительного комплекса типа «Нейрон-И9.66», а также других ПЭВМ, совместимых с ПЭВМ класса IBM PC AT.

В состав УДК входят: до нескольких десятков пультов ввода информации и столько же мониторов типа MS 6105 или других типов с размерами экрана по диагонали не менее 32 см, микро-ЭВМ либо ПЭВМ в комплекте с накопителями на гибких магнитных дисках и печатающим устройством, соединительные кабели, двухпроводная линия связи, а также коммутатор для включения рабочих мест в линию связи.

Разработанное программное обеспечение включает в себя: обучающие программы по языкам программирования, по элементарной математике, решение задач из области высшей математики, учебные программы по физике, медицинской статистике, деловые и развлекательные игры и др.

УДК «ХТИБО» в 1987, 1988 и 1989 гг. демонстрировался на ВДНХ СССР и награжден серебряными медалями выставки. В 1989 г. демонстрировался на ВДНХ УССР и награжден золотой медалью.

В 1988 г. УДК «ХТИБО» экспонировался на Международной ярмарке в г. Пловдиве (НРБ), а в 1989 г. — на Международной ярмарке в г. Дели (Индия) и на Международной выставке «Электроника-2000» в Москва.

При необходимости изготовления УДК заказчику в соответствии с договором поставляется конструкторская документация и оказывается помощь в настройке.

Адрес для справок:

2800016, г. Хмельницкий, ул. Институтская, 11;
тел. 2-51-34, 2-50-19.

Б.Ю. 11-41

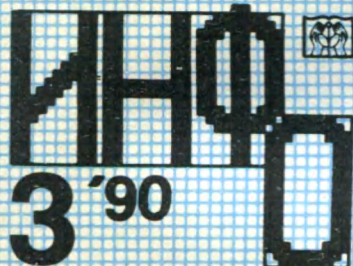
Цена 60 коп.

70423

OldPC.ru

7003

музей компьютеров



**ИНФОРМАТИКА
И ОБРАЗОВАНИЕ**

