

ДЕНЬГИ НЕ ГЛАВНОЕ?

Вы никогда не задумывались, на чем держится рынок freeware и того shareware, которое по факту freeware (когда что-то время от времени вежливо просят, но никак не ограничивают функциональность)?

Заметьте — среди программного обеспечения freeware и shareware во множестве встречаются не только мелкие узкоспециализированные программки, но и масштабные качественные продукты уровня архиваторов, файловых менеджеров, разнообразных редакторов, серверов, систем управления базами данных и даже операционных систем. Нередко свободно распространяемые программы лучше и удобнее их коммерческих аналогов.

Примеров можно приводить много, отмечу лишь первый, который пришел мне в голову, — пока я писал этот текст, у меня несколько раз “просыпалась” аська. Так вот, в качестве клиента я уже довольно давно использую QIP. Кто не знает — рекомендую: www.qip.ru. Конечно, строго говоря, фирменный клиент ICQ тоже бесплатная программа, но, во-первых, QIP избавляет меня от просмотра назойливой рекламы, а во-вторых, он просто удобнее и функциональнее.

Почему же так много хороших программистов, труд которых востребован и неплохо оплачивается, тратят немало времени на бесплатные или условно-бесплатные авторские разработки? Конечно, нельзя не признать, что и здесь коммерческий интерес, возможно, является доминирующим — успешные проекты, даже оставаясь бесплатными, начинают приносить значительную прибыль за счет издания литературы, технической поддержки и прочих сопутствующих услуг. За примерами далеко ходить не надо — сказанное касается и популярной СУБД MySQL, и сверхпопулярного языка программирования на стороне web-серверов PHP, и десятков, если не сотен, других широко распространенных программных продуктов.

Однако, во-первых, далеко не все свободно распространяемые программы способны, даже в перспективе, приносить прибыль, а во-вторых, прежде чем проект принесет первый рубль (или доллар), все равно приходится немало поработать, причем безо всякой уверенности в скором светлом будущем.

Так почему же?

Моя гипотеза: тому есть две причины. Во-первых, людей творческих, склонных создавать что-либо новое и полезное, в принципе довольно много. Но при этом программирование — уникальная сфера изобретательского творчества, в которой автор имеет реальную возможность и целиком создать продукт своими руками (нет необходимости в материальных составляющих — деталях, оборудовании и пр.), и самостоятельно представить его широкой (потенциально — практически необъятной) аудитории, и быстро получить обратную связь с пользователями-потребителями.

Я нередко обращаю внимание своих учеников на эту особенность профессии программиста. И мне кажется, что для некоторых из них, связывающих свою профессиональную деятельность с информационными технологиями, возможность “творить для миллионов”, участвовать в масштабных проектах по разработке свободного ПО является существенным фактором при выборе профессии.

С.Л. Островский,
главный редактор

СОДЕРЖАНИЕ НОМЕРА

ИНФОРМАЦИЯ

Набор слушателей на очно-заочные курсы повышения квалификации (для жителей Москвы и Московской области) 2

Набор слушателей на дистанционные курсы повышения квалификации. Второй поток 2

КУРСЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ

В.Ф. Бурмакина, И.Н. Фалина. Как готовиться к тестированию по проверке ИКТ-компетенции школьников. Лекция 5. Описание познавательных деятельностей, составляющих компетентность “оценка информации” 3–8

АЗЫ ПСИХОЛОГИИ — УЧИТЕЛЮ ИНФОРМАТИКИ

В.П. Арсланьян. Гении или чудаки? 9, 48

ПРОФИЛЬНАЯ ШКОЛА. ЭКЗАМЕНЫ

Е.А. Еремин, А.П. Шестаков. Примерные ответы на профильные билеты 10–17

ТЕЛЕКОММУНИКАЦИИ

А.А. Дуванов. Азы информатики. Выходим в Интернет 17–24

“НАЧАЛКА” № 14

Газета-клуб для всех, кто учит информатике маленьких детей

Н.Г. Иванова, М.А. Плаксин, О.Л. Русакова. Материалы конкурса “ТРИЗформашка-2006”. Часть I 25–36

РЕКЛАМА

Тематический выпуск “Профильное обучение” .. 36
Годовые подшивки газет на компакт-дисках 47

“В МИР ИНФОРМАТИКИ” № 80

Газета для пытливых учеников и их талантливых учителей

Моделирование

Е.А. Еремин. Моделирование работы ЭВМ с помощью программы Microsoft Excel 37–40

“Ломаем” голову

Три письма 40
Задачник
Постоянная Капрекара .. 41

Школа программирования
Н.М. Тимофеева. Основы программирования на Visual Basic 41–43

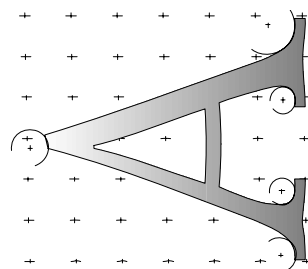
Microsoft Excel углубленно
Рекуррентные соотношения в Microsoft Excel 43–44

Games.exe

Еще раз о фокусе “Дата рождения” 44–45
Конкурс № 52
для учащихся 45–46

№ 21 (526)

1–15 ноября 2006



Методическая газета для учителей информатики

ИНФОРМАТИК



ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ «ПЕРВОЕ СЕНТЯБРЯ»
 ГАЗЕТА «ИНФОРМАТИКА»
 ОТДЕЛЕНИЕ ПЕДАГОГИЧЕСКОГО ОБРАЗОВАНИЯ ФГП МГУ ИМ. М.В. ЛОМОНОСОВА

ОБЪЯВЛЯЮТ НОВЫЙ НАБОР СЛУШАТЕЛЕЙ НА ТЕКУЩИЙ УЧЕБНЫЙ ГОД

НОВОЕ ПРЕДЛОЖЕНИЕ!

для жителей Москвы и Московской области

ОЧНО-ЗАОЧНЫЕ КУРСЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ

Объявляется набор на очно-заочные курсы повышения квалификации с нормативным сроком освоения учебного материала 72 часа (36 часов очно, 36 — заочно). Занятия будут проходить в феврале–апреле 2007 г., в удобное время во второй половине дня, 1 раз в неделю. Наш комфортный учебный центр расположен рядом со ст. м. «Киевская».

Мы предлагаем два курса для учителей информатики:

Код	Курс
107-001	А.А. Дуванов, С.Л. Островский. Основы web-дизайна и школьного сайтостроительства
107-002	Ю.А. Первин. Обоснования и методика школьного курса информатики

Мы также предлагаем два общепедагогических курса, предназначенных для всех работников образования:

121-001	В.М. Букатов. Режиссура урока в современной школе
121-002	А.П. Ершова. Театральное мастерство в работе современного учителя

Стоимость обучения — 3900 руб. Для тех, кто ранее обучался (или в настоящее время обучается) на наших дистанционных курсах, — 3400 руб. Заявки на очно-заочные курсы можно подавать по телефонам: (499) 249-47-82, (495) 249-52-53 и на сайте <http://edu.1september.ru> Регистрация слушателей производится с 1 ноября 2006. Количество мест в группах ограничено.

ВТОРОЙ ПОТОК

для всех работников образования вне зависимости от места их проживания

ДИСТАНЦИОННЫЕ КУРСЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ

Объявляется набор на **второй поток 2006/2007 учебного года**. Курсы проводятся в режиме дистанционного обучения. Взаимодействие со слушателями производится посредством обычной или, при наличии у слушателя возможности, электронной почты. Нормативный срок освоения учебного материала — 72 часа. Начало занятий — январь 2007 г.

После успешного окончания курсов слушатели получают удостоверение установленного образца о прохождении курсов повышения квалификации от Педагогического университета «Первое сентября» и Отделения педагогического образования ФГП МГУ им. М.В. Ломоносова.

Стоимость обучения составляет 990 рублей за один курс.

На втором потоке 2006/2007 учебного года мы предлагаем три дистанционных курса для учителей информатики:

Код	Курс
07-001	И.Г. Семакин. Информационные системы в базовом и профильном курсах информатики
07-002	Е.В. Андреева. Методика обучения основам программирования на уроках информатики
07-006	А.А. Дуванов. Основы web-дизайна и школьного сайтостроительства

Мы также предлагаем общепедагогический курс, предназначенный для всех работников образования:

21-001	С.С. Степанов. Теория и практика педагогического общения
--------	---

Для зачисления на курсы необходимо прислать в Педагогический университет «Первое сентября» заявку. Пожалуйста, используйте только приведенный ниже бланк или его ксерокопию. Регистрация слушателей производится с 1 ноября по 31 декабря 2006 г. (дата отправки заявки фиксируется по почтовому штемпелю предприятия-отправителя). Вам будет выслан комплект документов с подробной информацией о курсах и счетом для оплаты. Вы оплатите счет, если вас устраивают предлагаемые условия (факт подачи заявки ни к чему не обязывает). Заявки следует направлять по адресу: Педагогический университет «Первое сентября», ул. Киевская, д. 24, Москва, 121165.

Заявки также можно подавать on-line на сайте <http://edu.1september.ru> Справки по тел.: (499) 249-47-82, e-mail: edu@1september.ru

Прошу выслать мне комплект документов для зачисления на **ДИСТАНЦИОННЫЕ КУРСЫ** повышения квалификации

ФАМИЛИЯ	<input type="text"/>	Я хочу пройти обучение по дистанционным курсам (укажите коды выбранных вами курсов):
ИМЯ	<input type="text"/>	
ОТЧЕСТВО	<input type="text"/>	
ИНДЕКС	<input type="text"/>	
АДРЕС	<input type="text"/>	<input type="text"/> — <input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/> — <input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/> — <input type="text"/>

Телефон (с кодом города): () E-mail (если есть):

Место работы: Должность:

ВНИМАНИЕ! К обучению на курсах повышения квалификации допускаются сотрудники образовательных учреждений, работающие по соответствующей специальности.

Если вы обучались в прошлом учебном году на наших курсах, укажите ваш идентификатор: **07-21**

ГАЗЕТА "ИНФОРМАТИКА" И ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ "ПЕРВОЕ СЕНТЯБРЯ"
(лицензия А225682, № 020503 от 19.07.2006)
ПРЕДСТАВЛЯЮТ НОВЫЙ ДИСТАНЦИОННЫЙ КУРС
ПОВЫШЕНИЯ КВАЛИФИКАЦИИ ДЛЯ УЧИТЕЛЕЙ ИНФОРМАТИКИ

В.Ф. Бурмакина, И.Н. Фалина

Как готовиться к тестированию по проверке ИКТ-компетенции школьников

Учебный план

№ газеты "Информатика"	Учебные материалы
17/2006	Лекция 1. Цели и задачи проекта по оценке ИКТ-компетентности девятиклассников. Цели и метод тестирования. Описание когнитивных компетентностей, оцениваемых тестом. Структура теста.
18/2006	Лекция 2. Описание познавательных деятельности, составляющих компетентность "определение информации". Примеры заданий на выработку этого умения.
19/2006	Лекция 3. Описание познавательных деятельности, составляющих компетентность "доступ к информации". Как выбрать информационный ресурс, соответствующий заданным критериям? Какие стратегии можно и нужно использовать для поиска информации в многочисленных, часто противоречивых, источниках?
20/2006	Лекция 4. Описание познавательных деятельности, составляющих компетентность "управление информацией". Могут ли ваши ученики разработать самостоятельно или выбрать из предложенных такое представление исходной информации, которое будет наиболее понятно конкретной аудитории? Понимают ли ваши ученики, что с конфиденциальной информацией надо обращаться в соответствии с определенными нормами? Контрольная работа № 1.
21/2006	Лекция 5. Описание познавательных деятельности, составляющих компетентность "оценка информации". Школьнику предлагается несколько информационных источников (например, статьи, сайты и т.п.), из которых он должен выбрать один, наиболее полно удовлетворяющий заданной потребности.
22/2006	Лекция 6. Описание познавательных деятельности, составляющих компетентность "интеграция информации". Если школьник умеет анализировать различные источники по одной и той же теме и на основе этой деятельности создавать новую информацию в сжатом и точном изложении, то он обладает компетентностью "интеграция информации".
23/2006	Лекция 7. Описание познавательных деятельности, составляющих компетентность "создание информации". Школьник обладает компетентностью "создание информации", если он умеет сбалансированно осветить проблему на основе имеющейся, в том числе и противоречивой, информации. Контрольная работа № 2.
24/2006	Лекция 8. Описание познавательных деятельности, составляющих компетентность "сообщение информации". Могут ли ваши ученики адаптировать представленную информацию для конкретной аудитории, умеют ли грамотно цитировать источники, воздерживаться от провокационных высказываний при передаче информации конкретной аудитории?
Итоговая работа. Итоговая работа должна быть отправлена в Педуниверситет "Первое сентября" до 28 февраля 2007 г.	

ЛЕКЦИЯ 5.

Описание познавательных действий, составляющих компетентность "оценка информации"

Очевидно, что оценивание какого-либо события, явления или предмета должно проводиться по определенным критериям. Можно ли сказать, что информация обладает такими качественными характеристиками, по которым ее можно оценить? Конечно, в курсе информатики как раз-то и изучаются свойства, или качественные критерии информации. Достаточно удачно, с большим количеством примеров, эта тема описана в учебнике *Бешенкова С.А., Ракутиной Е.А.* "Информатика. Систематический курс" [1].

Казалось бы, имея критерии, по которым можно оценивать информацию, задача оценки информации должна быть достаточно проста. Однако тестирование показало, что далеко не все девятиклассники с такими задачами справились успешно. Напомним результаты тестирования в пилотных ре-

гионах по когнитивному умению "оценка информации":

- базовый уровень — 31%;
- ниже базового уровня — 28%;
- выше базового уровня — 41%.

На наш взгляд, основная причина невысоких результатов заключается в том, что умению оценивать информацию в школе практически не учат. Действительно, любая учебная литература содержит информацию достоверную, актуальную, полную и определенную. В наших учебниках нет разных точек зрения на одну и ту же проблему. Так уж сложилось, что наши учебники близки по духу к академическому стилю изложения: в них не принято давать неполную или противоречивую информацию. Однако с необходимостью оценивать информацию все мы — и школьники в том числе — сталкиваемся

Свойства информации

1. Объективность	Субъективность	Информация <i>объективна</i> , если она не зависит от чьего-либо мнения
2. Достоверность	Недостоверность (ложность)	Информация <i>достоверна</i> , если она отражает истинное положение дел
3. Полнота	Неполнота (недостаточность), а также избыточность	Информация <i>полна</i> , если ее достаточно для понимания и принятия решения
4. Актуальность (своевременность)	Неактуальность (устаревшая информация или преждевременная)	Информация <i>актуальна</i> , если она важна, существенна для настоящего времени
5. Полезность (ценность)	Бесполезность	<i>Полезность</i> информации оценивается по тем задачам, которые мы можем решить с ее помощью
6. Понятность	Непонятность	Информация <i>понятна</i> , если она выражена на языке, доступном для получателя
7. Определенность (однозначность)	Неопределенность (многозначность)	Это свойство важно для восприятия и понимания человеком текстов, записанных на естественном языке

постоянно. Например, компания ребят решила пойти в кино. Есть выбор, надо принять решение, т.е. оценить, какой фильм стоит посмотреть.

Что же нужно и можно сделать, чтобы на школьных уроках у учащихся формировалось когнитивное действие «оценка информации»? Нацеленность на формирование этого действия потребует от учителей в большей мере использования новых методов обучения и, в частности, использования в педагогической практике контекстного обучения [2].

Технология контекстного обучения

Данный раздел лекции построен на основе материалов книги: Лаврентьев Г.В., Лаврентьева Н.Б., Неудахина Н.А. «Инновационные обучающие технологии в профессиональной подготовке специалистов» [8].

Контекстное обучение следует относить к образовательным технологиям, чья главная задача состоит в оптимизации преподавания и учения с опорой не на процессы восприятия или памяти, а прежде всего на творческое, продуктивное мышление, поведение, общение. Вот почему в контекстном подходе особую роль играют активные методы и формы обучения, обеспечивающие интенсивное развитие личности учащегося и педагога. Конструирование учебного процесса в современной педагогической практике осуществляется либо на основе обучения через информацию, либо на основе обучения через деятельность. В работе А.А. Вербицкого дано определение контекстного обучения как концептуальной основы для интеграции различных видов деятельности учащихся (учебной, научной, практической) [4].

Обоснование контекстного подхода

Традиционная дидактическая система видит свою глобальную задачу в том, чтобы приобщить обучаемых к обобщенному и систематизированному опыту человечества. Из этого вытекает утверждение ве-

душей роли теоретических знаний в содержании обучения, ориентация на усвоение основ наук. Естественно, это ведет к интеллектуализму, отрыву теории от практики, к тому, что социальная практика замещается знаковой системой — учебной информацией. Знаковые системы (в основном — тексты) «замещают» реальную действительность для конкретного человека, как бы вырезают его из собственного пространственно-временного контекста. Обучаемые усваивают через массив учебной информации то, что наработано другими, им навязывают цели усвоения кем-то добытых знаний, в результате учебная информация теряет для них личностный смысл.

Для постиндустриального (информационного) общества эта традиционная обучающая схема устарела, к тому же в традиционном образовании есть целый ряд противоречий, которые невозможно устранить в рамках традиционной технологии. Одним из таких противоречий является противоречие между целями и задачами обучения и между содержанием обучения и содержанием образования. В психологии задачей называют то, что требует решения, то, что субъект принял бы как «свою» цель. В педагогике же это «чья-то» цель — общества, учителя, то есть она внеличностна. В традиционной школе цели ученика заданы ему учителем, при этом часто теряется смысл учения, поскольку выполнять чьи-то задачи — неинтересно и несправедливо. Не желая учиться, наши дети борются против собственного будущего — не в этом ли заключается драма обучения?

Эта же логика включается, когда надо развести понятия «содержание обучения» и «содержание образования». Содержание обучения — это то, что содержится в учебных планах и программах. Очевидно, что при одном и том же содержании обучения люди получают разный уровень образования, что зависит от индивидуальных особенностей людей, личности педагогов, типа выполняемой ими

деятельности, уровня активности человека, материально-технических условий обучения.

Содержание образования — это уровень развития личности, предметной и социальной компетентности человека, который формируется в процессе обучения [5]. Следовательно, целью образования становится “выращивание” личностного потенциала. Но можно ли вырастить личность авторитарными способами, методами и условиями традиционной технологии обучения? Очевидно, что нельзя. Поэтому педагоги-новаторы пытаются создать активную обучающую среду с соответствующими методами и формами обучения. К сожалению, в школе новые технологии сдерживаются консерватизмом дидактических схем и моделей обучения.

Контекст как базовая категория технологии контекстного обучения (ТКО)

Понятие “контекст” пришло в другие науки из логики и лингвистики, поэтому в психологии и педагогике статуса категории оно еще не приобрело и в словарях этих наук не описано.

В психологии контекст связан с понятием “ситуация” (система условий, побуждающих субъекта и опосредующих его активность). В ситуацию включаются и внешние условия, и сам субъект, и те люди, с которыми он контактирует.

Лингвopsихологи отводят контексту основную роль в процессе переработки информации, так как именно благодаря контексту человек знает, чего ему ожидать и как осмысливать продукт восприятия. Например, обычное слово “собака” в контексте может означать вопрос, восклицание, утверждение, угрозу, восхищение и др. Прежде чем приступить к действию, человек старается собрать как можно больше контекстной информации. Чем больше мы знаем о настоящем, тем легче можем просчитать или предугадать будущее. Психологи называют такое опережающее отражение (преднастройку, ожидание, интуицию) *антиципацией*. Антиципации создаются под влиянием контекстов. Если у человека нет образцов поведения, зафиксированных в определенных контекстах, например, поведения в условиях кризиса, славы и др., то его организм реагирует импульсивно.

Сущность технологии контекстного обучения

С позиции ТКО основная цель любого образования — формирование компетентного в своей области человека. В основе ТКО лежит теория А.Н. Леонтьева о деятельностном усвоении умений и навыков. Реализация ТКО как в школе, так и в вузе сталкивается с большими трудностями. Их сложно осмыслить, но еще труднее преодолеть. В этом, кстати, кроется еще одно объяснение, почему такая

умная, полезная и эффективная технология, как контекстная, не находит широкого применения.

Первая трудность состоит в том, что формирование компетентностей мы обеспечиваем в рамках и средствами качественно иной деятельности — учебной, которая характеризуется своими собственными особенностями.

Вторая трудность — сами формы учебной деятельности не адекватны способам проявления формируемой компетентности.

Чтобы сформировать компетентного человека, надо обеспечить переход от одного типа деятельности (учебной) к другому (практической) с соответствующей сменой потребностей, мотивов, целей, действий, средств и результатов.

Третья трудность вытекает из второй: как именно преодолеть противоречия между учебной деятельностью и практической деятельностью, в которой требуется проявить компетентность в решении поставленной задачи (мы обсуждаем ИКТ-компетентность)? Традиционное обучение не может решить это противоречие, отсюда феномен формальных знаний, невозможность применения их на практике, трудность интеллектуальной и социальной адаптации выпускников школы к реальным условиям и требованиям жизни.

Ниже приведены две схемы: (1) — схема традиционного обучения; (2) — схема действия человека при решении задач, связанных с обработкой информации.

Схема 1

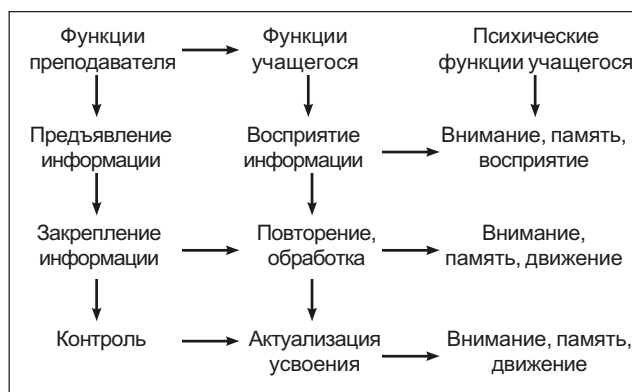


Схема 2



Как видно из схемы 1, в традиционном обучении функция мышления практически не востребована, основная нагрузка ложится на память, и в этом смысле, действительно, повторение — мать учения. Учащийся может извлечь из своей памяти лишь то, что в нее заложили в прошлом. Если учащийся и мыслит, то вопреки традиционной схеме обучения, а не благодаря ей.

Схема 2 позволяет развивать познавательную деятельность учащегося в обучении контекстного типа. В ней моделируется полный цикл мышления — от зарождения проблемной ситуации, порождения познавательной мотивации до нахождения способов разрешения проблемы и доказательства ее правильности. Именно такую схему целесообразно использовать при формировании ИКТ-компетентности.

Очевидно, что для реализации ТКО учитель должен использовать иные формы и методы учебной деятельности, которые заменяют пассивную роль учащегося в обучении (отвечает на вопросы преподавателя, выполняет задания по его указаниям) на активную позицию человека, которому надо принимать решения и нести за них ответственность. Контекстное обучение ориентируется на то, что знания, умения, навыки даются не как предмет, на который должна быть направлена активность учащегося, а в качестве средства решения практических задач. Если же быть совсем точным, то контекстное обучение рассматривает учение и труд не как разные виды деятельности, а как два этапа развития одной и той же деятельности [4].

Основной характеристикой учебно-воспитательного процесса контекстного типа, реализуемого с помощью системы новых и традиционных форм и методов обучения, является моделирование учебных заданий предметного и социального содержания. В учебных заданиях воссоздаются реальные жизненные ситуации. Таким образом, учащемуся задаются контуры его “взрослой жизни”. *Единицей работы* преподавателя и учащегося становится ситуация во всей ее предметной и социальной неоднозначности и противоречивости. Именно в ходе анализа ситуаций, деловых и учебных игр у учащихся формируются навыки обработки информации.

Характеристика познавательного действия “оценка информации”. Описание когнитивных действий, составляющих эту компетентность

В определении ИКТ-компетентности дается следующая структура когнитивного действия “оценка информации”:

Оценка информации	Выработка критериев для отбора информации в соответствии с потребностью
	Выбор ресурсов согласно выработанным или указанным критериям
	Умение остановить поиск

Когнитивная деятельность “оценка информации”, без сомнения, является общеучебным умением. Однако именно в курсе информатики четко выделяются и описываются свойства информации, но “назначать” ответственными за выработку этого умения учителей информатики было бы по крайней мере неверно, а по большому счету и вредно. На уроках информатики можно развить начальные навыки по оцениванию информации. Для этого, как всегда, на помощь придут специально разработанные дидактические задания и специально подобранные методы обучения. Рассмотрим каждую составляющую когнитивного действия “оценка информации” в свете такого подхода.

Формирование умения “выработка критериев для отбора информации в соответствии с потребностью”

В таблице на с. 4 приведены свойства (качественные критерии) информации. Четкое понимание значения каждого свойства — необходимое условие для успешной выработки критериев отбора информации в соответствии с заданной потребностью. Число различных комбинаций критериев велико, отработка на учебных заданиях каждого отдельного случая требует большого количества времени. На помощь в данной ситуации придут активные методы обучения. Например, можно использовать *анализ практических ситуаций*, или *баскет-метод, деловые и ролевые игры* [7].

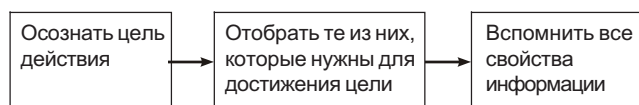
Пример. Игра “Секретарь” (форма проведения — баскет-метод: метод обучения на основе имитации ситуации).

Описание ситуации. Директор кондитерской фабрики “Сластена” задерживается в командировке. Он звонит секретарю и просит его отобрать материалы к докладу на Совете директоров на тему “Оценка нашей продукции СМИ”. Все материалы лежат у него на рабочем столе вместе с поступившей за прошедшие два дня корреспонденцией.

Дидактический материал. Учитель заранее готовит 6 одинаковых комплектов источников информации: письма от пенсионеров, крупных поставщиков и крупных покупателей продукции, вырезки из газет и журналов, буклеты с выставок-продаж и т.д.

Для каждого текста указывается автор (гражданин-пенсионер, директор фирмы — посредника, название газеты или журнала и т.д.), сам текст небольшой по объему. Тексты обязательно должны содержать противоречивую информацию.

Цель задания. Поставить ученика-секретаря перед необходимостью сформировать критерии, по которым надо отобрать информацию в соответствии с заданной потребностью. Схематично действия ученика можно представить так:



Группа поддержки должна провести непосредственный отбор информации в соответствии с выработанными критериями.

Проведение занятия. Класс делится на 3 группы. В каждой группе выбирается ученик, который собственно и будет секретарем. Остальные учащиеся — группа поддержки. Каждому ученику-секретарю и каждой группе поддержки выдается комплект источников информации. Секретарю дается 5 минут для выполнения задания: он должен на листе бумаги перечислить критерии отбора информации. Затем (в течение 5 минут) ученик-секретарь и группа поддержки по этим критериям независимо друг от друга отбирают материалы. Далее в совместном обсуждении (5–10 минут) группа вырабатывает общее решение, какие документы и почему надо отобрать для доклада директора. Затем ученик-секретарь защищает решение своей группы (5 минут каждому участнику).

Поведение учителя. После окончания выступления всех команд учитель благодарит участников игры. Спрашивает их мнение о проведенном занятии: что понравилось, что нет. Учителю запрещается оценивать предложенные решения. Можно сделать замечания по соблюдению протокола игры, по поведению отдельных участников. Смысл запрета на оценивание состоит в том, что учащиеся, не получив “правильного” и “единственно верного решения” из уст учителя, будут продолжать обсуждать эту игру (дома с родителями, с товарищами и т.д.). И только на следующем уроке надо обсудить решение этой проблемы.

Достижимые результаты. Ученик-секретарь попадает в ситуацию, которая в психологии называется *ситуация напряженной потребности*. Он не может подвести своих товарищей (не хочет выглядеть плохо в их мнении), поэтому за короткий отрезок времени должен попытаться решить поставленную задачу. Далее его результат будут (и, может быть, в жестких выражениях) обсуждать в группе поддержки при отборе информации. Может так получиться, что

ученик-секретарь и группа поддержки по одним и тем же критериям отберут разные материалы. Затем они совместно будут обсуждать полученные результаты. Происходит самообучение, анализ чужого опыта, воспитывается критичное отношение к чужому мнению, воспитывается умение отстаивать свое решение.

Выбор ресурсов согласно выработанным или указанным критериям

Для выбора ресурсов учащиеся должны уметь “знакомиться” с информационным источником. Это знакомство состоит из следующих шагов (причем это касается и информации, найденной в Интернете, и источников на бумажных носителях — книг, газет и т.д.):

1) посмотреть, кто является автором данного ресурса: тогда можно говорить об объективности или субъективности информации, достоверности или нет, о ее полноте;

2) посмотреть время создания (выпуска из печати) и последнего обновления ресурса: можно говорить об актуальности информации;

3) полезность — как правило, достаточно прочесть краткую аннотацию книги или соответствующий раздел интернет-ресурса;

4) важно посмотреть название издательства для книг и журналов и URL-адрес web-сайта, на котором расположен интернет-ресурс (достоверность информации, объективность, полнота).

Умение остановить поиск

Это трудноформализуемое умение, но очень важное. Очевидно, что нельзя подбирать бесконечно много информации по одному и тому же вопросу. Критерием останова поиска информации является субъективное ощущение достижения поставленной цели. Достижение цели может фиксироваться разными способами, например, отобрано заданное количество источников информации, или набрано предельное количество информации, или получен способ решения задачи и т.д.

Пример учебного задания “Выброс серной кислоты”

Время выполнения задания — 5 минут. Задание направлено на проверку сформированности умения вырабатывать критерии отбора информации в соответствии с потребностью.

Полный сценарий (его ученик видит на первом экране)

Вы готовите оперативную справку (достоверную и фактическую информацию) о состоянии экологической среды Луганской области. Вам известно, что в Луганской области,

Пресс-служба завода “Заря”	Последствия аварии не ухудшили экологической обстановки района. Были отмечены даже некоторые положительные ее последствия. Например, в близлежащих сельскохозяйственных угодьях погибли вредоносные насекомые на полях...
Агентство “Гринпис-Информ”	Превышение ПДК вредных веществ в воздухе г. Рубежное не редкость. Данный завод несет экологическую угрозу всей области, его необходимо закрыть, и последнее происшествие тому подтверждение...
Федеральная служба по гидрометеорологии и мониторингу окружающей среды	Констатирует в г. Рубежное превышение ПДК серной кислоты в три раза...
Газета “Авангард” (местная)	Экологическая обстановка в Луганске остается напряженной. Ситуация усугубилась в результате произошедшего выброса паров серной кислоты на химическом заводе “Заря”...

в районе г. Рубежное, расположен химический завод “Заря”, который производит товары широкого потребления из пластмасс и полиэтиленовой пленки. На днях в результате производственной аварии произошел выброс в атмосферу около 100 литров серной кислоты. Образовавшееся облако серной кислоты можно было наблюдать над г. Рубежное невооруженным глазом. Вы должны учесть эту информацию при подготовке справки.

После прочтения сценария ученик переходит к следующему экрану, состоящему из двух частей: “Краткий сценарий” и “Рабочее поле”.

Краткий сценарий

На ленте новостей вы видите 4 информационных сообщения, описывающих ситуацию в пострадавшем от химического выброса серной кислоты районе. Выберите несколько сообщений, которые вы включите в свою справку.

Рабочее поле (на новостной ленте выбор нужного сообщения осуществляется кликом мышки в окошке около этого сообщения). См. таблицу сверху страницы.

Критерии оценивания этого задания

Уровень	Выполненные действия
Высокий	Выбраны 3-й и 4-й источники
Средний	Выбран или 3-й, или 4-й источник
Низкий	Выбраны источники 1 или 2 по отдельности или вместе, или в любой комбинации с источниками 3 и 4

Замечание. Учащийся должен выработать критерий оценки информации в соответствии с заданной потребностью. Его потребность (цель) — составить оперативную сводку о состоянии экологической среды Луганской области. Критериями выбора информации являются: незаинтересованность источника информации, а также достоверность и фактическая ценность информации (в сценарии дана подсказка: необходимо выбрать достоверную и фактическую информацию). Начинаем источники анализировать.

- Источник — пресс-служба завода “Заря”. Скорее всего это заинтересованный источник. Если прочитать информацию от этого источника, то школьник убеждается, что данный источник выбирать не следует;

- Агентство “Гринпис-Информ”. В представленной им информации нет ни одного факта (нет дат, нет конкретных цифр и т.д.), кроме того, информация о том, что “завод несет экологи-

ческую угрозу всей области”, недостоверна, она ничем не подкреплена. Этот источник информации также не следует выбирать, т.к. он не удовлетворяет выбранным критериям;

- Федеральная служба по гидрометеорологии и мониторингу окружающей среды. Во-первых, это незаинтересованный источник информации, во-вторых, им переставлена фактическая информация (превышение ПДК в 3 раза). Данный источник информации следует выбрать;

- Газета “Авангард” (местная). Представлена достоверная информация. Данный источник информации следует выбрать.

Вопросы и задания

1. В чем суть контекстного обучения?
2. Что является обучающей единицей контекстного обучения?
3. Какие активные методы обучения целесообразно использовать при формировании ИКТ-компетентности школьников?

Литература

1. Бешенков С.А., Ракитина Е.А. Информатика. Систематический курс. Учебник для 10-го класса. М.: Лаборатория Базовых Знаний, 2001.
2. Вербицкий А.А. Контекстное обучение и становление новой образовательной парадигмы. Жуковский: МИМ ЛИНК, 2000, 41 с. (Научные труды, выпуск 2).
3. Лаврентьев Г.В., Лаврентьева Н.Б., Неудахина Н.А. Инновационные обучающие технологии в профессиональной подготовке специалистов (часть 2). (http://www/asu.ru/cppkp/index.files/ucheb.files/innov/part2/ch1/glava_1_1.html).
4. Вербицкий А.А. Активное обучение в высшей школе: контекстный подход: Методическое пособие. М.: Высшая школа, 1991.
5. Гершунский Б.С. Педагогическая прогностика: Методология, теория, практика. Киев: Вища шк., 1986.
6. Сквирский В.Я. Методические указания по разработке структуры учебной информации. М.: Изд-во МАДИ, 1980.
7. Фалина И.Н., Мохова М.Н. Использование активных методов обучения на уроках информатики. Информатика № 9/2006.
8. Лаврентьев Г.В., Лаврентьева Н.Б., Неудахина Н.А. Инновационные обучающие технологии в профессиональной подготовке специалистов. Барнаул: Изд-во Алтайского государственного университета, 2002.

Гении или чудаки?

В.П. АРСЛАНЬЯН

Существует миф, что все гении и таланты — чудаки или даже сумасшедшие. Действительно, гении нередко так погружены в сферу своих научных интересов, что их мало интересуют бытовые вопросы. Круг их интересов сужен до одной достаточно узкой проблемы. Со стороны они привлекают внимание необычностью, чужаковатостью поведения, парадоксальностью мышления. Но из того, что все гении — чудаки, не следует, что все чудаки — гении. Да и утверждение “все гении — чудаки” далеко от того, чтобы быть истинным.

Каковы факторы развития гениальности, одаренности? Безусловно, на первом месте стоят задатки, т.е. некоторые врожденные предпосылки, на основе которых развиваются способности. Ученые до сих пор спорят и не пришли к единому мнению в вопросе о том, что же представляют собой эти предпосылки, с какими задатками человек рождается, а что развивается уже в процессе обучения. Например, всем известен факт, что великий композитор Моцарт написал свою первую оперу в 4 года и в раннем возрасте выступал с концертами перед императором и широкой публикой. Но раскрылся бы этот талант, если бы отец Моцарта не был музыкантом, если бы он очень много времени не отдавал занятиям со своим сыном, не развивал в нем эти способности? Поэтому можно с уверенностью сказать, что раскрыть и развить врожденные задатки могут взрослые, опытные, талантливые, творческие педагоги.

Задатки чаще всего дают о себе знать уже в дошкольном детстве. Например, ученые отмечают, что дети, способные к математике, уже в раннем возрасте проявляют большой интерес к вычислениям, измерениям, взвешиванию или упорядочиванию предметов; проявляют необычное для своего возраста понимание математических отношений; демонстрируют легкость в восприятии и запоминании математических символов; с легкостью выполняют простейшие операции сложения и вычитания; разбираются в измерении времени (часы, календари); часто применяют математические навыки и понятия в процессе занятий, не имеющих отношения к математике.

Естественно, что и в период обучения в школе учеников, проявляющих

особые способности к математике, видно. Они, как правило, легче и быстрее сверстников усваивают новые темы, предлагают новые, нестандартные решения задач, проявляют повышенный интерес к урокам математики, предпочитают решать нестандартные задачи, а не “отрабатывать навыки” по закреплению какого-либо способа решения и т.д.

Отметим общие *особенности познавательной сферы одаренных детей*.

- Любопытство в исследовании окружающего мира уже на ранних этапах развития. У одаренных и талантливых детей биохимическая и электрическая активность мозга повышена, их мозг обладает повышенной способностью “переваривать” интеллектуальную пищу.

- В раннем возрасте таких детей отличает способность проследить причинно-следственные связи между предметами и явлениями и делать соответствующие выводы.

- Талантливые дети видят неожиданные, непривычные связи между событиями. Эти связи нередко составляют основу творчества и изобретательства.

- Обладая большой потребностью в выполнении интеллектуальных и познавательных задач, такие дети не терпят, когда им навязывают готовый ответ.

- Одаренные дети обычно обладают отличной памятью, которая базируется на ранней речи и абстрактном мышлении.

- Их отличают большой словарный запас и способность строить сложные синтаксические конструкции, умение формулировать вопросы.

- Талантливые дети легко справляются со сложными и долгосрочными заданиями, при этом познавательный интерес у них не угасает.

- Одаренного ребенка также отличают и повышенная концентрация внимания на чем-либо, упорство в достижении результата в сфере, которая ему интересна, стремление доводить все до полного совершенства.

- У таких детей великолепное чувство юмора, поскольку их воображение так живо, что они видят вокруг много несурзадного и смешного.

Все ли из перечисленных выше особенностей присутствуют у талантливого ребенка? Нет, не все. Но и сочетание некоторых из этих признаков ни в

каком случае не свидетельствует об одаренности. Например, ученик любит заниматься только тем, что ему интересно, и обладает повышенным чувством справедливости — это не свидетельствует о том, что перед нами одаренный ребенок.

Для одаренных детей характерно и определенное своеобразие личностной сферы, которое может способствовать обучению, а может и создавать проблемы, чаще всего социального характера. Почему? Одна из причин в том, что одаренные дети часто переходят из класса в класс, минуя какие-то ступени, например, из 2-го класса в 4-й, из 5-го в 7-й. Но при опережающем интеллектуальном развитии их личностное, эмоциональное, социальное развитие остается на уровне сверстников. Например, важным этапом в формировании личности является опыт юношеской дружбы, построения отношений с представителями противоположного пола, навыки решения проблемных ситуаций, и если подросток минует эти стадии развития в соответствующий возрастной период, то это неминуемо скажется на его дальнейшей адаптации в социуме.

Перечислим основные *особенности развития личности одаренных детей*:

- Одаренные дети обнаруживают обостренное чувство справедливости и достаточно остро могут реагировать на несправедливость окружающего мира.

- Они предъявляют высокие требования к себе и окружающим.

- Талантливые дети обладают критичностью и независимостью мышления и суждений, инициативой и настойчивостью. Это может вызывать раздражение у педагогов, так как они вступают в дискуссии с учителем, доказывают свою точку зрения, что может восприниматься учителем как желание ученика “подорвать” его авторитет.

- Им недостает эмоционального баланса; в раннем возрасте одаренные дети нетерпеливы и порывисты.

- Порой для них характерны преувеличенные страхи и повышенная уязвимость. Они чувствительны к неречевым сигналам окружающих.

- Нередко у одаренных детей развивается негативное самовосприятие, воз-

Примерные ответы на профильные билеты

Е.А. ЕРЕМИН, А.П. ШЕСТАКОВ,
г. Пермь

Продолжение. См. № 19, 20/2006

Билет № 5

1. Язык программирования. Типы данных. Реализация основных алгоритмических структур на языке программирования. Основные этапы разработки программ

Язык программирования — это набор правил для описания алгоритмов решения задачи с помощью ЭВМ.

Как известно (см. билет № 12), одним из базовых принципов архитектуры современных компьютеров до сих пор остается двоичный характер *любой* используемой информации, причем программа обработки также представляет собой двоичный код. Тем не менее программирование в двоичных кодах — занятие необычайно утомительное и требующее глубокого знания деталей архитектуры компьютера. Для облегчения данного процесса и предназначены языки программирования, используя которые человеку проще описать алгоритм решения задачи. Переход от языковых конструкций к машинным командам осуществляет специальная программа — *транслятор* языка.

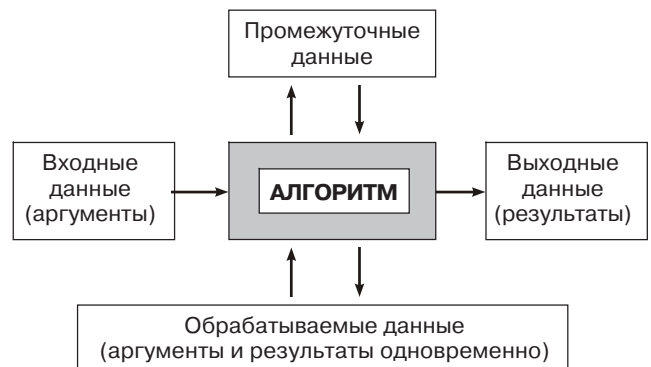
В соответствии с устройством языка (точнее говоря, со степенью его близости машине или человеку) различают языки низкого и высокого уровня. **Языки низкого уровня**, называемые еще машинными (или машинно-ориентированными) языками, — это языки, которые компьютер воспринимает непосредственно, т.е. языки машинных команд данной модели компьютера. **Языки высокого уровня**, напротив, ближе к естественному (человеческому); они не зависят от конкретного типа машины. Языком низкого уровня является ассемблер, а языков высокого уровня существует множество: Фортран, Алгол, Бейсик, Паскаль, Си, Ада, Пролог, Лисп, Ява и др.

Один и тот же язык программирования может быть реализован по-разному даже на одном и том же компьютере (пример: GW Basic и QBasic). В синтаксисе различных реализаций допускаются некоторые второстепенные отличия.

Одна из книг выдающегося специалиста по разработке языков программирования Никлауса Вирта (кстати, создателя языков Паскаль, Модула, Эйлер и Оберон) очень емко и глубоко названа “Алгоритмы + структуры данных = программы”. Таким образом, в теории программирования налицо две взаимосвязанные составляющие процесса решения за-

дачи: собственно **данные** и инструкции по их обработке, т.е. **алгоритм**.

Рассмотрение начнем с первой составляющей — **данных**. Одно из главных свойств алгоритма состоит в том, что он по определенным правилам преобразует исходные (входные) данные в выходные (чаще говорят — в результат). При этом в процессе выполнения алгоритма может потребоваться создать некоторые рабочие (промежуточные) данные, которые будут необходимы только в ходе обработки, а после ее завершения потеряют свое значение. Кроме того, для некоторых алгоритмов аргумент может одновременно являться и результатом (например: увеличить все элементы массива вдвое), что приводит к существованию еще одной разновидности данных. Специального термина для них в учебной литературе нет, поэтому приведенное на рисунке название несколько условно.



Описанное нами четкое функциональное разделение данных имеется в любом школьном учебнике, начиная с самого первого [1]. Перечисленные выше категории у А.Г. Кушниренко в [2] названы *видами* величин. Как определено в [2], “вид величины показывает ее информационную роль в алгоритме”. В конкретном языке программирования каждой величине соответствует своя *переменная*.

Помимо вида, каждая величина в алгоритме имеет свой *тип*. Прочитав еще раз учебник [2]: “Тип величины показывает, какие значения может

принимать величина и какие операции можно с ней выполнять”.

Кратко перечислим основные типы данных, использующихся в алгоритмических языках. Для вычислений используются различные **числовые** типы данных. Этот тип возник в ЭВМ самым первым, поэтому неудивительно, что он включает в себя достаточное количество разновидностей. Прежде всего назовем вещественные и целые числа. Последние могут быть как содержащими знак, так и беззнаковыми. В качестве примера вспомним в Турбо Паскале типы `integer` (значения от $-32\,768$ до $32\,767$) и `word` (от 0 до $65\,535$). Кроме того, конкретные реализации языков программирования чаще всего содержат несколько разновидностей целых и вещественных данных, что связано с различным объемом памяти, выделяемым для них. В качестве самого простого примера назовем вещественные числа обычной и двойной точности в языке Basic. Наконец, для иллюстрации многообразия числовых данных упомянем введенный в Delphi тип `currency` (валюта), специально предназначенный для максимально точного хранения значений денежных сумм и вычисления процентов от них.

Еще одна категория данных, которая часто используется в программировании, это **символьные** величины. В зависимости от конкретного языка программирования такие данные могут иметь некоторые не очень существенные особенности, но все они служат для хранения и организации обработки текстовой информации.

Наконец, очень важным типом данных для построения программ со сложной структурой являются **логические** величины. Их часто называют булевскими в честь ирландского математика Д.Буля, который был основоположником алгебры логики. Хотя логические переменные имеют всего два значения — “ложь” и “истина”, без них в языках программирования не было бы ни полноценной развилки, ни цикла.

В современных реализациях языков программирования появился особый тип данных — **вариантный**, который может принимать *любые* (числовые, символьные и т.д.) значения, причем тип текущего значения также хранится в самой переменной и может быть запрошен программистом. Подобный тип данных лежит в основе языка VBA (*Visual Basic for Applications*); на его базе реализуется общий формат ячеек электронной таблицы Excel.

До сих пор мы говорили о том, какие значения могут принимать те или иные типы данных. Не стоит забывать и о второй части определения, а именно о том, какие действия можно совершать над каждым из них. Поскольку это довольно простая часть вопроса, ограничимся лишь парой примеров. Для целых и вещественных типов данных операции

деления различны; они даже по-разному обозначаются: в языке Паскаль “`div`” и “`/`” соответственно. Или другой пример. Если для чисел вывести результат действия $123 + 45$, то получится 168, в то время как для строковых значений “123” + “45” приведет к появлению на экране совсем другой результирующей строки: “12345”. Здесь, наоборот, операции с одинаковым обозначением производят над различными типами данных разные действия.

Все рассмотренные выше типы данных являются **простыми**. Языки программирования позволяют сконструировать и **сложные** типы данных, причем для этого можно объединять как несколько простых, так и другие сложные данные. Самым известным сложным типом является массив, объединяющий в себе набор данных одного типа, например, массив из целых чисел или массив из логических величин. Кстати, конструкция массив из массивов также является вполне допустимой. Более подробно сложные типы данных будут рассмотрены в *билете* № 7.

На стандартный простой тип можно непосредственно ссылаться, а для сложного типа требуется предварительно описывать его структуру. Отметим попутно, что типы данных не всегда определены заранее. Во многих языках существует возможность определения собственного типа данных.

Зачем нужны типы данных? “Прежде всего они указывают, как кодировать данные в ЭВМ при их вводе и трансляции программ и как декодировать данные при их выводе и исполнении программ. ...Благодаря этому кодированию становится возможным контроль над многими ошибками в программе. Зная тип переменной, транслятор может обнаружить, что переменной присваивается недопустимое значение, что к ней применяется неправильная операция... и выдать программисту сообщение об ошибке”. [3]

Для обработки данных используются различные **алгоритмы**. Любой алгоритм состоит из последовательности команд, которые часто по-другому называются **операторами**. Набор операторов в различных языках программирования, как ни странно, относительно невелик и примерно одинаков. Основу любого алгоритмического языка составляет оператор **присваивания**. Именно он позволяет изменять значения данных программы и тем самым получать результат решения задачи. Как метко сказано в [4], “удивительно, что в обычных языках программирования есть только один оператор, который фактически что-то делает, — оператор присваивания. Все другие операторы, такие, как условные операторы и вызовы процедур, существуют только для того, чтобы управлять последовательностью выполнения операторов присваивания”.

Чтобы получить простейшую законченную линейную программу, к операторам присваивания надо

добавить какие-то средства ввода-вывода значений переменных. Ввод и вывод информации для различных языков отличаются наиболее сильно. Так, например, в языке Basic для ввода и вывода существуют особые операторы `input` и `print`, а в Паскале и Си для этой цели предусмотрены специальные системные процедуры.

Для организации разветвляющихся и циклических участков алгоритма в языках программирования предусмотрены специальные операторы, которые обычно называются **управляющими**. К ним относятся имеющиеся в любом алгоритмическом языке **условный** оператор (дополненный, как правило, некоторой разновидностью — конструкцией *выбора*) и несколько видов **циклов**. Помимо этого стандартного обязательного набора, могут существовать и другие операторы. В частности, отметим оператор безусловного перехода `goto`, который хотя и есть практически в любом языке, но фактически является “запрещенным”: дело в том, что его применение существенно усложняет чтение программы, тогда как перечисленные выше стандартные конструкции вполне позволяют без него обойтись.

В теории программирования строго доказывается, что для реализации *любого* алгоритма достаточно всего трех стандартных структур: следования (операторы выполняются в строгом соответствии с порядком написания), ветвления и цикла. Рассмотрим в качестве примера реализацию этих базовых структур в языке Паскаль.

Примечание. Если в школе изучался другой язык программирования, то, безусловно, надо рассматривать его; мы надеемся, что замена синтаксиса записи в этом случае не вызовет особых затруднений.

Условный оператор, обеспечивающий реализацию ветвления в алгоритмах, в рассматриваемом языке программирования имеет вид:

```
IF <логическое выражение> THEN <оператор1>
ELSE <оператор2>
```

— причем ветвь с `ELSE` не является обязательной. Рассказывая о структуре условного оператора, стоит подчеркнуть два момента. Во-первых, логическое выражение совсем не обязательно представляет собой математическое неравенство типа $x > 0$ — оно вполне может содержать еще и булевские переменные, а также объединять несколько логических величин с помощью операций `AND`, `OR` и `NOT`. Во-вторых, согласно синтаксису языка в качестве операторов 1 и 2 может использоваться только один оператор. Когда для программиста этого недостаточно, разрешается создавать так называемый *составной* оператор, заключая необходимое количество операторов внутри служебных слов `BEGIN...END`.

Приведем несколько конкретных вариантов записи условного оператора.

```
IF x = 0 THEN WRITELN('Нулевое значение!')
ELSE y := 1/x;
IF x <> 0 THEN y := 1/x
ELSE WRITELN('Нулевое значение!');
IF n1 < n2 THEN
    BEGIN
        n1 := n1 + 1;
        n2 := n2 - 1;
    END;
IF control AND (x1 > 0) AND (x2 > 0)
    AND (dx > 0) THEN
    WRITELN('Все параметры положительны')
```

(в последнем случае подразумевается, что переменная `control` является логической).

Рассмотрим теперь операторы цикла. Имеется их три основных вида: *с предусловием*, *с постусловием* и *с параметром*. Соответствующий синтаксис перечисленных разновидностей циклов приведен ниже.

```
WHILE <логическое выражение> DO <оператор>
REPEAT <операторы> UNTIL <логическое выражение>
FOR <параметр> := <начальное значение>
    TO <конечное значение> DO <оператор>
```

Правила записи логических выражений для циклов такие же самые, что и описанные выше для условного оператора. Заметим, что пара операторов `REPEAT...UNTIL` по смыслу эквивалентна “операторным скобкам” `BEGIN...END`, так что внутри данной конструкции допускается запись сразу нескольких операторов. Данная особенность цикла с постусловием является исключением из общего правила.

Цикл с параметром организует автоматическое изменение некоторой переменной (параметра цикла). Такая конструкция особенно удачно подходит для последовательного перебора всех значений индексов, что существенно облегчает работу с массивами.

Несколько примеров операторов цикла.

```
WHILE dy > eps DO
    BEGIN
        n := n + 1;
        dy := 1/(n * n);
        y := y + dy;
    END;
REPEAT READLN(s);
    res := res + s;
UNTIL s = '.';
FOR i := 1 TO n - 1 DO READLN(m[i]);
FOR s := 'A' TO 'Z' DO WRITE(s);
```

В других языках программирования правила записи операторов управления во многом аналогичны.

Чаще всего при написании реальной прикладной программы (здесь речь не идет об учебных упражнениях типа “в массиве найти максимальный элемент”) требуется не только программирование. Дело в том, что взятая из жизни задача обычно не пред-

ставлена в форме, немедленно готовой к решению на компьютере: приходится сначала получать ее, выбирая при этом подходящий метод. Кроме того, когда программа будет написана, введена и отлажена, потребуется провести анализ полученных числовых результатов и сделать выводы относительно исходной задачи.

В итоге можно выделить следующие основные этапы разработки программ.

1. *Постановка задачи* (выяснение наиболее существенных ее параметров и закономерностей, влияющих на изучаемую ситуацию; формулировка целей решения и определение необходимых для этого данных).

2. *Математическая формализация* (запись основных закономерностей в математической форме: в виде уравнений, соотношений, связей, условий).

3. *Выбор компьютерного метода* (обычно данный этап требуется, поскольку получившаяся математическая задача нуждается в преобразовании в эквивалентную форму, пригодную для обработки на компьютере, — например, дифференциальные уравнения приводятся к алгебраическим формулам, а для статистического моделирования требуется реализация датчика случайных чисел).

4. *Построение алгоритма*.

5. *Составление программы на языке программирования*.

6. *Отладка и тестирование программы* (устранение синтаксических и логических ошибок, анализ полученных результатов на предмет согласованности с имеющимися в науке закономерностями и ранее надежно установленными результатами; если необходимо — “ручная” проверка результатов для простейших случаев).

7. *Проведение расчетов и анализ полученных результатов* (все предшествующие этапы выполнялись с целью получить сведения о некотором реальном предмете или явлении, а не просто провести вычисления).

Примечание. Приведенная схема первоначально возникла для решения чисто вычислительных задач, но, в силу своей общности, она применима и к другим задачам обработки информации. Например, создание информационной системы с незначительными оговорками вполне в нее укладывается.

Проиллюстрируем описанную схему на примере некоторой конкретной задачи. Обратимся, например, к упражнению 7 на странице 77 в учебнике [1], которое формулируется следующим образом.

Найдите распределение температуры во внутренних точках тонкого однородного стержня, считая, что стержень разбит на 5 отрезков и что на концах стержня температура равна 10 и 20°. Начните с температуры 15° на всех отрезках. Выполните 5 шагов вычислений. Так же, как в случае плас-

тины, исходите из предположения, что температура каждого отрезка равна среднему арифметическому температур соседних отрезков.

Примечание для учителей. Подробное рассмотрение решения конкретной физической задачи авторы считают дополнительным материалом, требовать который с учеников во время экзамена нет необходимости. С другой стороны, рассмотренный пример кажется весьма удачным сочетанием простоты с наличием в решении всех перечисленных этапов (в большинстве учебных задач те или иные этапы “выпадают”).

1. Рассматривается задача распределения температуры в тонком однородном стержне. Последнее обстоятельство позволяет считать, что искомое распределение температуры является функцией только одной координаты. Теплоотдачу окружающей среде вдоль стержня и остальные тепловые факторы предполагаем пренебрежимо малыми. Параметрами задачи являются два значения температуры на концах стержня.

2. Процесс описывается одномерным уравнением теплопроводности с граничными условиями на концах стержня. Поскольку в физике его принято записывать в дифференциальном виде, то школьникам его представить затруднительно. К счастью, существует “облегченный” вариант, который даже с точки зрения физиков считается корректным: необходимо вместо дифференциального соотношения (для бесконечно малых величин) записать уравнение задачи в виде конечно-разностного аналога (для конечных, но малых разностей). Фактически это означает переход к следующему этапу. Поскольку данный этап в учебной задаче оказывается “свернутым”, в учебниках (см., например, [5]) его часто объединяют со следующим; подобный методический прием вполне допустим.

3. Задача формулируется в виде, пригодном для обработки на компьютере. Стержень разбивается на несколько отрезков, для каждого из которых записывается уравнение теплового баланса (подробности см. в [1]). В результате получается, что температура каждого отрезка равняется среднему арифметическому всех (в одномерном случае двух!) соседних, что описывается простой формулой

$$T_i = (T_{i-1} + T_{i+1}) / 2$$

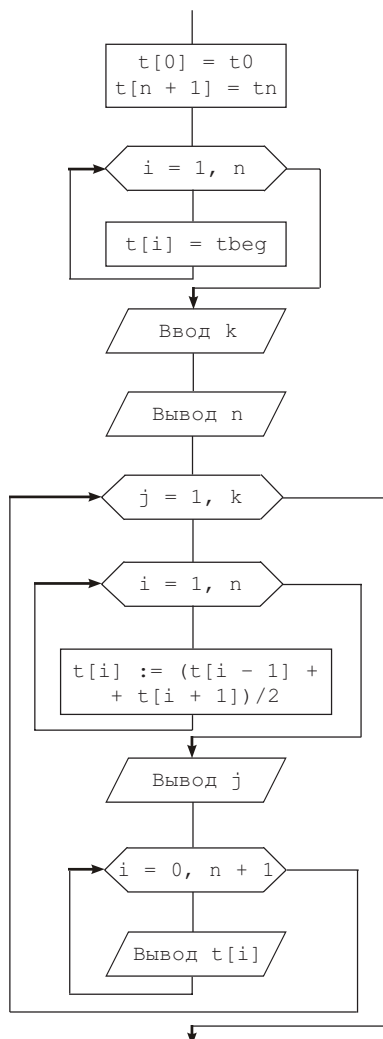
где i — это номер отрезка, изменяющийся от 1 до некоторого значения n . Последнее является “вычислительным” параметром задачи: в исходной формулировке его нет, но, поскольку результаты от него зависят, это влияние также необходимо проанализировать.

Фактически для значений T_i мы получили систему из n связанных алгебраических уравнений, которую при больших n решить затруднительно. Поэтому решение предполагается производить итерационным методом, суть которого заключается в следую-

щем. В начальный момент температура по всей стержню полагается однородной. Затем по выведенной формуле производится вычисление новых значений. Полученные значения температуры снова берутся в качестве исходных данных для дальнейших вычислений. Как правило, описанная процедура после некоторого количества повторений (итераций) k приводит к тому, что вычисляемые значения с практической точностью перестают изменяться (математики говорят, что метод сходится). Подобные итерационные методы используются в научных расчетах весьма часто.

Заметим, что при практической реализации метода удобно ввести два “фиктивных” граничных отрезка с номерами 0 и $n + 1$, в которые записать постоянные значения, взятые из условия задачи. Расчетные формулы применяются только для внутренних узлов с номерами от 1 до n .

4. Далее необходимо разработать алгоритм расчетов. Его возможная реализация показана в виде блок-схемы на следующем рисунке.



5. Написать программу по готовой блок-схеме, как правило, особого труда уже не представляет. Вот как, например, она может выглядеть на языке Паскаль.

```

PROGRAM temp(INPUT,OUTPUT);
CONST n = 5; {количество отрезков}
      t0 = 10; tn = 20; tbeg = 15;
VAR t: ARRAY [0..n + 1] OF REAL;
      i, j, k: INTEGER;
BEGIN t[0] := t0; t[n + 1] := tn;
      FOR i := 1 TO n DO t[i] := tbeg;
      READLN(k); {количество итераций}
      WRITELN('n=',n);
      FOR j := 1 TO k DO
        BEGIN FOR i := 1 TO n DO
              t[i] := (t[i - 1] + t[i + 1])/2;
              WRITE(j:2);
              FOR i := 0 TO n + 1 DO
                WRITE(t[i]:9:4);
                WRITELN
              END
            END.
  
```

6. Приведем теперь несколько рекомендаций по отладке и тестированию программы. Как только очевидные ошибки синтаксиса будут устранены и программа начнет считать, выдавая на экран необходимое количество чисел, можно заняться проверкой правильности проводимых расчетов. Для этого необходимо тщательно проверить выведенные на экран значения температуры: на границах стержня она должна равняться в точности 10 и 20 градусам, а значения во внутренних точках должны заключаться в интервале между ними. Интуитивно кажется (и расчет подтверждает это), что значения t_i должны нарастать слева направо. Для проверки правильности вычислений можно без компьютера провести 1–2 итерации при небольших значениях n^1 . Заметим, что для отладки вполне годятся даже весьма неразумные с физической точки зрения значения вроде $n = 1$.

Когда все числа будут тщательно проверены, появится некоторая уверенность в том, что получаются правильные результаты. Остается дополнительно выяснить, как на них влияют “счетные” параметры n и k . Их следует подбирать не очень большими, но такими, чтобы изменение значения в 1,5–2 раза меняло результирующие температуры в пределах практически требуемой точности (1% для учебной задачи вполне достаточно).

7. Наконец, все вычислительные тесты позади и можно приступить к анализу физических результа-

¹ Для $n = 2$, в частности, окончательные значения t_1 и t_2 легко вычисляются путем решения системы из двух уравнений для них, что позволяет провести еще один тест программы.

тов решения задачи. Построим график температуры вдоль стержня и убедимся, что он представляет собой прямую. Поменяем граничные температуры и проверим, как их значения влияют на температурное распределение.

Подчеркнем, что без подобного анализа, о котором идет речь в данном пункте, решение задачи на компьютере особого смысла не имеет.

Если читателей интересует реализация описанной общей схемы на более содержательных физических примерах, советуем заглянуть в подшивку газеты и в серии “Жаркое лето” этого года найти публикацию [6].

Литература

1. Основы информатики и вычислительной техники: Пробное учебное пособие для средних учебных заведений. Ч. 1 / А.П. Ершов, В.М. Монахов,

С.А. Бешенков и др.; под ред. А.П. Ершова и В.М. Монахова. М.: Просвещение, 1985, 96 с.

2. Кушниренко А.Г., Лебедев Г.В., Зайдельман Я.Н. Информатика. 7–9-е классы: Учебник для общеобразовательных учебных заведений. М.: Дрофа, 2000, 336 с.

3. Информатика: Энциклопедический словарь для начинающих / Сост. Д.А. Поспелов. М.: Педагогика-Пресс, 1994, 352 с.

4. Бен-Ари М. Языки программирования. Практический сравнительный анализ. М.: Мир, 2000, 366 с.

5. Семакин И.Г. Информатика. Базовый курс. 7–9-е классы / И.Г. Семакин, Л.А. Залогова, С.В. Русаков, Л.В. Шестакова. М.: БИНОМ, 2004, 390 с.

6. Бирих Р.В., Еремин Е.А., Чернатыйский В.И. Компьютерные модели в школьном курсе физики. Информатика, 2006, № 14, с. 3–45; № 15, с. 3–13.

2. Сформировать и выполнить запрос к готовой базе данных для поиска группы записей по заданным критериям

В качестве примера рассмотрим поиск информации в базе данных в формате MS Access. База данных содержит следующие поля.

Поле	Тип данных	Описание
Номер	Счетчик	
Фамилия	Текстовый	
Имя	Текстовый	
Дата	Числовой	Дата рождения
Пол (м)	Логический	Пол мужской? (да, нет)
Улица	Текстовый	
Дом	Числовой	
Квартира	Числовой	
Класс	Числовой	
Группа	Текстовый	Группа здоровья по физкультуре
Хобби	Текстовый	
Глаза	Текстовый	Цвет глаз

Запросы на поиск информации в базе данных

1. Кто родился весной?
2. Кто учится в одиннадцатом классе?
3. Кто не занимается тяжелой атлетикой?
4. Кто не определил свое хобби?
5. Кто из мальчиков живет на улице Чердынской?
6. У кого карие глаза и этот человек не умеет вязать?

Примерная информация для заполнения базы данных приведена ниже.

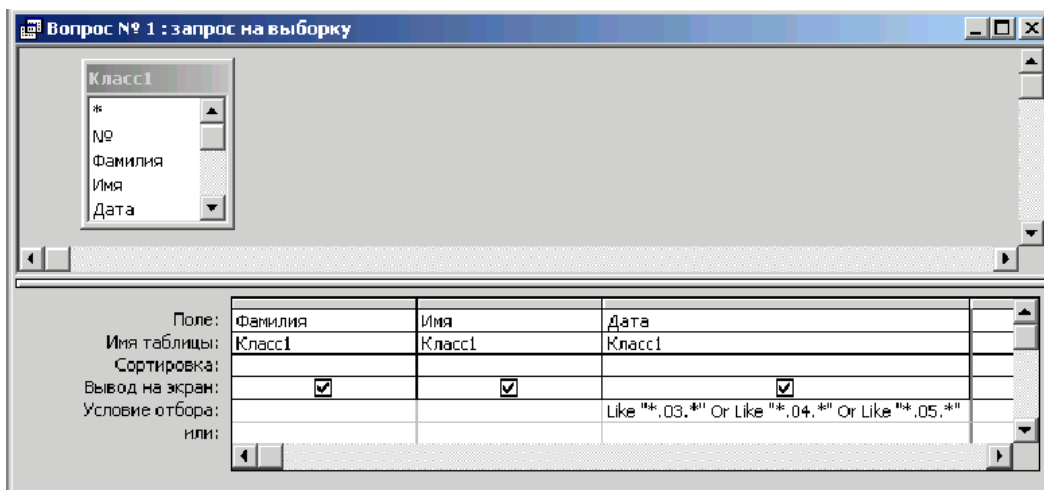
№	Фамилия	Имя	Дата	Пол	Улица	Дом	Квартира	Класс	Группа	Хобби	Цвет глаз
1	Суханов	Сергей	16.02.85	Да	Чердынская	23	74	10	Основная	Тяжелая атлетика	Зеленые
2	Пирогов	Юрий	5.12.87	Да	Куйбышева	6	31	8	Основная	Футбол	Голубые
3	Лебедева	Света	16.06.89	Нет	Пушкина	37	65	6	Специальная	Вязание	Карие
4	Голдобин	Сергей	23.05.92	Да	Леонова	12	10	3	Основная	Льжи	Голубые
5	Ельшина	Наташа	24.05.86	Нет	Чердынская	37	48	9	Специальная	Чтение	Серые
6	Суханова	Наташа	20.12.90	Нет	Ленина	12	22	5	Подготовительная	Шитье	Зеленые
7	Петрова	Света	18.04.86	Нет	Пушкина	37	3	9	Основная	Льжи	Серые
8	Горина	Оля	20.12.88	Нет	Свизева	66	99	7	Подготовительная	Аэробика	Карие
9	Попов	Михаил	7.07.91	Да	Леонова	72	6	4	Подготовительная		Голубые
10	Сергеев	Саша	30.11.93	Да	Куйбышева	3	31	2	Основная	Карате	Зеленые
11	Павлова	Елена	13.12.89	Нет	Пушкина	5	6	6	Основная	Аэробика	Карие
12	Емельянова	Наташа	25.05.85	Нет	Попова	40	47	10	Основная	Шитье	Зеленые
13	Евдокимов	Михаил	18.08.88	Да	Чердынская	3	40	7	Основная	Футбол	Зеленые
14	Евсеева	Елена	14.10.86	Нет	Ленина	14	82	9	Основная	Льжи	Серые
15	Суханова	Света	29.07.84	Нет	Куйбышева	37	32	11	Основная	Аэробика	Карие

Для указанных выше запросов логические выражения могут выглядеть так:

1. К весенним месяцам относятся март, апрель и май. Соответствующее логическое выражение запроса:

"Дата" = Like "*.03.*" Or Like "*.04.*" Or Like "*.05.*"

Диалоговое окно для создания запроса в режиме конструктора выглядит в этом случае так:



- "Класс" = 11
- "Хобби" <>"тяжелая атлетика"
- "Хобби" Is Null
- "Пол" = Истина and "Улица" = "Чердынская"
- "Глаза" = "карие" and "Хобби" <> "аэробика"

В данном примере рассматривается однотабличная база данных. Соответственно, запросы в этом случае строятся, хотя и с помощью сложных логических выражений, более простые, чем в случае многотабличной базы данных. Примеры исполь-

зования многотабличных баз данных и задания для формирования данного билета можно найти в элективном курсе информатики И.Г. Семакина, Е.К. Хеннера².

² Семакин И.Г. Информационные системы и модели. Элективный курс: Учебное пособие / И.Г. Семакин, Е.К. Хеннер. М.: БИНОМ. Лаборатория Знаний, 2005, 303 с.: ил.

Семакин И.Г. Информационные системы и модели. Элективный курс: Практикум / И.Г. Семакин, Е.К. Хеннер. М.: БИНОМ. Лаборатория Знаний, 2006, 87 с.: ил.

3. Подсчитать размер текстового файла при заданной кодовой таблице, формате страницы и количестве страниц. Оценить размеры того же текста в других изученных форматах текстовых редакторов

Задача. Используется кодовая таблица CP1251 (Windows Cyrillic). Сколько килобайт будет занимать файл в простом текстовом формате (plain text), если в тексте 300 страниц, на странице 25 строк, а в строке в среднем 45 символов? Оцените объемы того же текста в форматах doc, rtf и в HTML. Как изменится объем файла в этих форматах, если применить элементы форматирования (использовать различные шрифты и начертания)?³

Решение. В заданной системе кодировки один символ занимает 1 байт. Поэтому для ответа на поставленный вопрос достаточно подсчитать количество символов в тексте.

$300 \times 25 \times 45 = 337\,500$ символов, что соответствует 337 500 байтам, или $\approx 329,59$ Кб.

В HTML собственно текстовая информация дополняется служебной информацией, тегами для раз-

метки текста, что в среднем увеличит размер предыдущего файла до ≈ 500 Кб. Добавление дополнительных элементов форматирования (за счет добавления соответствующих тегов) может незначительно увеличить размер на 10–15 Кб⁴.

Оценивая объемы того же текста в форматах doc, rtf, можно лишь предположить, что использование некоторых команд разметки и т.д., присутствующих в этих форматах, приведет к росту размера файла, увеличив его в 2,5–3 раза, т.е. около 1 Мб. При этом дополнительные элементы форматирования несущественно изменят объем. Не следует забывать, что в файлах этих форматов хранится и некоторая служебная информация, объем которой оценить затруднительно.

Что касается формулирования аналогичных заданий для экзаменационного билета, то необходимо основательно продумать корректность формулировок, если предполагается получить четкий и однозначный ответ.

³ Следует признать формулировку задания (как общего, так и этого частного) некорректной, поскольку, говоря об оценке размера файла, необходимо знать о внутренних форматах файлов, а, например, формат .doc является закрытым (открытым является формат файлов WinWord 97).

⁴ Если для добавления новых элементов форматирования используется стиль (в частности, его описание хранится в отдельном файле), то объем файла в формате HTML просто не изменится.



«Портфолио»

Фестиваль исследовательских
и творческих работ учащихся



Издательский дом «Первое сентября» объявляет о проведении в 2006/07 учебном году Второго всероссийского фестиваля «Портфолио» и приглашает принять в нем участие учащихся учреждений начального, среднего и дополнительного образования и их педагогов.

Участвуя в фестивале, учащиеся могут формировать общедоступное портфолио своих работ. Также формируется портфолио педагога, в которое входят работы учащихся, выполненные под его руководством.

Все материалы будут опубликованы. По результатам фестиваля будут изданы: книга — сборник тезисов (описаний) работ и компакт-диски с полными версиями работ. Полные версии работ также публикуются на сайте фестиваля <http://portfolio.1september.ru>, который является одним из разделов сайта Издательского дома «Первое сентября» — самого популярного образовательного ресурса русскоязычного Интернета. **Книги и компакт-диски будут высланы всем участникам. Все ученики и их руководители будут отмечены дипломами.**

Заявки на участие принимаются с 1 июля по 15 декабря 2006 г.

Заявки можно подавать на сайте <http://portfolio.1september.ru> или по почте, используя бланки, публикуемые в газетах.

РАЗДЕЛЫ ФЕСТИВАЛЯ

- Научно-исследовательские работы:
 - Астрономия, космонавтика и авиация
 - Биология
 - География
 - История, археология
 - Искусствоведение
 - Лингвистика
 - Литературоведение
 - Математика
 - Религиоведение
 - Экономика, социология и право
 - Физика
 - Химия
 - Здоровье человека, психология
 - Физкультура и спорт
 - Экология
- Художественное творчество
- Техническое творчество
- Информационные технологии
- Литературное творчество
- Музыкальное творчество
- Краеведение
- В помощь учителю (дополнительный раздел)

Азы информатики. Выходим в Интернет

А.А. ДУВАНОВ,
г. Переславль-Залесский,
kurs@robotland.pereslavl.ru

Продолжение. См. № 17–20/2006

Часть II. Как работает Интернет

Слово к читателю

Уважаемые коллеги!

Свершилось! Наконец, газета объяснила “на пальцах”, как работают компьютерные сети (см. № 17–20)!

Удовольствие получено!

А если так, то теперь давайте разберемся, как работает самая популярная сеть, название которой у всех на устах.

Напоминаю, что эта публикация является рабочей презентацией новой книги автора по курсу “Азы информатики” — “Выходим в Интернет” (8-й класс).

Краткое содержание нового учебника:

- Компьютерные сети;
- Как работает Интернет;
- Электронная почта;
- Сервисы Web;
- Web-дизайн и основы сайтостроения.

Электронные книги-лаборатории по курсу “Азы информатики” можно заказать на сайте Роботландии www.botik.ru/~robot/sale или письмом автору по адресу kurs@robotland.pereslavl.ru.

Бумажные учебники выпускает издательство “БХВ-Петербург” (алгоритм заказа для школ по специально низким ценам приводится на роботландском сайте по адресу www.botik.ru/~robot/ru/bhv.htm).

Подробную презентацию курса “Азы информатики” (с демо-версией) можно скопировать с адреса:

<ftp://ftp.botik.ru/rented/robot/univer/azinf.d.zip>
(6,3 Мб).

Отзывы пользователей можно посмотреть на странице:

www.botik.ru/~robot/sale/reasons.htm

Также можно посмотреть доклад, сделанный автором на конференции “ИТО-2005” (доклад оформлен в виде иллюстрированного гипертекстового приложения):

<ftp://ftp.botik.ru/rented/robot/univer/ito2005.zip>
(1 Мб).

Читальный зал

Текст, который в диалогах Читального зала произносится от имени ЯНЗ, действительно, принадлежит ЯНЗ (Якову Наумовичу Зайдельману) и включен в книгу с его разрешения.

Пакетная передача, коммутация пакетов

— Мне кажется, что вокруг Интернета слишком много шума. — Васина рожица полна сарказма. — Что в нем особенного? Два компьютера соединяются по телефону и вместо речи обмениваются файлами.

Да, удобно! Но о какой информационной революции идет речь? Это обычный телефон! Получается, что паровоз, который возил людей, стал перевозить железные болванки, а все вокруг попадали от восторга!

Петя не сердится на брата. Очень часто Васины “наивные” вопросы заставляют его самого лучше разобраться в предмете обсуждения.

Петя. Нет, Вася. Интернет — это не телефон! Хотя бы потому, что каналами передачи информации в Интернете служат не только телефонные линии.

Вася. Телефония тоже не стоит на месте! Сотовая связь, например, не пользуется традиционными телефонными линиями, но остается всего лишь “продвинутым” телефонным средством. Чем же Интернет принципиально отличается от телефона?

Петя. Попробую объяснить.

Во-первых, в Интернете по телефону и другим каналам связи передаются не файлы, а *пакеты*. Исходное сообщение “нарезается” на части, каждая часть снабжается заголовком, который содержит адрес отправителя, адрес получателя, номер части, другую служебную информацию, и образовавшийся пакет запускается в сеть в самостоятельное “плавание”. Достигнув сервера назначения, пакеты, принадлежащие одному сообщению, собираются в единое целое (рис. 4.1).

Деление сообщения на пакеты позволяет использовать канал связи для одновременной передачи по нему нескольких сообщений: вместо длительной передачи одного большого файла по каналу передаются короткие пакеты, принадлежащие разным сообщениям. Получается, что сеть одновременно обслуживает всех пользователей, никто не простаивает в очереди.

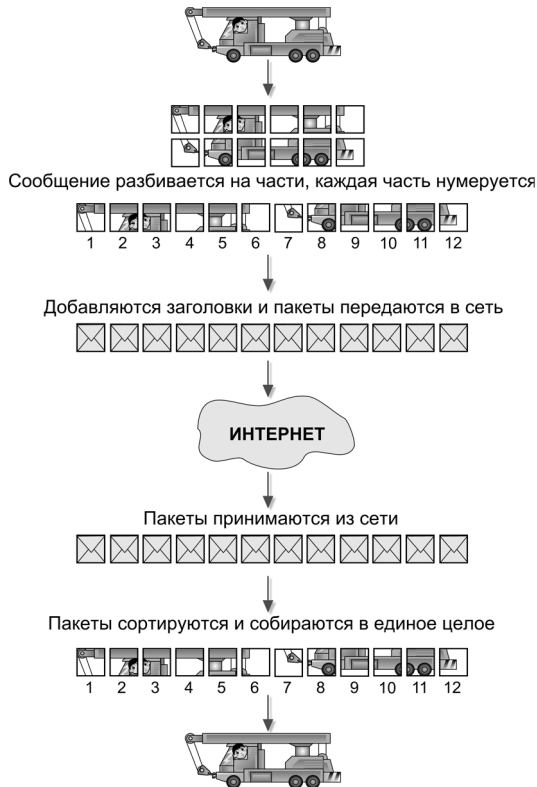


Рис. 4.1. Передача сообщения по сети Интернет

Вася. Да, но общее время передачи каждого сообщения не уменьшается, а увеличивается, так как пакеты содержат дополнительную служебную информацию, да и разбиение на пакеты, и сборка исходного сообщения требуют дополнительного времени.

Петя. Верно, но передача маленьких порций работает гораздо надежнее, чем передача больших.

Вася. Почему?

Петя. Крупные порции передаются долго, и, следовательно, возрастает вероятность отказа канала или возникновения в нем помех во время передачи сообщения.

Вася. Если на линии связи возникают помехи, которые искажают передачу, то файл все равно будет запорчен независимо от того, передается он целиком или частями!

Петя. Верно! Но при пакетной технологии не надо заново передавать весь файл, достаточно повторить передачу только “испорченных” частей-пакетов (рис. 4.2).

Вася. Понятно! Значит, для надежности размер пакета надо делать очень маленьким?

Петя. Слишком маленькие пакеты — тоже плохо! Каждый пакет в дополнение к исходным данным содержит служебную информацию, а это увеличивает общее время передачи. Плюс время разбиения на пакеты в стартовой точке и время сборки сообщения в конечном пункте.

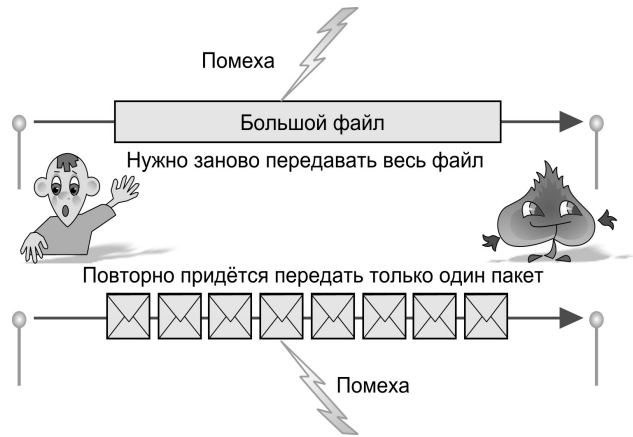


Рис. 4.2. Передача с помехами

Вася. Каковы же размеры пакетов, циркулирующих в сети Интернет?

Петя. Стандартный размер пакета в Интернете составляет 576 байт. С пакетами такой длины обязаны работать все сети TCP/IP (это прописано в стандартах TCP/IP). Реально могут использоваться и другие длины пакета, например, 1500 байт.

Вася. Еще один вопрос: как сервер отличит “хороший” пакет от “испорченного”?

Петя. В заголовок пакета перед отправкой в сеть записывается контрольная сумма, которая отвечает за целостность пакета. На принимающем сервере контрольная сумма вычисляется заново, и результат сравнивается с записью в заголовке. Если значения совпадают, пакет признается “хорошим”.

Вася. Что представляет собой контрольная сумма?

Петя. Как конкретно в IP-пакете вычисляется контрольная сумма, я не знаю (существуют разные алгоритмы), но могу рассказать общую идею.

Компьютерные данные (в том числе и сообщение, предназначенное для передачи в сеть) представляют собой цепочку двоичных цифр (нулей и единиц). Контрольная сумма — это сумма этих цифр.

Предположим для простоты, что пакет передается в виде трех десятичных цифр, первые две из которых — данные, а последняя — контрольная сумма (рис. 4.3).



Рис. 4.3. Передача сообщения с контрольной суммой

Вася. Разве одной цифры хватит для контрольной суммы в таком пакете? Чему будет равна контрольная сумма при передаче числа 99 (рис. 4.4)? Ведь сумма $9 + 9 = 18$ содержит две цифры!



Рис. 4.4. Сумму нельзя записать одной цифрой

Петя. В качестве контрольной суммы указывают не саму сумму цифр, а остаток от деления ее на число $N + 1$, где N — максимальное число, которое можно записать в поле, отведенном под контрольную сумму.

Вася. В нашем случае $N = 9$, $N + 1 = 10$, значит, контрольная сумма равна 8 (остаток от деления числа 18 на 10) (рис. 4.5).



Рис. 4.5. Контрольная сумма записывается одной цифрой

Петя. Да, верно! Вот тебе задание для тренировки: пусть в информационном блоке пакета передаются 27 шестерок. Назови контрольную сумму, если в пакете для нее отводится 2 цифры.

Вася. Сумма цифр информационного блока равна $27 * 6 = 162$.

Максимальное число, которое можно записать двумя цифрами: 99, значит, $N = 99$, $N + 1 = 100$, и тогда контрольная сумма равна 62 (остаток от деления числа 162 на 100).

Петя. Все правильно! В реальном интернет-пакете контрольная сумма вычисляется не для всего пакета, а только для его заголовка, и для нее отводится 16 бит, значит, $N + 1$ в этом случае есть двоичное число, в котором за первой единицей следуют 16 нулей: 1000000000000000.

В десятичной системе счисления это число равно 65536.

Вася. Думаю, что совпадение контрольных сумм в начальной и конечной точке не гарантирует правильность полученного пакета!

Петя. Да, конечно! Во-первых, контрольная сумма сама записывается с потерями: (62 вместо 162), во-вторых, искажения цифр могут компенсировать друг друга (двоичные данные “01” и “10” дают одну и ту же контрольную сумму, но данные — разные).

Совпадение контрольной суммы не гарантирует правильность полученных данных, а вот несовпадение однозначно говорит об ошибке. Несмотря на неполноту проверки, этот способ контроля достаточно эффективен и широко используется в компьютерной практике (например, архиваторами).

Вася. С контрольной суммой теперь все понятно.

Но мне кажется, что заголовок пакета, кроме адреса отправителя, адреса получателя, номера части и контрольной суммы, должен содержать еще две записи.

Петя. Интересно, какие?

Вася. Во-первых, нужно знать, на сколько частей разбито сообщение в стартовой точке, чтобы в конечном пункте понять, что все пакеты получены. Во-вторых, нужно уметь различать пакеты, принадлежащие разным сообщениям.

Петя. Молодец! Все правильно! Такие записи присутствуют в заголовке пакета.

Поле *Идентификация* содержит число, играющее роль номера сообщения, — у всех пакетов, относящихся к одному сообщению, это число одно и то же.

Порядковый номер пакета содержит поле с соответствующим именем: *Номер пакета*.

А поле *Флаги* содержит особую пометку для последнего пакета.

Таким образом, заголовок пакета содержит следующую служебную информацию:

- *Идентификация* (указывает принадлежность пакета одному сообщению);
- *Флаги* — статус пакета: обычный или последний;
- *Номер пакета* (для правильной сборки в конечном пункте);
- *Время жизни пакета* (предотвращает бесконечное циркулирование пакета в сети);
- *Контрольная сумма* (позволяет распознавать испорченные пакеты);
- *Адрес источника*;
- *Адрес приемника*;
- Другая служебная информация.

Замечание. Петя не совсем точно объяснил Васе содержание поля *Номер пакета*. В этом поле записывается не номер пакета, а *смещение* (в байтах) данных, содержащихся в пакете, от начала сообщения. В силу чего и само поле называют не номером пакета, а *Смещением пакета*. Но смысл этого поля Петя передал совершенно точно: оно играет роль порядкового номера. Более подробно о формате пакета можно прочитать в разделе *Академия*.

Петя. Продолжая сравнивать Интернет с телефонией, отмечу, что для передачи пакета в Интернете не требуется удерживать связь между отправителем и получателем, как это делается при телефонном разговоре. Это второе существенное свойство передачи данных в Интернете.

При обычном разговоре на телефонных линиях прокладывается маршрут между аппаратами абонентов, и эта связь сохраняется на все время разговора. Пока вы говорите, никто не может дозвониться ни вам, ни вашему абоненту: линии заняты (рис. 4.6).

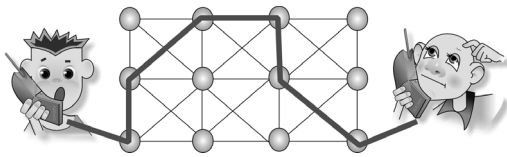


Рис. 4.6. Коммутация каналов

Такую связь называют связью с *коммутацией каналов* (или линий).

Передача пакетов происходит по-другому. Связь между исходной и конечной точкой не устанавливается, каждый сервер по пути следования передает пакет соседу. Получается, что ответственность за прохождение данных в сети Интернет возлагается на сами данные. Информация, содержащаяся в заголовке пакета, управляет работой каждого передающего сервера (рис. 4.7).

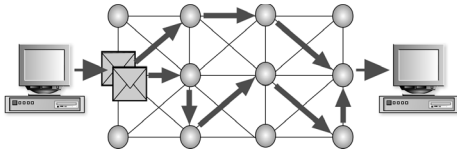


Рис. 4.7. Коммутация пакетов

Такую связь называют связью с *коммутацией пакетов*.

Вася. Это очень похоже на обычную бумажную почту. Адресная часть на конверте управляет проходом письма от отправителя к адресату.

А не может пакет заблудиться в сети?

Петя. Такое возможно. Кроме того, пакет может быть случайно уничтожен из-за неисправностей на сервере или в каналах связи.

Вася. Значит, конечный сервер отдаст пользователю сообщение с пропусками?

Петя. Если через определенное время какие-то пакеты сообщения получены не будут, сервер назначения pošлет на исходный сервер просьбу переслать недостающие части заново.

Вася. Получается, что передача информации в Интернете имеет две главных особенности:

- **Пакетная передача.** На исходном сервере сообщение разделяется на части-пакеты. Каждый пакет снабжается служебным заголовком, который содержит адрес отправителя, адрес получателя, номер пакета, контрольную сумму и другую служебную информацию. На конечном сервере из полученных пакетов сообщение восстанавливается.

- **Коммутация пакетов.** Пакеты передаются в сеть ближайшему серверу по пути следования. Информация, содержащаяся в заголовке пакета, сама управляет прохождением пакета к пункту назначения.

Петя. Ты все понял правильно. Самыми важными частями заголовка пакета являются адреса отправителя и получателя. О том, как устроен интернет-адрес, интересно поговорить отдельно.

Адресация в Интернете

Узлы в Интернете имеют двойную адресацию — символическую (*доменная адресация*) и числовую (*IP-адресация*).

Доменная адресация

Петя. Электронные адреса в Интернете строятся по иерархическому принципу. Иерархия — это зависимость по подчинению, или включению одних объектов в другие.

Обычные почтовые адреса тоже имеют иерархическую структуру (рис. 4.8).



Рис. 4.8. Иерархия обычного почтового адреса

Иерархия удобна тем, что исключает повторение адреса для разных почтовых пользователей.

Дом с номером 1, как правило, есть на любой улице. Но это не страшно — название улицы входит в состав почтового адреса. Становится неважным и то, что улица Строителей есть и в Москве, и в Санкт-Петербурге, ведь в адресе указывается город. Город Павловск есть и в Воронежской области, и в Ленинградской, но это не мешает доставке писем, ведь в адресе указывается область.

Интернет-адреса (электронные адреса) строят подобным образом.

Иерархическую систему электронных адресов называют *доменной* (от *domain* — область, район).

Для каждой страны выделяется свое обозначение (первая часть адреса, домен страны) и предоставляется полная свобода для выбора обозначений других частей.

Страна регистрирует вторую часть адреса для крупных организаций и объединений пользователей, предоставляя им право решать вопрос о выборе обозначений для остальных адресных частей.

Таким образом, каждый сервер Интернета получает свой уникальный сетевой адрес. Провайдер сервера назначает адреса своим клиентам, добавляя к своему собственному адресу дополнительную часть — домен пользователя.

Вася. А какой адрес в сети у нашего компьютера?

Петя. Наш адрес *kuk.pereslavl.ru*. Адрес строится справа налево, и его части отделяются друг от друга точками:

ru — это адрес России;
pereslavl.ru — это адрес нашего провайдера (он находится в Переславле);
kuk.pereslavl.ru — это адрес нашего компьютера.

Адреса в Интернете могут иметь следующий вид:

компьютер.подразделение.организация.страна

Вполне возможен, например, такой адрес: *kisa.roga.obender.ru*.

Или такой: *john.managers.microsoft.com*.

Вася. А какая страна имеет домен *com*?

Петя. Такой страны нет... Исторически сложилось так, что изначально в США были только тематические домены:

Домен	Тематическая направленность
com	коммерческие организации
net	компьютерные сети
edu	образование
gov	правительственные учреждения
mil	военные организации
org	общественные и некоммерческие организации

Когда Интернет стал распространяться по всему миру, появились домены для обозначения разных стран, в том числе и территориальный домен США, а тематические домены закрепились для международного использования.

Ниже приводятся названия доменов некоторых стран:

Домен	Страна	Домен	Страна
by	Белоруссия	jp	Япония
ca	Канада	pt	Португалия
ch	Швейцария	ru	Россия
de	Германия	se	Швеция
es	Испания	su	Бывший СССР
fi	Финляндия	tw	Тайвань
fr	Франция	tr	Турция
il	Израиль	ua	Украина
in	Индия	uk	Великобритания
it	Италия	us	США

IP-адресация

Яков Наумович Зайдельман — добрый товарищ Куков — известен своим ворчливым нравом и въедливым характером. Впрочем, его ворчливость и въедливость проявляются исключительно в профессиональном отношении, когда ему приходится принимать участие в научных дискуссиях. В обычном общении он приятный собеседник.

Понятно, что никому не нравится критика! Но Вася и Петя за долгие годы привыкли к ЯНЗ (так называют Якова Наумовича товарищи) и очень рады, когда он принимает участие в их обсуждениях.

Вот и сегодня ЯНЗ появился у Куков как раз в разгар разговора об адресах в Интернете. И сразу начал с критики.

ЯНЗ. То, что вы рассказали об адресах, не имеет отношения к реальным основам маршрутизации сообщений.

— Привет и тебе, Яша! — только и могли в ответ сказать ошеломленные Куки.

ЯНЗ. Реально при пересылке информации в Интернете используются не символьные, а числовые адреса. Каждый компьютер в сети имеет свой собственный номер, который называется IP-адресом.

Замечание. На самом деле Интернет начинался именно с доменной системы адресации, таблицы маршрутизации состояли из строк, в которых для каждого доменного адреса указывалось, какому соседнему узлу нужно передать пакет для его продвижения к пункту назначения. Но такие таблицы строились, когда сеть работала по протоколу УИУСР (смотрите ответ на вопрос “Как работает сеть по протоколу УИУСР?” в разделе FAQ). Сейчас, когда Интернет работает по протоколу TCP/IP, в таблицы маршрутизации заносятся числовые адреса, о которых собирается рассказать ЯНЗ. В этом смысле Яков Наумович, конечно, прав: в современном Интернете маршрутизаторы работают только с числовыми IP-адресами!

ЯНЗ. IP-адрес — это целое число от 0 до 4 294 967 295. Отсюда видно, что теоретически в Интернет может входить не более чем примерно 4 миллиарда компьютеров. Это число кажется очень большим, но уже сейчас ощущается нехватка свободных IP-адресов для подключения к сети новых компьютеров.

Дело в том, что некоторые IP-адреса имеют специальный смысл и не могут использоваться для обычных сетевых подключений. Кроме того, имеется много “мертвых” адресов, которые были когда-то выделены, но в настоящее время не используются. Найти и вернуть такие адреса довольно сложно. Поэтому сейчас разрабатываются новые правила IP-адресации, которые позволят решить эти проблемы.

Вероятно, Васе покажется странным столь некруглое число возможных IP-адресов. На самом деле число это очень даже круглое; только, чтобы увидеть это, надо записать его в двоичной системе. Для записи IP-адреса используется 32 бита, и указанное число — это просто максимально возможное значение, для записи которого хватает 32 бит.

Но человеку неудобно запоминать длинные последовательности бит или соответствующие им десятичные числа. Поэтому для записи IP-адресов обычно применяют специальный формат: 32 бита, образующие IP-адрес, разбиваются на 4 байта, каждый байт записывается в виде десятичного числа от 0 до 255. Получается 4 небольших десятичных числа, которые отделяются точками (*рис. 4.9*).

32-битный IP-адрес:

11000001	11101000	10101110	00000001
1-ый байт	2-ой байт	3-ий байт	4-ый байт

Десятичная запись того же IP-адреса:

193.232.174.1

$11000001_2 = 193_{10}$
$11101000_2 = 232_{10}$
$10101110_2 = 174_{10}$
$00000001_2 = 1_{10}$

Рис. 4.9. Запись двоичного IP-адреса в десятичном виде

Например, IP-адрес сервера Переславы, услугами которого пользуется Роботландия, — 193.232.174.1. Получается что-то вроде номера телефона, который несложно запомнить.

Но даже такая форма записи человеку не очень удобна. Хотя IP-адрес содержит много интересной информации (например, можно довольно точно определить, где находится соответствующий компьютер, это похоже на то, как по номеру телефона часто можно узнать район города), но эта информация понятна только специалистам.

Поэтому в дополнение к числовой адресации используется описанная вами доменная система. Доменные имена удобнее для запоминания, они содержат информацию в более понятном для человека виде. Но реальная пересылка все равно ведется по IP-адресам.

Специальные серверы — они называются DNS-серверы (*Domain Name System*, доменная система имен) — хранят список доменных имен и соответствующих им IP-адресов. Обычный сервер обращается к DNS-серверу, указывая ему доменное имя. В ответ DNS-сервер сообщает IP-адрес, и обычный сервер использует его при отправке в сеть информационного пакета.

Чтобы лучше разобраться в этой двойной адресации, можно продолжить аналогию с обычной почтой.

В почтовом адресе есть индекс, который однозначно указывает почтовое отделение. Если индекс правильный, то ни область, ни город можно не указывать. Но человеку неудобно запоминать числа, он может ошибиться, поэтому, кроме индекса, мы пишем и обычный адрес.

Автоматы по сортировке писем читают на конверте индекс, а живые почтальоны — обычный адрес. В компьютерных сетях роль индекса играют IP-адреса, а аналог обычного — доменный адрес.

ЯНЗ на секунду замолчал, чтобы перевести дух, и Петя сумел вставить в разговор свое слово.

Петя. Отмечу независимость двух систем адресации. Не существует алгоритма, который мог бы из доменного адреса вывести соответствующий ему IP-адрес и наоборот. DNS-серверы для перевода доменного адреса в IP-адрес используют обычную таблицу: в одном ее столбце расположены доменные, а в другом числовые адреса. Из этого следует, например, что одному и тому же IP-адресу вполне могут соответствовать несколько доменных адресов — сообщения не будут теряться в сети.

ЯНЗ. Верно. Независимость числовой и символьной адресации особенно удобна, когда пользователь меняет провайдера. Он меняет IP-адрес, а доменное имя может сохранить прежним.

Более подробно о формате IP-адреса можно прочитать в разделе *Академия*.

Протоколы Интернета

Вася внимательно выслушал монолог ЯНЗ, реплику брата и, как всегда, задал один из своих наивных вопросов.

Вася. Что означает приставка “IP” в обозначении адреса Интернета? Почему вы говорите IP-адрес?

ЯНЗ уже открыл было рот, чтобы продолжить свою лекцию, но лихой Петя сумел опередить его.

Петя. Дело в том, что IP — это название основного протокола Интернета (*Internet Protocol*), по которому выполняется передача информационных пакетов в сети. Сами информационные пакеты тоже называются IP-пакетами.

Вторым базовым протоколом Интернета является протокол TCP (*Transmission Control Protocol*, протокол управления передачей).

Протокол TCP разбивает информацию на порции и нумерует их, чтобы при получении можно было правильно собрать исходное сообщение. Далее протокол IP снабжает порции адресами отправителя и получателя, контрольной суммой, другой служебной информацией и отправляет образовавшиеся IP-пакеты в сеть.

В конечной точке протокол TCP проверяет, все ли части сообщения получены. Так как разные пакеты приходят в конечную точку разными путями, порядок их получения может быть нарушен. После получения всех частей TCP сортирует их в правильном порядке и собирает в единое целое.

На *рис. 4.10* условно показана работа протоколов TCP и IP при передаче сообщения.

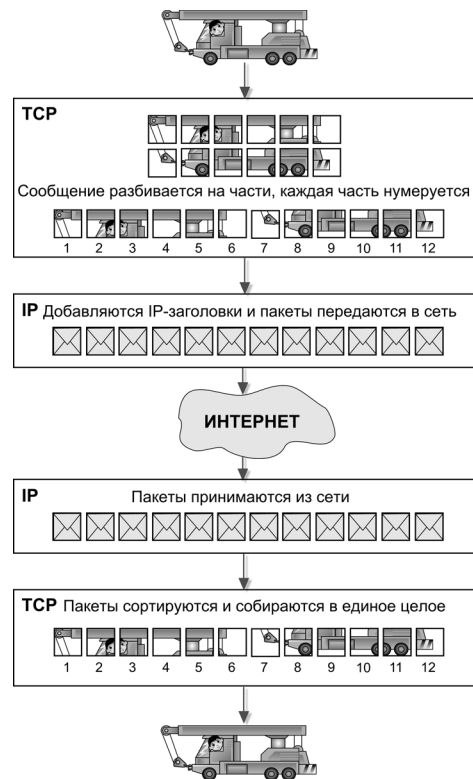


Рис. 4.10. Работа протоколов TCP, IP

ЯНЗ. Так как протоколы TCP и IP работают совместно, то обычно говорят о них как об одном базовом протоколе Интернета, называя его TCP/IP.

Вася. Меня смущает, как Петя описывает работу сети: “протокол разбивает, нумерует...”. Как будто

протокол — это человек или программа, а ведь протокол — это свод правил, если я правильно понял?

Петя. Ты прав! Мои выражения — фигуры речи! Я использовал их, чтобы не говорить длинно: “сетевое программное обеспечение согласно протоколу ТСР разбивает информацию на порции и нумерует их...”!

Вася. Ладно! Используй и дальше свои фигуры, теперь буду понимать, что за ними скрывается!

Маршрутизация пакетов в Интернете

— Все, о чем мы говорили, мне понятно. — Вася по привычке почесал свой вихрастый затылок. — Но понятно по отдельности. Подобно протоколу ТСР, мне трудно сложить цельную картину из отдельных частей при явной недостатке их количества. У меня ощущение, что Интернет — чертовски сложная штука!

— Не буду с этим спорить, — согласился Петя. — Интернет гораздо легче использовать, чем разобраться во всех его технических деталях. Но такое положение вещей является обычным. Включая телевизор, ты не часто задумываешься, каким образом говорящая голова попадает в твой ящик! Но если ты сможешь объяснить это хотя бы в общих чертах, тебя будут считать культурным человеком.

ЯНЗ. А может быть, в будущем ты станешь сетевым специалистом, и тогда все нюансы этой технологии станут для тебя ясными до самых глубин.

— Еще один вопрос для повышения моей культуры. — Вася поднял руку, как на школьном уроке. — Как информация прокладывает себе путь от начального сервера к конечному?

Ведь создается впечатление, что пакет бросают на произвол судьбы, и ему самому приходится прилагать усилия для выбора правильного направления. Кажется просто невероятным, что информация в итоге добирается до станции назначения.

— Ты не одинок в своих сомнениях! — Петя снова опередил ЯНЗ, захватив лекционную трибуну. — Где-то я читал, что представители телефонной компании “АТ&Т” тоже не признавали идею сети, основанной на коммутации пакетов. Они заявляли, что это никогда не будет работать! Сеть, в которой информация сама находит маршрут, — это утопия. Так говорили они и предлагали проверенный старый способ — коммутацию линий.

Однако сети с коммутацией пакетов успешно работают, информация находит пункт назначения, как бы ни казалось это невероятным.

Прохождение пакета по сети можно представить следующим образом. Вообрази, что в совершенно незнакомом городе нужно пройти пешком из пункта А в пункт Б. Карты города нет, но на каждом перекрестке стоит постовой, который показывает нужное направление.

Вася. Похоже, с такими постовыми появляется шанс дойти до конечного пункта! Понятно, на что ты намекаешь: город — это Интернет, перекрес-

ток — это сервер Интернета, путник — это IP-пакет. А кто является постовым?

Петя. Роль постовых, указывающих направление, в Интернете выполняют *маршрутизаторы* — специальные устройства с компьютерной “начинкой” или специальное программное обеспечение, установленное на сервере Интернета.

Маршрутизаторы связаны между собой по иерархическому принципу построения адресов в сети. Они хранят таблицы соединений серверов, постоянно их обновляя (обычно каждые 30 секунд).

Вася. А как маршрутизаторы получают информацию для построения своих таблиц?

Петя. Маршрутизаторы посылают запросы в сеть. Соседние серверы “откликаются” и сообщают о своих непосредственных соединениях с другими серверами. На основе этой информации обновляются адресные таблицы.

Вася. Признаюсь, что все сказанное тобой не помещается в моей голове и норовит вылезти из нее как булавки из тряпичной головы Страшилы!

Петя. Попробую рассказать подробнее. Когда сервер получает пакет, он смотрит на адрес получателя в его заголовке и “думает”, куда бы переслать полученную информацию. Адреса своих клиентов сервер знает, и если сообщение — для них, проблем нет.

Вася. Этот случай простой, и здесь все ясно.

Петя. Если сообщение не “местное”, сервер смотрит свои таблицы соединений и ищет в них получателя сообщения.

Вася. Если в таблицах получатель есть, маршрутизатор сообщает первый пункт следования, извлекая его из соответствующей табличной строки. Это понятно.

Петя. Если получатель в таблицах не найден, маршрутизатор передает запрос другому маршрутизатору, своему предку по иерархической адресной лестнице, и тот ищет получателя среди своих потомков или, в свою очередь, запрашивает маршрутизатор своего родителя.

Вася. И, как это ни удивительно, в конце концов маршрут в сети будет проложен?

Петя. С большой степенью вероятности — да! Если, конечно, адрес получателя вообще существует.

Пусть, например, в сети, показанной на *рис. 4.11*, пакет нужно передать из пункта А в пункт Б.

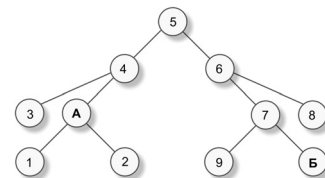


Рис. 4.11. Сеть для примера

Сервер А не видит получателя среди своих потомков и передает запрос родителю — серверу 4. Сервер 4 не находит адресата на сервере 3 и передает запрос на сервер 5. Сервер 5 находит получателя среди своих потомков — маршрут проложен.

Продолжение следует

НАЧАЛКА

газета-клуб для всех,
кто учит информатике
маленьких детей



№ 14 (1-15 ноября)

Материалы конкурса "ТРИЗформашка-2006". Часть I

Н.Г. ИВАНОВА, М.А. ПЛАКСИН, О.Л. РУСАКОВА,

г. Пермь

В № 5, 7/2006 были напечатаны материалы "пермских" конкурсов по информатике "ТРИЗформашка" за 2004 и 2005 гг. Сегодня мы начинаем публикацию материалов конкурса "ТРИЗформашка-2006". Материалы эти настолько объемны, что потребуют трех газетных выпусков. Публикация заданий начнется в этом номере и закончится в следующем. Ответы и правила оценивания будут опубликованы в двух следующих номерах.

Краткая характеристика конкурса "ТРИЗформашка-2006" дана в следующей таблице:

Класс	Всего заявок	Всего участников	Место			Балл			Всего выполнено заданий	Кол-во заданий на команду		
			Высшее	Среднее	Низшее	Высший	Средний	Низший		Мак	Среднее	Мин
1	3	3	17	111.3	202	549.75	241.1	3.0	19	10	6.33	3
2	10	10	56	136.7	197	304.50	132.7	14.5	66	12	6.60	2
3	26	25	4	119.1	201	746.31	193.8	7.4	190	14	7.60	1
4	44	36	2	114.0	193	938.57	211.0	23.0	304	19	8.44	3
5	52	41	6	107.4	195	642.35	213.9	18.0	316	14	7.71	1
6	34	32	1	90.8	199	1214.65	300.9	9.0	264	20	8.25	2
7	60	55	3	80.2	200	834.75	298.8	9.0	455	16	8.27	1
Всего	229	202					244.2		1614		7.99	

Обратим внимание на динамику количества участников конкурса: 2004 г. — 45 команд, 2005 г. — 143 команды, 2006 г. — 202 команды.

Задания конкурса "ТРИЗформашка-2006"

1. Оксюмороны. Истина и ложь (противоречие) — это две несовместимые характеристики одного высказывания. Противоречие недопустимо в строгом рассуждении, когда оно смешивает истину с ложью. Но в художественной речи противоречия играют важную роль, добавляя ей образности, выразительности.

Оксюморон — сочетание логически враждующих понятий, вместе создающих новое представление.

Например:

— *Пышное природы увяданье* (А.С. Пушкин). Действительно, как богато, пышно, красиво выглядит листва деревьев осенью, перед увяданьем!

— *Свеча темно горит* (А.С. Пушкин). Значит, свеча дает недостаточно света.

— *"Книга обо всем и еще о многом другом"*. Так называется книга испанского писателя XVI—XVII вв.

Ф.Кведо. Это название подчеркивает отсутствие какой-либо определенной темы у книги.

Придумайте свои оксюмороны и объясните их смысл.

Ответ — текст.

2. ММЧ. Паяльная лампа — это переносной нагревательный прибор с открытым пламенем (горелка). Он состоит из металлического бака (с ручкой), в который заливается горючее (бензин, керосин, спирт). Сверху над баком располагается разбрызгиватель (форсунка). Струи горючего выбрызгиваются из форсунки, зажигаются и образуют факел.

На рисунках показаны современная паяльная лампа (не работающая), работающая паяльная лампа, которую используют знаменитые диснеевские Спасатели Чип, Дейл, Гайка и Рокфор, применение паяльной лампы для нагрева металлической детали.

Смоделируйте работающую паяльную лампу с помощью маленьких человечков.

Ответ — рисунок.



3. Рыбки. Пусть изображение рыбки состоит из тела, хвоста, одного верхнего и одного нижнего плавника. Даны три разных вида хвостов, три вида верхних плавников, три вида нижних плавников. Составьте как можно больше разных рыбок, используя предложенные детали.

Ответ — рисунок.



4. Морфоящик. Придумайте изобретательскую машину (морфологический ящик) для изобретения столов.

Ответ — текст.

5. Три поросенка — Ниф-Ниф, Нуф-Нуф и Наф-Наф — изготовили шифровальную машину из лампочек для елочных новогодних гирлянд. Поросята придумали специальный шифр для кодирования чисел от 0 до 15: числа можно зашифровать набором горящих и негорящих лампочек, причем каждое число кодируется четырьмя лампочками.

Код поросят (горящие лампочки — светлые):

● ● ● ●	0	○ ● ● ●	8
● ● ● ○	1	○ ● ● ○	9
● ● ○ ●	2	○ ● ○ ●	10
● ● ○ ○	3	○ ● ○ ○	11
● ○ ● ●	4	○ ○ ● ●	12
● ○ ● ○	5	○ ○ ● ○	13
● ○ ○ ●	6	○ ○ ○ ●	14
● ○ ○ ○	7	○ ○ ○ ○	15

Наф-Наф придумал интересную игру. Он формировал различные гирлянды и предлагал решить примеры. Решите примеры Наф-Нафа.

Ответ — рисунок.

1) Если первое число разделить на второе, результат умножить на третье и затем из полученного числа вычесть четвертое, то получится то, что нужно добавить к гирлянде.



Дополните гирлянду.

2) Первое число разделили на второе. К результату прибавили третье, и получилось четвертое. Но вот беда: в коде третьего числа перегорела лампочка. Замените лампочку и зажгите ее.



6. Коврик бабушки Глафиры. Бабушка Глафира Петровна из деревни Серебряные Росы славится своими ковриками. Она их шьет из квадратных лоскутков одинакового размера с пятью различными узорами, соединяя их так, чтобы все соседние лоскутки были различны и чтобы ни один лоскуток не имел двух соседей с одним и тем же узором. Соседними считаются лоскутки, имеющие общие стороны.

Создайте рисунок бабушкиного коврика размером 10×10 .

Ответ — рисунок.

7. Красавица Глаша и сэр Джон. (Краткое содержание предыдущих задач. В задачах “ТРИЗ-формашки”-2004, 2005 было рассказано о том, как любитель экстремального отдыха сэр Джон отправился на каникулы из Оксфорда в гости к своей тетке Марфе в подмосковную деревню Ромашкино. В соседней деревне Васильково он встретил красавицу Глашу. Сэр Джон и красавица Глаша стали хорошими друзьями. Во время совместных путешествий им приходится составлять алгоритмы, решать логические задачи и т.д.)

Сэр Джон давно мечтал увидеть русскую зиму. Перед Рождеством ему выдалась возможность вырваться из Оксфорда в дорогу его сердцу деревню Ромашкино. Тетка Марфа была довольна: “Вот сейчас позволю красавице Глаше, приглашу ее елку украшать к Новому году”. “Замечательно! Где тут у вас елки продают?” — поинтересовался Джон. “Не надо покупать, зачем рубить красоту? Мы елочку в лесу нарядим!”

На следующее утро Марфа, Глаша, сэр Джон, а также собака Чернильница и кошка Роксолана вышли в лес елку наряжать. Тропа была узкая, поэтому шли друзья гуськом:

1) Тетка Марфа шла перед красавицей Глашей, но после кошки Роксоланы.

2) Сэр Джон и Роксолана не шли друг за другом.

3) Собака Чернильница не находилась рядом ни с кошкой, ни с Марфой, ни с сэром Джоном.

Друзья дошли по узкой тропинке до красивой поляны и увидели невысокую стройную елочку, стоящую в центре поляны. С большим воодушевлением они нарядили елочку, поводили вокруг нее хороводы, спели новогодние и рождественские песенки на двух языках. В хорошем настроении дружная компания выстроилась в обратный путь на узенькой тропинке. “Интересно, мы стоим в обратной последовательности по отношению к тому, как шли сюда”, — в задумчивости сказал сэр Джон.

В каком порядке тетка Марфа, красавица Глаша, сэр Джон, а также собака Чернильница и кошка Роксолана вышли в обратный путь?

Ответ — текст.

(Продолжение истории сэра Джона и красавицы Глаши см. в одном из следующих конкурсов.)

8. Во время очередного похода в Хогсмид Гарри Поттер, Рон, Гермиона, Джинни, Фред и Джордж зашли в трактир “Три метлы”, чтобы немного перекусить.

1) Гарри взял то же, что и Рон, и вдобавок еще печенье.

2) Гермиона купила то же, что Джинни, но не стала покупать сахарные перья.

3) Джордж ел то же, что Фред, но без мармеладных лягушек.

4) Рон взял то же, что и Фред, но эклерам он предпочел засахаренные орешки.

5) Джинни ела то же, что и Фред, но вместо сливочного пива пила тыквенный сок.

Что купил каждый из детей? Расположите продукты, купленные друзьями, в порядке убывания их популярности.

Ответ — текст.

9. Назовите как можно больше систем, обладающих следующей функцией:

(1) отрезалка, (2) соединялка, (3) шумелка.

Назовите как можно больше систем, обладающих сразу двумя из названных функций, сразу тремя. Укажите, какими функциями обладают перечисленные системы.

Ответ — текст.

10. Назовите как можно больше функций системы “автомобиль”.

Ответ — текст.

11. “Черные ящики”. Дано описание трех систем в виде “черных ящиков”. Определите функции этих систем. Для каждого “ящика” запишите его функцию и продолжите заполнение таблицы “Вход-выход”.

Ответ — текст.

1)

Вход	а	б	аб	абв	абвг	ба	бав	баг	в	вв	даааа	даа	н	эюя	малции				
Выход	б1	в1	б2	б3	б4	в2	в3	в3	г1	г2	е5	е3					д4	и2	л3

2)

Вход	а	1	2	10	15	100	50	250	55	66	563	505	647
Выход	Не понимаю	2	4	20	210	200	100	4100	1010	1212	10126	10010	12814

Вход	763	29	091										
Выход				2010	4814	1024							

3)

Вход	1	а	б	в	аб	ав	ба	вб	абв	абг	бва	абвгд	ааааа	ддддд
Выход	Не понимаю	а	а	а	б	б	б	б	в	в	в	д	е	д

Вход	дгвба	бэби	вега	клмно	бац	алкомн							
Выход	д	г	г				е	е	ж				

12. Структура системы. Молекула сложного химического вещества представляет собой систему, состоящую из атомов простых веществ, связанных друг с другом.

Одна молекула нитробензола состоит из 6 атомов углерода, одного атома азота, двух атомов кислорода и 5 атомов водорода.

Каждый атом углерода имеет 4 связи с другими атомами, атом азота — 5 связей, каждый атом кислорода — 2 связи, каждый атом водорода — одну.

Шесть атомов углерода соединены в кольцо. Причем каждый из них имеет две связи с одним из своих соседей и одну связь с другим.

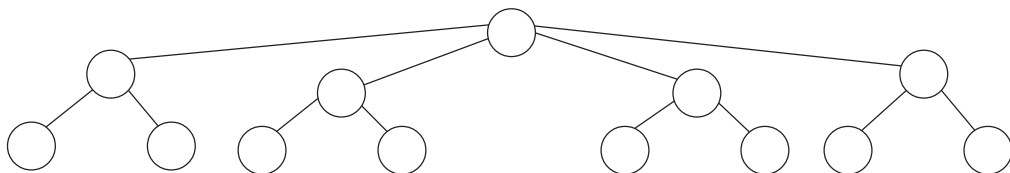
5 атомов углерода соединены одной связью с атомами водорода. Шестой атом углерода соединен одной связью с атомом азота. Кроме того, атом азота соединен с двумя атомами кислорода, с каждым — двумя связями.

В химии атом углерода обозначается буквой С, атом азота — буквой N, атом кислорода — буквой O, атом водорода — буквой H.

Изобразите структуру системы “молекула нитробензола”.

Ответ — рисунок.

13. Назовите, пожалуйста, как можно больше систем, имеющих следующую структуру:



Дайте пояснения.

Ответ — текст.

14. Таблица решений. В таблице решений приведены данные о том, какие лекарственные растения можно использовать при тех или иных заболеваниях.

Пользуясь таблицей решений, ответьте на поставленные вопросы.

Для простоты ответ можно дать в виде перечисления номеров решений (P31, P29, P43) или номеров условий (У31, У29, У43).

Ответ — текст.

У1	При ушибах	+	-	-	-	-	-	-	-	-
У2	При ссадинах	-	+	-	-	-	-	-	-	-
У3	При высокой температуре	-	-	+	-	-	-	-	-	-
У4	Общеукрепляющие	-	-	-	+	-	-	-	-	-
У5	При боли в горле	-	-	-	-	+	-	-	-	-
У6	При кашле	-	-	-	-	-	+	-	-	-
У7	При сыпи на коже	-	-	-	-	-	-	+	-	-
У8	Успокаивающие	-	-	-	-	-	-	-	+	-
У9	При воспалении во рту	-	-	-	-	+	-	-	-	-
У10	При порезах	-	+	-	-	-	-	-	-	-
У11	При простуде	-	-	+	+	+	-	-	-	-
У12	При ожогах	-	+	-	-	-	-	-	-	-
У13	При фурункулах	-	+	-	-	-	-	-	-	-
У14	При нарушении сна (бессоннице)	-	-	-	-	-	-	-	+	-
У15	При зубной боли	-	-	-	-	-	-	-	-	+
Р1	Подорожник (листья)	+	+			+				
Р2	Лопух (листья)	+								
Р3	Капуста (листья)	+								
Р4	Мята перечная (листья)					+	+		+	
Р5	Ромашка аптечная					+	+	+		+
Р6	Зверобой					+			+	
Р7	Багульник (побеги)						+			
Р8	Ноготки (цветы)					+	+			
Р9	Фиалка трехцветная						+			
Р10	Солодка (корень)						+			
Р11	Валериана (корень)								+	
Р12	Пустырник (корень)								+	
Р13	Малина (ягоды)			+						
Р14	Шиповник				+					
Р15	Рябина (ягоды)				+					
Р16	Клюква			+	+					
Р17	Иван-чай						+			
Р18	Чистотел		+							
Р19	Шалфей				+				+	+
Р20	Облепиха (масло)		+			+		+		

Вопросы

Что можно рекомендовать для повышения иммунитета (общего укрепления организма)?

1. Какие лекарственные растения можно предложить при простуде без температуры?
2. Какие растения являются одновременно успокаивающими и противокашлевыми?
3. Сравнить по возможным способам лечения боль в горле и кашель.
4. Что можно лечить подорожником?
5. Какое из перечисленных растений имеет самое широкое применение?
6. В каких случаях можно использовать облепиху вместе с ромашкой аптечной?
7. Сравнить по возможным способам применения подорожник и облепиху.
8. Составьте набор лекарственных растений, перечисленных в таблице, который позволяет лечить все перечисленные заболевания. Сделайте этот набор минимальным.
9. Перечислите все средства, которые можно использовать при лечении простуды.

15. Словарь.

1) Даны 60 слов. Сформируйте из них словарь с верхними индексами. В словаре должно быть 3 разворота. На каждом развороте по 4 колонки, в каждой колонке — по 5 слов.

2) Дано слово “напротив”. Запишите в столбик те слова, которые надо просмотреть при быстром поиске этого слова в нашем словаре. Рядом с ними выпишите буквы, которые при этом следует читать.

3) Дано слово “наугад”. Запишите в столбик те слова, которые надо просмотреть при быстром поиске этого слова в нашем словаре. Рядом с ними выпишите буквы, которые при этом следует читать.

Ответ — текст.

Слова для составления словаря:

начерно	наперерез	наоборот	нарочно	наскоро
налево	наповал	наклеить	налегке	например
наискосок	надвое	наперебой	наизнанку	наедине
насекомое	наподобие	накануне	наравне	назад
напильник	накись	нанимать	напротив	наперечет
налог	намерение	награда	невзгода	народ
наперекор	невидимка	население	напиток	напомнить
небосклон	нараспев	надолго	наизусть	наотмашь
надеяться	наряд	невод	начеку	наспех
начисто	нарасхват	настурция	напрямик	натощак
накидка	натура	наука	наяву	нафталин
напрокат	начало	направо	невежда	налим

16. Заполните табл. 1, используя информацию из табл. 2, 3, 4.

Запишите в табл. 1 информацию о трех федеральных округах, в которых находятся самые крупные города России.

Таблица 1

**Информация о трех федеральных округах,
в которых находятся самые большие по числу жителей города России**

Самый большой город	Количество жителей в городе (тыс. чел.)	Федеральный округ	Самый большой по площади регион округа	Площадь региона (тыс. кв. км)	Второй по числу жителей город региона	Количество жителей в городе (тыс. чел.)

Таблица 2

Федеральные округа Российской Федерации (на 1 января 2004 г.)

Федеральный округ	Площадь (тыс. кв. км)	Население (тыс. чел.)	Число жителей на 1 кв. км	Кол-во районов	Кол-во городов	Столица
Центральный	651	37 733	58	423	304	Москва
Северо-Западный	1678	13 832	8	154	146	Санкт-Петербург
Южный	589	22 850	39	257	135	Ростов-на-Дону
Приволжский	1038	30 902	30	448	196	Нижний Новгород
Уральский	1789	12 316	7	116	114	Екатеринбург
Сибирский	5115	19 901	4	325	132	Новосибирск
Дальневосточный	6216	6634	1	143	70	Хабаровск
Всего	17 075	144 168	8	1866	1097	

Таблица 3

Регионы Российской Федерации (на 1 января 2004 г.)	Площадь (тыс. кв. км)	Население (тыс. чел.)	Кол-во городов	Столица	Федеральный округ
Агинский Бурятский автономный округ	19	73	0	пос. Агинское	Сибирский
Алтайский край	169	2583	12	Барнаул	Сибирский
Амурская область	364	894	9	Благовещенск	Дальневосточный
Архангельская область	587	1318	14	Архангельск	Северо-Западный
Астраханская область	44	1001	6	Астрахань	Южный
Белгородская область	27	1513	10	Белгород	Центральный
Брянская область	35	1361	16	Брянск	Центральный
Владимирская область	29	1504	23	Владимир	Центральный
Волгоградская область	114	2673	19	Волгоград	Южный
Вологодская область	146	1255	15	Вологда	Северо-Западный
Воронежская область	52	2353	15	Воронеж	Центральный
Еврейская автономная область	36	190	2	Биробиджан	Дальневосточный
Ивановская область	22	1130	17	Иваново	Центральный
Иркутская область	768	2561	22	Иркутск	Сибирский
Кабардино-Балкарская Республика	13	899	8	Нальчик	Южный
Калининградская область	15	950	22	Калининград	Северо-Западный
Калужская область	30	1029	19	Калуга	Центральный
Камчатская область	472	355	4	Петропавловск-Камчатский	Дальневосточный
Карачаево-Черкесская Республика	14	437	4	Черкесск	Южный
Кемеровская область	96	2872	20	Кемерово	Сибирский
Кировская область	121	1479	18	Киров	Приволжский
Коми-Пермяцкий автономный округ	33	134	1	Кудымкар	Приволжский
Корякский автономный округ	302	24	0	пос. Палана	Дальневосточный
Костромская область	60	726	12	Кострома	Центральный
Краснодарский край	76	5106	26	Краснодар	Южный
Красноярский край	2340	2942	25	Красноярск	Сибирский
Курганская область	71	1004	9	Курган	Уральский
Курская область	30	1214	10	Курск	Центральный
Ленинградская область	85	1660	31	Санкт-Петербург	Северо-Западный
Липецкая область	24	1201	8	Липецк	Центральный
Магаданская область	461	178	2	Магадан	Дальневосточный
г. Москва	...	10391	1		Центральный
Московская область	47	6622	76	Москва	Центральный
Мурманская область	145	880	16	Мурманск	Северо-Западный
Ненецкий автономный округ	177	42	1	Нарьян-Мар	Северо-Западный
Нижегородская область	77	3479	28	Нижний Новгород	Приволжский
Новгородская область	55	683	10	Великий Новгород	Северо-Западный
Новосибирская область	178	2673	14	Новосибирск	Сибирский
Омская область	140	2059	6	Омск	Сибирский
Оренбургская область	124	2163	12	Оренбург	Приволжский
Орловская область	25	850	7	Орел	Центральный
Пензенская область	43	1436	11	Пенза	Приволжский
Пермская область	161	2791	25	Пермь	Приволжский
Приморский край	166	2051	12	Владивосток	Дальневосточный
Псковская область	55	748	14	Псков	Северо-Западный
Республика Адыгея	8	445	2	Майкоп	Южный
Республика Алтай	93	203	1	Горно-Алтайск	Сибирский
Республика Башкортостан	144	4092	21	Уфа	Приволжский
Республика Бурятия	351	974	6	Улан-Удэ	Сибирский
Республика Дагестан	50	2602	10	Махачкала	Южный
Республика Ингушетия	4	476	4	Магас	Южный
Республика Калмыкия	76	291	3	Элиста	Южный

Регионы Российской Федерации (на 1 января 2004 г.)	Площадь (тыс. кв. км)	Население (тыс. чел.)	Кол-во городов	Столица	Федеральный округ
Республика Калмыкия	76	291	3	Элиста	Южный
Республика Карелия	172	709	13	Петрозаводск	Северо-Западный
Республика Коми	416	1005	10	Сыктывкар	Северо-Западный
Республика Марий Эл	23	722	4	Йошкар-Ола	Приволжский
Республика Мордовия	26	876	7	Саранск	Приволжский
Республика Саха (Якутия)	3103	949	13	Якутск	Дальневосточный
Республика Северная Осетия — Алания	8	707	6	Владикавказ	Южный
Республика Татарстан	68	3773	20	Казань	Приволжский
Республика Тыва	171	306	5	Кызыл	Сибирский
Республика Хакасия	62	543	5	Абакан	Сибирский
Ростовская область	101	4366	23	Ростов-на-Дону	Южный
Рязанская область	40	1208	12	Рязань	Центральный
Самарская область	54	3218	11	Самара	Приволжский
г. Санкт-Петербург	...	4624	1		Северо-Западный
Саратовская область	100	2644	18	Саратов	Приволжский
Сахалинская область	87	538	18	Южно-Сахалинск	Дальневосточный
Свердловская область	195	4448	47	Екатеринбург	Уральский
Смоленская область	50	1032	15	Смоленск	Центральный
Ставропольский край	67	2726	19	Ставрополь	Южный
Таймырский автономный округ	862	39	1	Дудинка	Сибирский
Тамбовская область	34	1159	8	Тамбов	Центральный
Тверская область	84	1444	23	Тверь	Центральный
Томская область	317	1041	6	Томск	Сибирский
Тульская область	26	1645	21	Тула	Центральный
Тюменская область	1435	3290	28	Тюмень	Уральский
Удмуртская Республика	42	1560	6	Ижевск	Приволжский
Ульяновская область	37	1364	6	Ульяновск	Приволжский
Усть-Ордынский Бурятский автономный округ	22	135	0	нас. пункт Усть-Ордынский	Сибирский
Хабаровский край	789	1427	7	Хабаровск	Дальневосточный
Ханты-Мансийский автономный округ	523	1457	16	Ханты-Мансийск	Уральский
Челябинская область	88	3574	30	Челябинск	Уральский
Чеченская Республика	17	1121	54	Грозный	Южный
Читинская область	432	1144	10	Чита	Сибирский
Чувашская Республика	18	1305	9	Чебоксары	Приволжский
Чукотский автономный округ	738	52	3	Анадырь	Дальневосточный
Эвенкийский автономный округ	768	18	0	пос. Тура	Сибирский
Ямало-Ненецкий автономный округ	750	515	7	Салехард	Уральский
Ярославская область	36	1351	11	Ярославль	Центральный

Таблица 4

Города Российской Федерации

Город	Регион	Население (тыс. чел.)	Город	Регион	Население (тыс. чел.)
Москва	Москва	10 392	Мурманск	Мурманская область	329
Санкт-Петербург	Санкт-Петербург	4624	Сочи	Краснодарский край	328
Новосибирск	Новосибирская область	1413	Смоленск	Смоленская область	322
Нижний Новгород	Нижегородская область	1297	Владикавказ	Республика Северная Осетия — Алания	315
Екатеринбург	Свердловская область	1287	Владимир	Владимирская область	313
Самара	Самарская область	1144	Чита	Читинская область	312
Омск	Омская область	1122	Череповец	Вологодская область	311
Казань	Республика Татарстан	1107	Волжский	Волгоградская область	311

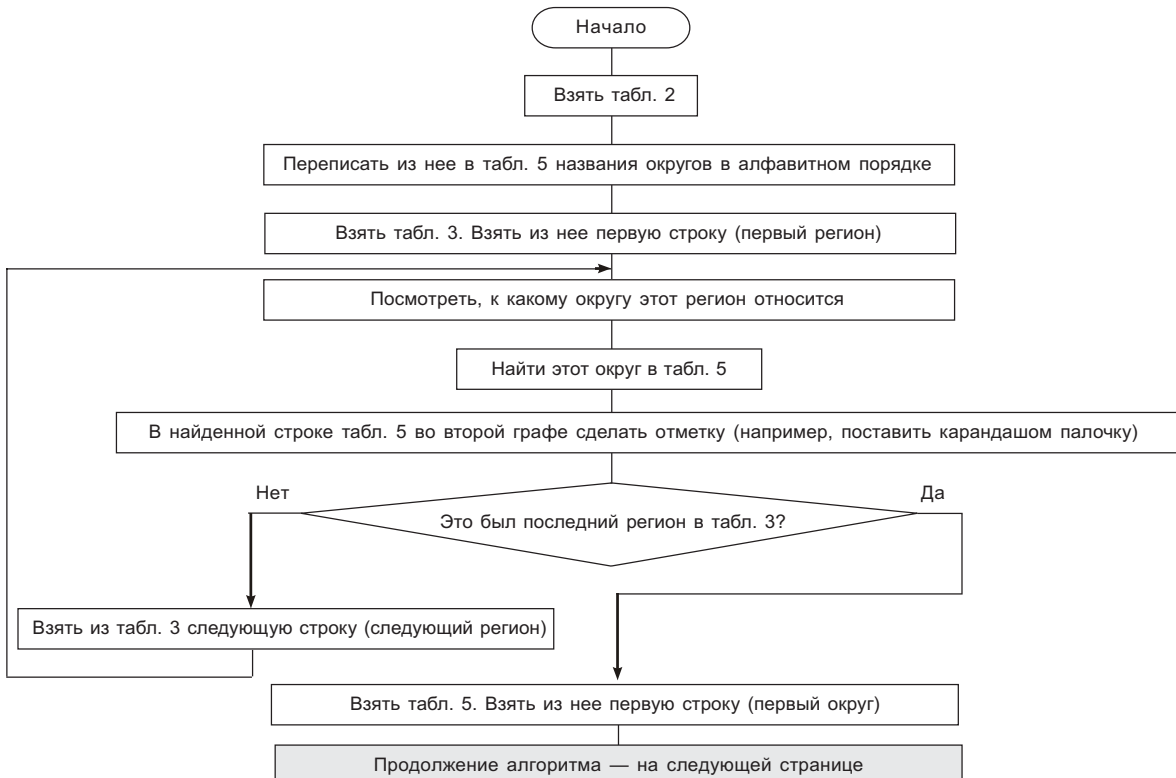
Город	Регион	Население (тыс. чел.)
Челябинск	Челябинская область	1071
Ростов-на-Дону	Ростовская область	1062
Уфа	Республика Башкортостан	1041
Волгоград	Волгоградская область	1004
Пермь	Пермская область	995
Красноярск	Красноярский край	913
Саратов	Саратовская область	865
Воронеж	Воронежская область	850
Краснодар	Краснодарский край	719
Тольятти	Самарская область	704
Барнаул	Алтайский край	636
Ульяновск	Ульяновская область	628
Ижевск	Удмуртская Республика	628
Ярославль	Ярославская область	608
Владивосток	Приморский край	590
Иркутск	Иркутская область	589
Хабаровск	Хабаровский край	580
Новокузнецк	Кемеровская область	548
Оренбург	Оренбургская область	543
Рязань	Рязанская область	517
Пенза	Пензенская область	515
Тюмень	Тюменская область	510
Набережные Челны	Республика Татарстан	509
Липецк	Липецкая область	504
Астрахань	Астраханская область	503
Томск	Томская область	489
Кемерово	Кемеровская область	480
Тула	Тульская область	472
Махачкала	Республика Дагестан	464
Киров	Кировская область	452
Чебоксары	Чувашская Республика	442
Брянск	Брянская область	428
Калининград	Калининградская область	428
Иваново	Ивановская область	424
Магнитогорск	Челябинская область	416
Курск	Курская область	408
Тверь	Тверская область	403
Нижний Тагил	Свердловская область	387
Улан-Удэ	Республика Бурятия	357
Ставрополь	Ставропольский край	355
Архангельск	Архангельская область	353
Курган	Курганская область	339
Белгород	Белгородская область	337
Калуга	Калужская область	332
Орел	Орловская область	331
Рубцовск	Алтайский край	162
Мытищи	Московская область	161
Салават	Республика Башкортостан	159
Люберцы	Московская область	158
Уссурийск	Приморский край	157
Майкоп	Республика Адыгея	157

Город	Регион	Население (тыс. чел.)
Саранск	Республика Мордовия	302
Вологда	Вологодская область	291
Сургут	Тюменская область	290
Тамбов	Тамбовская область	290
Комсомольск-на-Амуре	Хабаровский край	278
Таганрог	Ростовская область	277
Кострома	Костромская область	277
Нальчик	Кабардино-Балкарская Республика	274
Петрозаводск	Республика Карелия	266
Стерлитамак	Республика Башкортостан	265
Дзержинск	Нижегородская область	259
Братск	Иркутская область	258
Йошкар-Ола	Республика Марий Эл	256
Орск	Оренбургская область	250
Ангарск	Иркутская область	246
Нижневартовск	Тюменская область	240
Новоросийск	Краснодарский край	232
Бийск	Алтайский край	229
Сыктывкар	Республика Коми	229
Нижнекамск	Республика Татарстан	227
Прокопьевск	Кемеровская область	221
Шахты	Ростовская область	220
Рыбинск	Ярославская область	220
Благовещенск	Амурская область	220
Старый Оскол	Белгородская область	217
Якутск	Республика Саха (Якутия)	217
Великий Новгород	Новгородская область	214
Грозный	Чеченская Республика	214
Псков	Псковская область	201
Балаково	Саратовская область	200
Северодвинск	Архангельская область	199
Петропавловск-Камчатский	Камчатская область	197
Златоуст	Челябинская область	193
Армавир	Краснодарский край	193
Энгельс	Саратовская область	192
Сызрань	Самарская область	186
Каменск-Уральский	Свердловская область	185
Подольск	Московская область	180
Балашиха	Московская область	175
Южно-Сахалинск	Сахалинская область	174
Березники	Пермская область	171
Королев	Московская область	171
Новочеркасск	Ростовская область	169
Волгодонск	Ростовская область	166
Абакан	Республика Хакасия	165
Миасс	Челябинская область	157
Ковров	Владимирская область	154
Коломна	Московская область	150
Находка	Приморский край	146
Электросталь	Московская область	146
Альметьевск	Республика Татарстан	142

Город	Регион	Население (тыс. чел.)
Химки	Московская область	142
Пятигорск	Ставропольский край	140
Одинцово	Московская область	134
Норильск	Красноярский край	133
Новомосковск	Тульская область	132
Невинномысск	Ставропольский край	132
Первоуральск	Свердловская область	132
Димитровград	Ульяновская область	130
Кисловодск	Ставропольский край	130
Серпухов	Московская область	129
Назрань	Республика Ингушетия	127
Камышин	Волгоградская область	126
Новочебоксарск	Чувашская Республика	125
Муром	Владимирская область	125
Хасавюрт	Республика Дагестан	123
Нефтекамск	Республика Башкортостан	122
Орехово-Зуево	Московская область	122
Черкесск	Карачаево-Черкесская Республика	117
Ногинск	Московская область	117
Ачинск	Красноярский край	117
Елец	Липецкая область	116
Щелково	Московская область	113
Сергиев Посад	Московская область	112
Новокуйбышевск	Самарская область	112
Ленинск-Кузнецкий	Кемеровская область	110

Город	Регион	Население (тыс. чел.)
Арзамас	Нижегородская область	108
Железнодорожный	Московская область	107
Батайск	Ростовская область	106
Кызыл	Республика Тыва	106
Новотроицк	Оренбургская область	106
Обнинск	Калужская область	106
Киселевск	Кемеровская область	105
Великие Луки	Псковская область	104
Элиста	Республика Калмыкия	104
Дербент	Республика Дагестан	103
Ухта	Республика Коми	103
Междуреченск	Кемеровская область	102
Канск	Красноярский край	102
Сарапул	Удмуртская Республика	102
Соликамск	Пермская область	101
Жуковский	Московская область	101
Глазов	Удмуртская Республика	101
Новошахтинск	Ростовская область	101
Усть-Илимск	Иркутская область	100
Тобольск	Тюменская область	91
Нерюнгри	Республика Саха (Якутия)	75
Ржев	Тверская область	64
Нефтеюганск	Тюменская область	110
Октябрьский	Республика Башкортостан	109
Северск	Томская область	108

17. Используя таблицы из предыдущего задания, выполните следующий алгоритм.



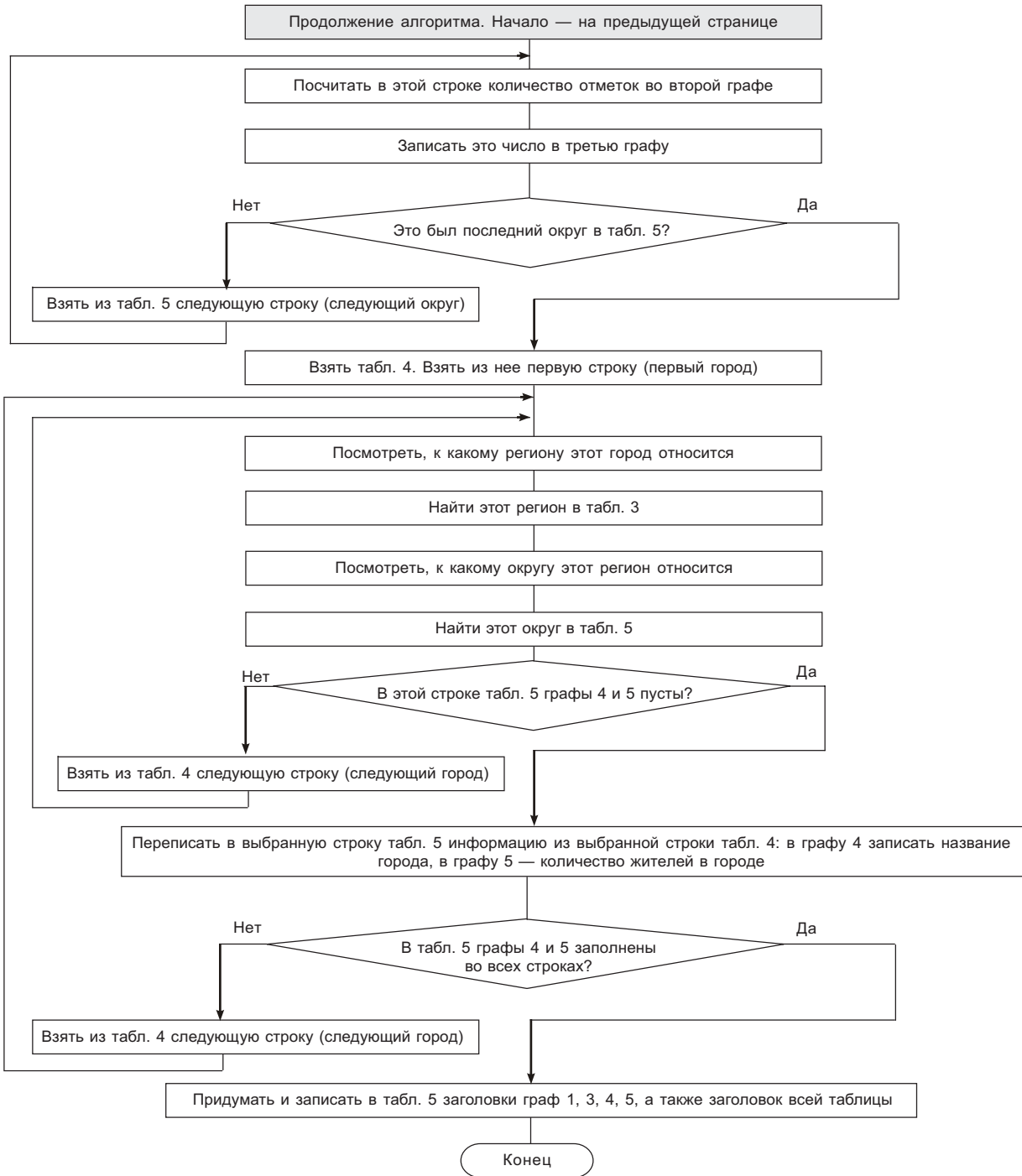


Таблица 5

Продолжение следует

САМАЯ ГОРЯЧАЯ ТЕМА

сегодняшнего российского образования
во всех газетах Издательского дома «Первое сентября»

в январе 2007 года вы получите

ИНФОРМАТИКА

специальный тематический выпуск «ПРОФИЛЬНОЕ ОБУЧЕНИЕ»

В этом номере:

- Ликбез по теме «Информатика и профильная школа»
- Стандарт профильного уровня по информатике
- Результаты седьмого тура методического конкурса «Как это делаю я» по теме «Информатика в профиль». Практический опыт профильного обучения информатике
- Обзор изданных элективных курсов по информатике
- Примерные ответы на профильные билеты (продолжение осенних публикаций)

Шеф-редактор номера А.Л. Семенов, ректор МИОО, д.ф.-м.н., член-корр. РАН

Хотите быть в курсе?
Еще есть время подписаться
на газету «ИНФОРМАТИКА»

Подписной индекс 33371 по каталогу
агентства «Роспечать» «Газеты. Журналы»



При подписке на 6 месяцев — **подарок:** три выпуска
Библиотечки «Первого сентября» серии «Информатика»

В мир информатики

80 (1—15 ноября)

Газета для пытливых учеников
и их талантливых учителей

Моделирование работы ЭВМ с помощью программы Microsoft Excel

Е.А. Еремин,
г. Пермь

Предисловие для учителя

Как известно, современные табличные процессоры пригодны для решения самого широкого круга задач, причем не обязательно чисто вычислительных (см., например, статьи [1—4]). Возможности пользователей еще более возрастают, если электронные таблицы имеют встроенный язык программирования. Широко распространенный в нашей системе образования программный продукт Microsoft Excel, имеющий встроенный язык VBA (*Visual Basic for Application*), не является в этом отношении исключением.

Хотя наиболее часто областью задач все же является экономика [5], для образовательных целей заметный интерес как раз представляют примеры “неэкономического” содержания. Творческие преподаватели разработали большое количество таких примеров; в частности, в Интернете можно отыскать огромное разнообразие подобных моделей, начиная от размножения кроликов [6] до демонстрации осциллограмм напряжения [7]. В связи с этим подбор тематики заданий для моделирования является собой определенную методическую проблему: из широчайшего перечня доступных задач надо выбрать интересные, познавательные, соответствующие уровню знаний учеников и т.д.

В данной статье в качестве одного из достойных внимания вариантов предлагается для углубления знаний учеников в области информатики смоделировать наиболее важные принципы работы ЭВМ. В новом стандарте знаний по школьному курсу информатики [8] (попутно заметим, что он отличается некоторым уменьшением внимания к устройству компьютера) предусмотрено изучение программного принципа

работы ЭВМ, так что имитация процесса исполнения программы есть задача, безусловно, полезная.

Моделирование в данной статье базируется на применении макросов. При изложении предполагается, что читатели знакомы с основами языка Basic, способны набрать простейшую программу макроса и подключить ее к созданной кнопке. В противном случае можно обратиться к весьма распространенным учебникам [9—10] или к статьям [13—14].

В качестве основы для изучения удобно взять наиболее простую из моделей, которые описываются в учебной литературе. Если проанализировать школьные учебники, то с позиции простоты устройства выбор практически неизбежно будет сделан в пользу ЭВМ “Кроха”, которую предложил в своем учебнике информатики [11] коллектив авторов из Екатеринбурга. Данная модель “по духу” близка к ЭВМ первых двух поколений, что делает ее несколько устаревшей, но зато по простоте и наглядности она не имеет себе равных. Поскольку главной целью нашего моделирования является изучение на основе простейших методов наиболее общих принципов работы вычислительной машины, то, по мнению автора, для этих целей “Кроху” можно признать вполне подходящей.

Предисловие для учащихся

Электронные таблицы, к которым относится известная программа Microsoft Excel (их иногда называют “табличными процессорами”), могут быть использованы для решения широкого круга задач, среди которых — моделирование различных процессов. В данной статье мы рассмотрим методику моделирования работы... электронно-вычислительных машин. Сделаем мы это на примере учебной ЭВМ “Кроха”, описанной в одном из учебников информатики. С помощью созданной модели можно проиллюстрировать фундаментальные принципы работы компьютеров. При этом заметим, что описанную в статье методику моделирования можно применить и к более сложной машине — ничто, кроме трудоемкости работы, не может помешать, например, создать в Excel модель реального компьютера на базе процессора фирмы Intel(!).

Моделирование

Учебная ЭВМ “Кроха”

Эта ЭВМ не случайно была названа “Крохой” — в памяти этой машины всего восемь (!) ячеек, и она способна выполнить столько же различных операций. Но, несмотря на столь примитивное устройство, “Кроха” дает весьма правильное представление о наиболее общих принципах работы ЭВМ.

Основу модели “Кроха” составляет оперативная память (оперативное запоминающее устройство — ОЗУ). Кстати, в классической работе по описанию основ работы ЭВМ [12], до сих пор не утратившей своего фундаментального значения, роль организации памяти в конструкции машины в целом также подчеркивалась особо: “Мы начинаем с рассмотрения органа памяти, так как наш опыт показывает, что выбор данного элемента в качестве элементарной ячейки памяти более или менее определяет также и значительную часть организации машины”.

На количество ячеек ОЗУ и их разрядность заметное влияние оказывает структура машинной команды.

Мы будем использовать так называемую “трехадресную систему команд”, которая обычно применялась в первых ЭВМ, как наиболее наглядную и “логичную” (очевидно, что все арифметические операции имеют два операнда и результат). В ней команда состоит из кода операции и трех адресов ОЗУ, с содержимым которых выполняются действия:

Код операции Адрес 1 Адрес 2 Адрес 3

Ясно, что разрядность (длина) команды будет существенным образом зависеть от максимальной длины адреса ячеек. Поскольку, как упоминалось выше, ОЗУ “Крохи” состоит из восьми ячеек, для их нумерации достаточно трех двоичных разрядов ($2^3 = 8$). Кроме того, естественно положить длину кода операции также равную трем битам. В итоге общая длина команды составит 12 двоичных разрядов (три адреса и код операции, по три разряда каждый).

Длину обрабатываемых чисел также удобно принять равной 12 битам.

Для выполнения арифметических операций в ЭВМ “Кроха” служит специальный регистр — *сумматор*. Он символизирует собой *арифметико-логическое устройство* — АЛУ, которое в реальной ЭВМ, разумеется, гораздо сложнее.

Все команды в вычислительной машине должны выполняться автоматически, что обеспечивается отдельным узлом — устройством управления (УУ). В его состав в “Крохе” входит *счетчик команд*, отслеживающий движение по программе и запоминающий номер очередной команды, а также *регистр*

команд, где хранится исполняемая в данный момент команда. Подробнее механизм выполнения программы будет рассмотрен ниже.

Напомним, что АЛУ и УУ в современных компьютерах объединяются в единый блок, который принято называть *процессором*.

Наконец, в качестве устройства вывода у “Крохи” имеется мини-дисплей, на котором могут отображаться значения трех чисел.

Устройства ввода информации в ЭВМ в учебнике [11] не конкретизируются, видимо, по умолчанию полагается, что это некий специализированный пульт управления. Ввод данных в ОЗУ “Крохи” не связан с наличием какой-либо инструкции программы. На первый взгляд это может показаться удивительным, но во многих старых моделях вычислительных машин именно так и было: адрес ОЗУ и его содержимое набиралась на пульте с помощью специальных переключателей, причем машина при этом находилась в остановленном состоянии. Еще более удивительным оказывается тот факт, что подобная “древняя” философия прекрасно согласуется с технологией ввода данных в электронную таблицу: набор идет непосредственно в ее ячейки и затем табличный процессор производит необходимые вычисления.

Полная система команд учебной ЭВМ “Кроха” приведена в следующей таблице:

Операция	Код		Комментарии
	2 с.с.	8 с.с.	
<i>Перетись</i>	000	0	$A3 = A1$
<i>Сложение</i>	001	1	$A3 = A1 + A2$
<i>Деление нацело</i>	010	2	$A3 = A1 \text{ div } A2$
<i>Модуль разности</i>	011	3	$A3 = A1 - A2 $
<i>Переход по =</i>	100	4	при $A1 = A2$ — перейти к $A3$
<i>Умножение</i>	101	5	$A3 = A1 * A2$
<i>Переход по ></i>	110	6	при $A1 > A2$ — перейти к $A3$
<i>Стоп и вывод</i>	111	7	$A1, A2, A3$ на дисплей; стоп

Примечание. В столбце с заголовком “2 с.с.” указаны коды операций в двоичной системе счисления, с заголовком “8 с.с.” — в восьмеричной. $A1, A2$ и $A3$ соответствует значениям, записанным в ОЗУ по адресам Адрес 1, Адрес 2 и Адрес 3 команды (см. выше структуру трехадресной команды).

Содержание арифметических операций над целыми положительными числами очевидно из таблицы. “Кроха” также имеет две команды условного перехода (когда результат последней операции равен нулю или если он больше нуля), которые обеспечивают возможность программирования нелинейных алгоритмов. Наконец, наиболее экзотически

смотрится последняя команда: она не просто останавливает машину, но предварительно выводит содержимое ячеек с указанными адресами на дисплей. И хотя такой команды в реальной ЭВМ по многим причинам не бывает, подобная специфическая универсальная инструкция позволяет уложиться в выбранное количество команд.

Рассмотрим теперь подробнее, каким образом “Кроха” выполняет программу, что обязательно нам потребуется в процессе моделирования. При пуске счетчик команд всегда устанавливается на ноль, так что исполнение программы всегда начинается с нулевого адреса. Далее машина следует приведенному ниже несложному алгоритму.

1. Из адреса ОЗУ, на который указывает счетчик команд, считывается очередная команда — она помещается в регистр команд.

2. Счетчик команд увеличивается таким образом, чтобы получился адрес следующей команды (в “Крохе” его содержимое достаточно увеличить на единицу¹).

3. Считанная в регистр команд машинная инструкция расшифровывается и выполняется; при этом извлекаются операнды, над которыми производится необходимое арифметическое действие или операция сравнения, а результат, как правило, записывается по третьему адресу (исключение составляют команды перехода, когда содержимое параметра Адрес 3 команды заносится в счетчик команд, а также совмещенная с остановкой инструкция вывода).

4. Если выбранная инструкция не является командой остановки, алгоритм повторяется, начиная с пункта 1.

Вследствие особой важности сформулированный выше алгоритм имеет специальное название — его часто называют *основным алгоритмом работы процессора*. Наличие подобного алгоритма является неотъемлемым свойством классической фоннеймановской архитектуры², которая с некоторыми модификациями продолжает использоваться до сих пор.

Нетрудно понять, что благодаря основному алгоритму процессор способен автоматически выполнять последовательно расположенные в памяти инструкции программы. Однако такие (линейные) программы на практике встречаются довольно редко — необходимы еще программы, обеспечивающие разные действия в зависимости от получившихся в ходе обработки результатов (ветвления) и многократное повторение серии одних и тех же команд (циклы). Средствами для реализации двух последних видов программ являются инструкции переходов (см. таб-

личку с системой команд на с. 38). Их суть состоит в том, что на этапе 3 в случае выполнения указанного условия происходит замена текущего содержимого счетчика команд на значение, закодированное в инструкции перехода. Таким путем осуществляется нарушение естественной последовательности команд и происходит переход на любое место программы.

Описанные принципы выполнения программы на ЭВМ “Кроха” и будут служить предметом моделирования в табличном процессоре Excel.

Завершая описание учебной ЭВМ “Кроха”, рассмотрим пример программы для нее. Этот пример пригодится нам в дальнейшем в качестве тестового для проверки работоспособности нашей модели.

Реализуем задачу вычисления значения факториала числа n . Обозначим k рабочую переменную, которая является текущим множителем для расчета факториала и меняется от 1 до n . Начальное значение k придется задавать перед каждым запуском “вручную” в ячейку 4, т.к. для команды пересылки константы 1 из одной ячейки в другую у “Крохи” уже не хватает памяти. То же самое можно сказать и про начальное значение $n!$, которое перед каждым запуском естественно установить равным 1 в ячейке 5.

Не забудем также в ячейку 6 занести число $n + 1$, которое является верхней границей цикла (цикл будет выполняться, пока $k < n + 1$, и, следовательно, завершится после умножения на n). Вся программа с комментариями приведена в таблице на с. 40.

При анализе программы полезно обратить внимание на эффект переполнения, который для быстрорастущего значения факториала достигается довольно быстро. В самом деле, как подробно обсуждается в учебнике [11], максимально допустимое число для 12-разрядной “Крохи” равняется 4095. Таким образом, уже попытка вычислить $7! = 5040$ приведет наш “суперкомпьютер” к непреодолимым вычислительным трудностям.

Методику моделирования работы ЭВМ “Кроха” средствами программы Microsoft Excel мы рассмотрим в следующем выпуске. Здесь же в заключение предлагаем разработать программы решения на этой ЭВМ следующих задач³:

- 1) вычисления значения по некоторой “своей” формуле;
- 2) определения и вывода на экран большего из двух чисел;
- 3) нахождения суммы первых 100 натуральных чисел.

¹ В случае исполнения линейного алгоритма. — Ред.

² Джон фон Нейман, американский математик, разработавший основные принципы устройства и работы компьютера. — Ред.

³ Программы (можно не все) присылайте в редакцию. Все приславшие правильные программы будут награждены дипломами. — Ред.

Адрес	Содержимое	Расшифровка	Комментарий
0	5545	Умножить содержимое адреса (5) на содержимое адреса (4) и поместить результат по адресу (5)	$n! * k ==> n!$
1	1474	Сложить содержимое адресов (4) и (7) и поместить результат по адресу (4)	$k + 1 ==> k$
2	6640	Если содержимое адреса (6) больше содержимого адреса (4), то перейти к адресу 0	Переход при $k < n + 1$
3	7555	Остановить работу, вывести содержимое адресов (5), (5), (5)	Вывод $n!$
4	1	Задается начальное значение k	Рабочая ячейка
5	1	Задается начальное значение $n!$	Результат
6	$n + 1$	Задается значение $n + 1$	Константа
7	1	Задается значение 1	Константа (шаг изменения k)

Литература

1. Попова О.Н. Моделирование сетевых задач средствами MathCad и Excel. / "Информатика и образование", 2003, № 12.

2. Ракитина Е.А., Бешенков С.А., Галыгина И.В., Галыгина Л.В. Решение типовых задач по информационным технологиям. / "Информатика и образование", 2004, № 4.

3. Богуславский А.А., Щеглова И.Ю. Моделирование физических задач в электронных таблицах MS Excel. / "Информатика и образование", 2004, № 7.

4. Юнова Н.Н. Интерактивная лекция на тему "Построение и анализ информационных моделей в среде электронных таблиц Microsoft Excel 97". / "Информатика и образование", 2004, № 10.

5. Мур Д., Уэдерфорд Л.Р. Экономическое моделирование в Microsoft Excel. СПб.: Издательский дом "Вильямс".

6. Сергеев А. Кролики в электронных клетках, или Экспонента как она есть. / "Школьная компьютерра", 11.04.2003, <http://school.computerra.ru/doityourself/25705/>.

7. Мясников В. Моделирование мультивибратора. <http://www.vbstreets.ru/VBA/Articles/66068.aspx>.

8. Стандарт основного общего образования по информатике и информационным технологиям. / "Информатика и образование", 2004, № 4.

9. Шафрин Ю.А. Информационные технологии. М.: Лаборатория Базовых Знаний, 2000.

10. Угринович Н.Д. Информатика и информационные технологии. М.: Лаборатория Базовых Знаний, 2000.

11. Основы информатики и вычислительной техники: Пробное учебное пособие для средних учебных заведений. / А.Г. Гейн, В.Г. Житомирский, Е.В. Линецкий и др. Свердловск: Изд-во Уральского ун-та, 1989.

12. Беркс А., Голдстейн Г., Нейман Дж. Предварительное рассмотрение логической конструкции электронного вычислительного устройства. / "Кибернетический сборник", вып. 9. М.: Мир, 1964.

13. Макросы. / "В мир информатики" № 48 ("Информатика" № 1/2005).

14. Игра "Быки и коровы" в среде Microsoft Excel. / "В мир информатики" № 78 ("Информатика" № 19/2006).

"Ломаем" голову

Три письма

Разведчик-резидент направил своему агенту три письма. В первом письме был листочек с квадратной таблицей:

Э	А	П	Я	Т	З
Р	О	Е	Т	Ы	О
В	Ш	В	О	Ш	Е
А	Р	И	Т	Е	Ф
Р	К	О	Т	Т	С
А	Н	Я	Н		А

в третьем — с таблицей:

Второе письмо, содержащее пояснения по использованию этих таблиц, потерялось. Помогите агенту прочитать послание шефа.

Постоянная Капрекара

“Д.Капрекар — индийский математик. Он мал ростом, но велик разумом и сердцем. Более сорока лет он занимается замечательными исследованиями по занимательной теории чисел... За пределами Индии Капрекар более всего известен как автор открытия, совершенного им более двадцати лет назад⁴. Я имею в виду открытую им “постоянную Капрекара”.

Выберите любое четырехзначное число, в котором не все цифры одинаковы. Расположите цифры сначала в порядке убывания, затем, переставив их в обратном порядке, образуйте новое число и вычитайте новое число из старого. Повторяя тот же процесс с разностями, не более чем за... шагов вы придете к постоянной Капрекара ..., которая будет воспроизводить себя. Производя перестановки цифр и вычитания, нули следует сохранять. Например, начав с числа 2111 и вычитая из него 1112, вы получите 0999. На следующем шаге перестановка цифр даст число 9990, из которого вы вычитите 0999, и т.д.”

Основы программирования на Visual Basic

Н.М. Тимофеева,
г. Обнинск Калужской обл.

Продолжение. Начало см. “В мир информатики” № 69–73, 79 (“Информатика” № 4–8, 20/2006)

Переменные величины в процедурах Visual Basic

Наиболее часто применяемыми в программах на языке Visual Basic являются операторы присваивания. Как известно, они используются для присваивания значения свойству объекта. Несколько примеров:

```
txtСообщение.Text = "Привет"
lblЗаголовок.FontUnderline = True
cmdУдалить.FontSize = 8
```

В первом операторе присваивается значение строкового типа String, во втором — логического (булева) типа Boolean, в третьем — числового типа Numeric.

⁴ Приведенные слова были написаны в конце 80-х годов прошлого века — см. далее об источнике цитаты.

Это отрывок из книги замечательного американского популяризатора науки М.Гарднера “Путешествие во времени” (М.: Мир, 1990).

Можно показать, что множество трехзначных чисел также имеет постоянную, обладающую указанным свойством, — это число ..., а двузначные и n -значные числа при $n = 5, 6, \dots, 25$ — не имеют.

Составьте программу, с помощью которой можно определить постоянную Капрекара для четырехзначных чисел⁵, аналогичную величину для трехзначных чисел, а также максимальное число шагов, за которое получаются эти значения.

Полученные четыре значения, пожалуйста, пришлите в редакцию (адреса указаны в рубрике “Внимание! Конкурс”). Лучшие ответы мы поощрим.

Правильные ответы на задания, опубликованные в статьях “Непростой кубик” и “Рональд, Жак и Джордж” (см. в газету “В мир информатики” № 74 / “Информатика” № 9/2006) представил Гайсин Рашит, г. Уфа, Республика Башкортостан, школа № 18, учитель **Искандарова А.Р.**

Задачник

Школа программирования

Однако очень часто использования одних только свойств объектов в программах недостаточно. Так, в рассмотренной в предыдущем выпуске программе необходимо было получить какое-то случайное целое число и в зависимости от него вывести соответствующее изображение. Для хранения значения этого числа и потребуется переменная. Подобно свойствам, переменные имеют имя (идентификатор) и тип данных (тип данных определяет множество значений, который может хранить переменная).

Так же, как и свойства, переменная в большинстве случаев получает значение с помощью оператора присваивания:

```
Кость1 = Int(Rnd * 6) + 1
```

Кость1 — имя переменной. Этой переменной присваивается значение целого типа (Integer).

```
Адрес = "ул. Киевская, 24"
```

Адрес — имя переменной. Значение, которое присваивается переменной, — строкового типа (String).

Имена переменных

Вы сами решаете, какое имя должно быть у переменной, но существует несколько правил.

⁵ Напомним, что речь идет о числах, в которых не все цифры одинаковы.

1. Имя должно начинаться с буквы.
2. Имя должно содержать только буквы, числа и символ подчеркивания.
3. В имени должно быть не более 40 символов.
4. Имя не должно содержать так называемые “зарезервированные” слова Visual Basic (слова, которые Visual Basic использует в коде).

Примеры допустимых имен переменных: V, diskriminant, Очень_длинное_но_допустимое_имя, День3, Мое_Имя.

Примеры недопустимых имен переменных: 2A (начинается с цифры), Как_дела? (используется недопустимый символ), Then (зарезервированное слово Visual Basic).

Рекомендуется подбирать “содержательные” имена для переменных. Это поможет и вам, и другим лучше понять текст программы.

Переменные и их тип данных

Основные типы данных Visual Basic представлены ниже в таблице.

По умолчанию (если вы явно не установили тип данных) переменным назначается тип данных Variant. Переменные (и значения свойств объектов) с типом данных Variant могут хранить данные большинства других типов.

Объявление переменных

В принципе объявлять переменные в Visual Basic не требуется. Обязательное объявление переменных

программистом может быть применено с помощью специальных установок для окна кода (пункт меню **Tools**, подпункт **Options...**). Убедитесь, что установлен флажок около Require Variable Declaration. Рекомендуется всегда устанавливать этот параметр. Кроме приучения к хорошей практике программирования, это помогает выявлять ошибки⁶.

В процедуре вы объявляете переменную при помощи оператора Dim, задавая имя для переменной и при необходимости указывая ее тип:

```
Dim имя переменной [As тип данных]
```

Примеры:

```
Dim Moe_Имя As String
```

```
Dim Кость1 As Integer, Кость2 As Integer
```

Каждый раз, когда процедура выполняется, оператор Dim обеспечивает “новый старт” для переменной — старые значения теряются. Если вы хотите, чтобы какая-то переменная сохранила свое значение до того момента, когда процедура будет выполняться в следующий раз, эта переменная должна быть объявлена при помощи оператора Static:

```
Static имя переменной [As тип данных]
```

⁶ О возможных ошибках, связанных с отсутствием объявления (описания) переменных величин, см. статью “Игра “Быки и коровы” в среде Microsoft Excel” в газете-вкладке “В мир информатики” № 78 (“Информатика” № 19/2006). — Ред.

Название типа	Особенности данных	Диапазон значений	Размер (представление в памяти), байт
Integer (целый)	Целые числа	От -32 768 до 32 767	2
Byte (байт)	Целые числа	От 0 до 255	1
Long (длинное целое)	Целые числа	От -2 147 483 648 до 2 147 483 647	4
Single (вещественный обычной точности)	Вещественные числа (с точностью до 7 цифр)	От $-3,402823 \cdot 10^{38}$ до $3,4402823 \cdot 10^{38}$	4
Currency (денежный)	Вещественные числа (4 цифры после десятичной точки, 15 перед)	От $-9 \cdot 10^{14}$ до $9 \cdot 10^{14}$ (примерно)	8
String (строковый фиксированной длины)	Строки символов	От 0 до приблизительно 65 400 символов	Длина строки
String (строковый переменной длины)	Строки символов	От 0 до приблизительно 2 миллиардов символов	10 + длина строки
Boolean (логический)	True (Истина) или False (Ложь)	True, False	2
Date (дата)	Значения даты и времени	Даты от 1 января 100 г. до 31 декабря 9999 г. Время от 0:00:00 до 23:59:59	8
Variant (“универсальный”)	Числа Строки символов Значения даты и времени	Аналогично соответствующим типам	16

Область видимости переменных

Переменные, которые объявлены в процедуре с помощью операторов Dim или Static, доступны для использования (как говорят, видимы) только в этой процедуре. Они называются *локальными* переменными. Даже если бы вы объявили переменную с одним и тем же именем в двух разных процедурах, на самом деле они были бы “неизвестны” друг другу.

Вам может понадобиться переменная, которая доступна для использования в нескольких процедурах. В таком случае следует записать оператор Dim с этой переменной не в процедуре, а в разделе кода General (выбирается в левом списке в верхней части окна кода). Такая переменная называется *переменной уровня формы*. Однако переменные уровня формы рекомендуется объявлять только в случае необходимости, все остальные переменные лучше по возможности оставлять локальными.

Если же требуется, чтобы переменная имела еще более широкую область видимости, так, чтобы она распознавалась каждой формой в приложении, она должна быть объявлена с помощью оператора Public в специальной области кода Code module. Но нам нужно многое изучить до того, как перед нами возникнет необходимость использовать такие переменные.

Вопросы и задания для проверки знаний

1. Чем обосновывается рекомендация объявлять все переменные, используемые в приложении?
2. Определите, какие из следующих имен переменных допустимы, а какие нет. В случае недопустимых имен обоснуйте причину:
2ая_Попытка
Имя_нашей_компании
Select
A_L2
3. Напишите объявления для двух локальных переменных с именами Рост и Вес.
4. Какой тип должен быть установлен для переменной, которая хранит имя человека?
5. Напишите объявление для переменной Население, которая может содержать целое число до одного миллиарда.
6. Какие значения могут принимать логические переменные?
7. Напишите объявление для логической переменной с именем Военнообязанный.
8. Какие переменные могут распознаваться только в конкретной процедуре?
9. В каком случае переменная в процедуре объявляется с помощью оператора Static?

Рекуррентные соотношения в Microsoft Excel

Окончание. Начало см. “В мир информатики” № 79 (“Информатика” № 20/2006).

В первой части статьи была описана методика получения последовательностей чисел, построенных на основе так называемых “рекуррентных соотношений” — формул, выражающих очередной член последовательности через один или несколько предыдущих членов в среде электронной таблицы Microsoft Excel, причем с использованием небольшого количества ячеек листа. Был рассмотрен пример расчета с применением рекуррентного соотношения, в котором следующий член определяется только предыдущим элементом.

Описанную методику можно использовать также, когда очередной член последовательности зависит от двух и более предыдущих членов. Примером такой последовательности является последовательность чисел 1, 1, 2, 3, 5, 8, 13, ..., в которой каждый следующий член, начиная с третьего, равен сумме двух предыдущих (так называемая “последователь-

Microsoft Excel углубленно

ность Фибоначчи”). Для нее рекуррентное соотношение имеет вид: $a_i = a_{i-2} + a_{i-1}$, $a_1 = 1$, $a_2 = 1$. Вот как выглядит лист, оформленный для определения n -го члена последовательности Фибоначчи, в исходном состоянии:

	A	B	C
1	Для начала расчетов введите любой текст ->		
2	Член последовательности, предшествующий предыдущему	1	
3	Предыдущий член последовательности	1	
4	Очередной член последовательности	2	
5			

Формулы в ячейках (логику расчетов можно посмотреть в [1]):

- в ячейке B2: =ЕСЛИ(B1="";1;B3);
- в ячейке B3: =ЕСЛИ(B1="";1;B4);
- в ячейке B4: =B2+B3.

Обратим внимание на то, что предельное число итераций, устанавливаемое с помощью меню (Сервис — Параметры, вкладка Вычисления), должно быть равно $n - 3$, где n — номер искомого члена последовательности.

Раз уж речь зашла о предельном числе итераций (расчетов по рекуррентной формуле), то заметим,

что оно может быть ограничено не только числом — количеством итераций, но и так называемой “относительной погрешностью” расчетов. Смысл этого параметра состоит в том, что Excel будет выполнять расчеты до тех пор, пока рассчитываемое значение не изменится на меньше, чем заданная погрешность. Такой способ расчетов используется тогда, когда каждый очередной член последовательности отличается от предыдущего все меньше и меньше и известна погрешность измерений, которой можно пренебречь (и прекратить вычисления).

Задания для самостоятельной работы

1. Последовательность чисел v_1, v_2, v_3, \dots образуется по закону: $v_1 = 3, v_2 = 5; v_i = v_{i-2} * 5 - v_{i-1} * 3$ ($i = 3, 4, \dots$). Получить 20-й член этой последовательности.

2. Последовательность чисел v_1, v_2, v_3, \dots образуется по закону: $v_1 = v_2 = 0; v_3 = 1,5,$

$$v_i = \frac{i+1}{i^2+1} v_{i-1} - v_{i-2} v_{i-3}, i = 4, 5, \dots$$

Получить 14-й, 15-й, ..., 23-й члены этой последовательности.

3. При положительном a рекуррентное соотношение

$$x_i = \frac{x_{i-1}}{2} + \frac{a}{2x_{i-1}}$$

можно использовать для вычисления \sqrt{a} , т.к. элементы последовательности, построенной на таком соотношении, при увеличении i очень быстро приближаются к \sqrt{a} [2]. Вот, например, как выглядит начало этой последовательности при $a = 2$:

$$\begin{aligned} x_1 &= 1; \\ x_2 &= 1,5; \\ x_3 &= 1,4166666667; \\ x_4 &= 1,4142156863; \\ x_5 &= 1,4142135624; \\ x_6 &= 1,4142155624; \\ &\dots \end{aligned}$$

Подготовить лист для определения \sqrt{a} для заданного a с погрешностью 0,0001 в виде:

	A	B	C
1	Задайте число, из которого необходимо извлечь корень ->		
2	Корень из этого числа =		
3			

Принять $x_1 = 1$ (это может быть любое число).

Литература

1. Школа программирования: Тематический выпуск газеты-вкладки “В мир информатики” (“Информатика” № 11/2005).

2. Корнилов В.С. Как ЭВМ извлекает квадратный корень. / “В мир информатики” № 22 (“Информатика” № 10/2004).

Еще раз о фокусе “Дата рождения”

В статье [1] был описан фокус, связанный с отгадыванием даты рождения любого человека, который демонстрируется с использованием калькулятора. Напомним, что при показе фокуса человек, дата рождения которого отгадывается, должен провести определенные расчеты и сообщить результат. Для демонстрации фокуса можно также разработать несложную компьютерную программу. На школьном алгоритмическом языке она имеет вид:

алг Отгадывание_даты_рождения

нач **цел** результат, день, месяц, год,
разность, год_рожд,

лит название_месяца

вывод **нс**, "Сейчас я отгадаю дату"

вывод " (день, месяц и год)
вашего рождения"

Games.exe

```
|Приостановка программы
|до нажатия любой клавиши
```

нц

```
|
кц_при inkey() <> ""
```

вывод **нс**, "Умножьте день своего
рождения на 20"

```
|Приостановка программы до нажатия
|любой клавиши
```

нц

```
|
кц_при inkey() <> ""
```

вывод **нс**, "К полученному произведению
прибавьте 3"

```
|Приостановка программы до нажатия
|любой клавиши
```

...

вывод **нс**, "Сумму умножьте на 5"

```
|Приостановка программы до нажатия
|любой клавиши
```

...

вывод **нс**, "К полученному произведению
прибавьте"

```

Вывод НС, "номер месяца, в котором Вы родились"
|Приостановка программы до нажатия
|любой клавиши
...
Вывод НС, "Умножьте полученное число на 20"
|Приостановка программы до нажатия
|любой клавиши
...
Вывод НС, "К полученному произведению
прибавьте 3"
|Приостановка программы до нажатия
|любой клавиши
...
Вывод НС, "Сумму умножьте на 5"
|Приостановка программы до нажатия
|любой клавиши
...
Вывод НС, "К полученному произведению
прибавьте"
Вывод "число, образованное двумя
последними цифрами года,"
Вывод " в котором Вы родились"
Вывод НС, "Сколько получилось?"
Ввод результат
разность := результат - 1515
день := div(разность, 10000)
месяц := div(mod(разность, 10000), 100)
год := mod(разность, 100)
|Определяем название месяца

```

выбор

```

при месяц = 1: название_месяца := "января"
при месяц = 2: название_месяца := "февраля"
...
при месяц = 12: название_месяца := "декабря"

```

все

```

|Определяем 4-значное значение года рождения
год_рожд := 1900 + год
|Выводим ответ
Очистка_Экрана
Вывод НС, "Вы родились ", день,
название_месяца,
год_рожд, " года"

```

кон

— где Очистка_экрана — стандартная процедура очистки экрана.

Для приостановки программы до нажатия любой клавиши в программах на других языках программирования можно также использовать оператор цикла с постусловием:

- 1) в языке Паскаль:

```

Repeat
Until KeyPressed

```
- 2) в языке Бейсик:

```

DO
LOOP UNTIL INKEY$ <> ""

```

Литература

1. Фокус с днем рождения. / "В мир информатики" № 64 ("Информатика" № 23/2005).



Внимание!
Конкурс

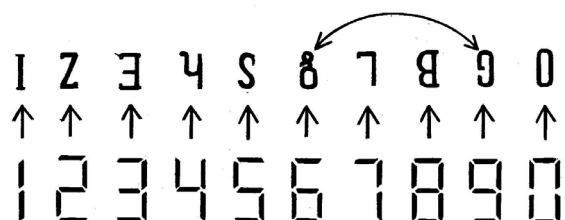
Конкурс № 52
для учащихся

Числа-слова

Если вычислить на калькуляторе выражение

$$170 \cdot \sqrt{123^2 - 12^2 \cdot 91} + 84$$

то получится число 7734. Если перевернуть калькулятор на 180°, то можно увидеть символы, похожие по начертанию на латинские буквы h, E, L и L (немецкое слово hell означает *светлый, ясный*). Для всех десятичных цифр соответствие между ними и буквами изображено на рисунке:



Используя соответствие между цифрами на калькуляторе и буквами латинского алфавита, найдите русский вариант слов, “полученных” в результате расчета приведенных ниже выражений.

1. Для изучающих английский язык:

- 1) $\sqrt{152^2 + 302^2 - 8^2}$;
- 2) $31^2 - 19^2 + \sqrt{259,2 \cdot 1,25}$;
- 3) $(\frac{12,345}{\sqrt{6,25}} + 2) \cdot 1,5 + 26^2 - 2^3$;
- 4) $(85,2 \cdot 45,5 + 1 : 2,5) \cdot 1,85 - 1289 : 20$;
- 5) $(458 \cdot 0,85 + 0,7) \cdot 1,5 + 31,2 \cdot 2,5$;
- 6) $0,875 \cdot 400 + 0,35 \cdot 760$;
- 7) $(222 \cdot 355 + 1190) \cdot 4,5 + 25,6 \cdot 649,0625$;
- 8) $16^3 \cdot 3^2 + 5 \cdot 3^5 - 3$;
- 9) $\frac{22}{0,16} + \frac{368}{0,64} + \frac{130,9}{1,4}$;
- 10) $(544 \cdot 6,25 + 56,5 \cdot 32) \cdot 6,75 - 4 \cdot 37$;
- 11) $(8, 125 \cdot 4,4 + 0,25)^2 + 298,4 \cdot 16,25 + 116 \cdot 10,25$;
- 12) $(50,25^2 - 1 : 16) \cdot 2,8 + \frac{724,5}{\sqrt{1,265625}}$.

2. Для изучающих немецкий язык:

- 1) $153^2 + 155^2 + 157^2 + 17 \cdot 103$;
- 2) $1248 \cdot 84,21 - 1 : 12,5 - 171^2 - 2^{11} - 1$;
- 3) $2^2 \cdot 3^2 \cdot (33^2 - 2 \cdot 5^2)$;
- 4) $((4,5)^4 + 0,9375) \cdot 3^4 + 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 - 15 \cdot 419$;
- 5) $12,5 \cdot (1 : 3,125 + 6,5 \cdot 464)$;
- 6) $6,25 \cdot (4,4 \cdot 15^2 - 3 \cdot 3,143,12)$;

- 7) $(1 : 0,03125 + 1 : 0,008) \cdot 14,85 + 5 \cdot (31^2 - 0,7^2)$;
- 8) $876^2 + 79^2 + 19^2 - \sqrt{20164}$;
- 9) $26 \cdot (\sqrt{240100} + \sqrt{242064} + \sqrt{246016} + \sqrt{248004})$;
- 10) $((101 : 0,015625 - 3) \cdot 9 + 0,76) \cdot 6,25$;
- 11) $33^3 + 14^2 + \sqrt{3,86 \cdot 1,25 - 1,825} \cdot 3$;
- 12) $11^3 + 3,5^3 + 33 \cdot 0,125$.

3. Для изучающих французский язык:

- 1) $29 \cdot 1300 + \sqrt{22,5^2 - 13,5^2}$;
- 2) $7^3 + \sqrt{84,5 \cdot 52 - 6,25 \cdot 507,04}$;
- 3) $21^3 - 17^3 + 12,5 \cdot 65,76 - 2^3$;
- 4) $2^3 \cdot (54,5 \cdot 28 - 555 \cdot 1,6) + 4$;
- 5) $15^2 + 16^2 + 17^2 - 108,5 : 1,75$;
- 6) $18,75 \cdot 2^{11} - 0,75 \cdot 2^5$;
- 7) $18,75 \cdot 2030,72 + 1,83 - \sqrt{4,57 \cdot 1,3 - 2,5921}$;
- 8) $151,3 \cdot 673,4 + 21^2 \cdot 25^2 + 12,5 \cdot 64,2864$;
- 9) $7,3^3 + 3 \cdot 4,363$;
- 10) $23 \cdot (11^2 \cdot 13 + 6)$.

Примечание. В ряде случаев следует рассматривать целую часть от полученного результата.

Ответ отправьте в редакцию до 1 декабря по адресу: 121165, Москва, ул. Киевская, д. 24, “Первое сентября”, “Информатика” или по электронной почте: inf@1september.ru. Пожалуйста, четко укажите в ответе свои фамилию и имя, населенный пункт, номер и адрес школы, фамилию, имя и отчество учителя информатики.

РЕДАКЦИОННАЯ ПОЛИТИКА

Общие вопросы

Вкладка “В мир информатики” выходит в каждом номере, начиная со с. 37 основной газеты. Она адресована учителям, но ее материалы могут использоваться и для самостоятельной работы учеников.

Интерактивность

Дорогие коллеги! Мы обязательно поощрим любую активность ваших учеников. Предложите им поучаствовать в конкурсах (которые специально не слишком сложны) — уверены, ребятам будет приятно увидеть свою фамилию в газете, в списке приславших правильные ответы. Может быть, они заметят какие-то погрешности, неточности в наших материалах, просто захотят поделиться своим мнением? Пусть обязательно напишут. Любую активность мы поощрим.

Можно ли опубликовать свои материалы?

Как и основная газета, вкладка является “открытой” — мы рассматриваем любые материалы, присланные нашими читателями и их учениками. За публикации выплачиваются гонорары в том же размере, что и в основной газете. При подготовке материалов для вкладки следует учитывать ее направленность и объем.

Технические вопросы

Если статья не посвящена конкретному языку программирования, для записи алгоритмов используется Школьный алгоритмический язык. Таким образом, мы стремимся сделать алгоритмы максимально понятными и переносимыми.

ПЕРВОЕ СЕНТЯБРЯ
издательский дом

ИНФОРМАЦИЯ О ПОДПИСКЕ

Подписка на почте

Подписка на первое полугодие 2007 г. на газету «Информатика — Первое сентября» (два номера в месяц по 48 страниц А4) принимается **во всех** отделениях связи России по каталогу «Газеты. Журналы» агентства «Роспечать».

Подписные индексы:

- **32291** для индивидуальных подписчиков (на срок от одного до шести месяцев)
- **32591** для предприятий и организаций (на срок от одного до шести месяцев)

В каталоге указана цена (соответственно 450 и 900 рублей за полугодие), на основании которой работники связи формируют подписную цену (в каждом регионе — своя).

Правило шести месяцев

При подписке на **6 месяцев** по индексу **32291** — подарок:

- три выпуска Библиотечки «Первого сентября» серии «Информатика»

При подписке на **6 месяцев** по индексу **32591** — тройной подарок:

- три выпуска Библиотечки «Первого сентября» серии «Информатика»
- три выпуска Библиотечки «Первого сентября» серии «Воспитание. Образование. Педагогика»
- двенадцать выпусков газеты «Первое сентября».

Ф. СП-1		Министерство связи Российской Федерации «Роспечать»		32291 (индекс издания)							
		Информатика — Первое сентября									
наименование издания		Количество комплектов									
на <u>2007</u> год по месяцам											
1	2	3	4	5	6	7	8	9	10	11	12
×	×	×	×	×	×						
Куда											
(почтовый индекс)		(адрес)									
Кому											
		(фамилия, инициалы)									

Подписаться можно и так

Напрямую через редакцию Издательского дома «Первое сентября»:

- по телефонной заявке, позвонив на номер: (499) 249-47-58
- по почтовой заявке, написав по адресу: ул. Киевская, дом 24, Москва, 121165
- по электронной заявке на адрес: podpiska@1september.ru
- или on-line через сайт Издательского дома www.1september.ru

Почтовая или электронная заявка составляется в свободной форме («Прошу оформить подписку на газету «Информатика» с такого-то месяца на такой-то срок по такому-то адресу и на такое-то имя»).

По заявке вам будет выслана квитанция, которую вы сможете оплатить в любом отделении Сбербанка России. Единственное ограничение: через Издательский дом не принимаются подписки на период менее чем три месяца.

Для организаций: при оформлении подписки через редакцию по безналичной оплате необходимо вместе с заявкой прислать платежные реквизиты.

						ДОСТАВОЧНАЯ КАРТОЧКА					
ПВ	место	ли-тер	на газету		32291 (индекс издания)						
		Информатика — Первое сентября									
		(наименование издания)									
Стоимость	подписки	_____ руб.		Количество комплектов							
	пере-адресовки	_____ руб.									
на <u>2007</u> год по месяцам											
1	2	3	4	5	6	7	8	9	10	11	12
×	×	×	×	×	×						
Куда											
(почтовый индекс)		(адрес)									
Кому											
		(фамилия, инициалы)									

Гении или чудаки?

Окончание. Начало на с. 9

никают трудности в общении со сверстниками, социальная дезадаптация. Проблемы во взаимоотношениях со сверстниками могут быть обусловлены особыми интересами таких детей, концентрацией внимания на научных проблемах, а не на взаимоотношениях с одноклассниками.

Хорошо это или плохо? Ни то и ни другое. Просто необходимо принять тот факт, что талантливые в какой-либо области дети совершенно другие. Надо помнить, что творчески одаренный ребенок вынужден противостоять значительной социальной группе, не приемлющей отклонений. Взрослые редко относятся терпимо к ребенку, который настолько увлечен своими особыми интересами, что игнорирует все остальные области учебной деятельности. Часто приходится слышать возгласы негодования учителей, которые замечают полное отсутствие интереса к своему предмету при повышенном увлечении другими науками. Например, ученик говорит, что литература ему не нужна и не интересна, что она не так структурирована, как математика и т.д. Это воспринимается педагогом как вызов его профессионализму и компетентности. А это неправильно. Такое увлечение ученика какой-либо наукой должно радовать, ведь у каждого учителя, неравнодушного к своему предмету, есть свои "звездочки", которые, возможно, остаются безынициативными и безразличными на других уроках.

Еще одна важная проблема, которая возникает в процессе адаптации учителя и ученика, — это принятие того факта, что ученик может быть более талантлив, чем сам учитель. И не стоит конкурировать с ним, доказывать всему классу и самому ребенку, что вы были и останетесь для него самым умным, непререкаемым авторитетом, эдаким "гуру". Можно только мечтать, чтобы ученик пошел дальше своего учителя. Но ведь дали старт ему именно вы, раскрывали его талант тоже вы. Как в спорте: тренеры, которые дают хо-

рошую начальную подготовку, отбирают таланты на начальных этапах формирования спортивного мастерства, затем передают своих учеников другим тренерам, которые готовят из них уже мастеров высшего класса. Нередко тренеры сами не могут сделать того, чему успешно обучают своих учеников.

При встрече с одаренным ребенком всегда встает вопрос: как с ним работать? В первую очередь необходимо учитывать те особенности, которые мы назвали. То есть не стоит воспринимать желание ребенка отстаивать свою точку зрения как конкуренцию с учителем за право быть неоспоримым авторитетом в классе. Необходимо учитывать, что такие дети очень чувствительны.

Но самое главное — необходимо удовлетворить его интеллектуальный голод. Это, безусловно, дополнительная нагрузка на учителя, который постоянно должен готовить отдельно задания для такого ученика. К сожалению, не всегда у такого школьника есть возможность и желание перейти в другую школу, где обучение ориентировано на ребят, проявляющих особые способности, например, в математике или информатике. Для детей из сельских школ или небольших поселков и городов, где нет физико-математических школ или классов, организуют зимние или летние школы в каникулярное время, вывозят на олимпиады. В стенах школы для развития талантливых ребят организуют кружки, факультативы, творческие мастерские и т.д.

Хотелось бы еще раз акцентировать внимание на том, что в рамках этой статьи мы говорили об одаренных учениках, то есть о тех, кто проявляет особые таланты в узкой области знаний. Таких ребят в школах единицы. Все, что здесь сказано, мало относится к тем детям, которые успешно усваивают многие школьные предметы, в том числе математику и информатику, не обладая ярко выраженными способностями, а в силу того, что уровень развития их интеллекта выше среднестатистического.

**ИЗДАТЕЛЬСКИЙ ДОМ
«ПЕРВОЕ СЕНТЯБРЯ»**
главный редактор —
А.С. Соловейчик

ГАЗЕТЫ
ИЗДАТЕЛЬСКОГО ДОМА
Первое сентября
гл. ред. — Е.В. Бирюкова,
индекс подписки — 32024;

Английский язык
гл. ред. — Е.В. Громушкина,
индекс подписки — 32025;

Библиотека в школе
гл. ред. — О.К. Громова,
индекс подписки — 33373;

Биология
гл. ред. — Н.Г. Иванова,
индекс подписки — 32026;

География
гл. ред. — О.Н. Коротова,
индекс подписки — 32027;

Дошкольное образование
гл. ред. — М.С. Аромштам,
индекс подписки — 33373;

Здоровье детей
гл. ред. — Н.В. Семина,
индекс подписки — 32033;

Информатика
гл. ред. — С.Л. Островский,
индекс подписки — 32291;

Искусство
гл. ред. — М.Н. Сартан,
индекс подписки — 32084;

История
гл. ред. — А.Л. Савельев,
индекс подписки — 32028;

Литература
гл. ред. — С.В. Волков,
индекс подписки — 32029;

Математика
и. о. гл. ред. — Л.О. Рослова,
индекс подписки — 32030;

Начальная школа
гл. ред. — М.В. Соловейчик,
индекс подписки — 32031;

Немецкий язык
гл. ред. — М.Д. Бузоева,
индекс подписки — 32292;

Русский язык
гл. ред. — Л.А. Гончар,
индекс подписки — 32383;

Спорт в школе
гл. ред. — О.М. Леонтьева,
индекс подписки — 32384;

Управление школой
гл. ред. — Я.А. Сартан,
индекс подписки — 32652;

Физика
гл. ред. — Н.Д. Козлова,
индекс подписки — 32032;

Французский язык
гл. ред. — Г.А. Чесновицкая,
индекс подписки — 33371;

Химия
гл. ред. — О.Г. Блохина,
индекс подписки — 32034;

Школьный психолог
гл. ред. — И.В. Вачков,
индекс подписки — 32898.

Гл. редактор
С.Л. Островский
Зам. гл. редактора
А.И. Сенокосов
Редакция
Е.В. Андреева
Д.М. Златопольский (редактор
вкладки "В мир информатики")
Л.Н. Картвелишвили
С.Б. Кишкина
Н.П. Медведева
Ю.А. Первин (редактор вкладки
"Началка")
Корректор
Е.Л. Володина
Дизайн и верстка
Н.И. Пронская

©ИНФОРМАТИКА 2006
Выходит два раза в месяц
При перепечатке ссылка
на ИНФОРМАТИКУ обязательна,
рукописи не возвращаются

Адрес редакции
и издателя:
Киевская, 24, Москва, 121165
тел. 8-499-249-48-96
Отдел рекламы: 8-499-249-98-70

ИНДЕКС ПОДПИСКИ
для индивидуальных подписчиков 32291
комплекта изданий 32744

Тел.: 8-499-249-31-38, 249-33-86. Факс 8-499-249-31-84

Учредитель: ООО "Чистые пруды"

Зарегистрировано в Министерстве РФ по делам
печати. ПИ № 77-7230 от 12.04.2001.
Отпечатано в ОИД "Медиа-Пресса",
ул. Правды, 24, Москва, ГСП-3, А-40, 125993
Тираж 6500 экз.
Срок подписания в печать по графику 05.10.2006.
Номер подписан 05.10.2006.
Заказ № 615521
Цена свободная

Internet: inf@1september.ru
WWW: <http://www.1september.ru>