

# Эпоха перемен

Пару лет назад у меня в классе учился мальчик из Китая. Самый настоящий. Его родители давно работали в Москве, и он к тому времени ходил в нашу школу уже несколько лет. По-русски он научился говорить совершенно свободно, поэтому никаких проблем в общении у нас не было. Не было их в учебе — учеником он был и способным, и прилежным.

Не помню уже точно, по какому поводу я напомнил ему известную китайскую мудрость про эпоху перемен, жить в которую не стоит. Реакция его была неожиданной. Сначала он на некоторое время задумался и сообщил, что, к сожалению, упомянутая мудрость ему не знакома. Ну, бывает... Потом задумался на более продолжительное время и спросил: а почему “не дай Бог”, так ли уж плохо жить в эпоху перемен? Ведь это должно быть **интересно!**

Конечно, все мы хотим жить в спокойной, стабильной и безопасной стране и, применительно к мироустройству, действительно хотели бы избежать революционных перемен. И многие наверняка предпочтут неинтересные новостные выпуски захватывающим. Но так ли уж верна древняя китайская мудрость применительно к нашей профессиональной деятельности?

Мы преподаем замечательный предмет. Очень сложный, динамичный, очень разный. Это доставляет нам массу хлопот и неудобств. Мы не знаем, что делать с детьми, пришедшими из других школ, в которых их учили совсем другому и совсем по-другому, мучаемся, не поспевая за постоянным обновлением аппаратного и программного обеспечения, путаемся в паутине стандартов, программ и учебников. Сравните современный вид нашего предмета с информатикой десятилетней давности. О, ужас! Ничего общего! Как же все изменилось! Но как же здорово, что все изменилось! Ну скажите, вы можете представить информатику, например, без замечательных “флешек” (не железных, а тех, которые \*.fla ☺)? Я — нет.

Впереди новый учебный год. Среди прочих добрых слов мне обязательно хочется пожелать вам, дорогие коллеги, чтобы год этот был для вас интересным. Нам никуда не деться от рутины и текучки, но информатика оставляет школьным учителям необъятное поле для творчества и поиска. Творите и получайте от этого удовольствие!

Здоровья вам и успехов! С новым учебным годом!

*С.Л. Островский,*  
главный редактор

## СОДЕРЖАНИЕ НОМЕРА

### ИНФОРМАЦИЯ

Набор слушателей на курсы повышения квалификации .....	2
Десятые Соловейчиковские чтения .....	47
Фестиваль педагогических идей “Открытый урок”-2006/2007 .....	47
Роботландский университет открывает набор студентов на 2006/2007 учебный год .....	48

### КУРСЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ

<i>В.Ф. Бурмакина, И.Н. Фалина.</i> Как готовиться к тестированию по проверке ИКТ-компетенции школьников. Лекция 1. Цели и задачи проекта по оценке ИКТ-компетентности девятиклассников ....	3–11
--	------

### ТЕЛЕКОММУНИКАЦИИ

<i>А.А. Дуванов.</i> Азы информатики. Выходим в Интернет .....	12–19
--	-------

### СЕМИНАР

<i>Е.А. Еремин.</i> Как компьютер читает файл .....	20–32
--	-------

### “НАЧАЛКА” № 11

<i>Газета-клуб для всех, кто учит информатике маленьких детей</i> <i>Ю.А. Первин.</i> Начальная школа в летнем лагере .....	33–36
---	-------

### “В МИР ИНФОРМАТИКИ” № 76

<i>Газета для пытливых учеников и их талантливых учителей</i> Самым активным читателям — Похвальная грамота газеты “Информатика” .....	37
<i>История информатики</i> Усовершенствованные счеты .....	38

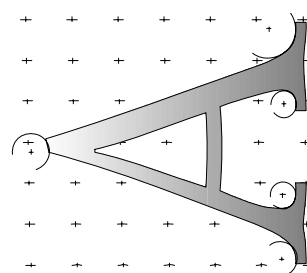
<i>Задачник</i> Ответы, решения, разъяснения .....	39–42
Шестнадцать студентов .....	42
Игра Баше .....	43

<i>“Ломаем” голову</i> Коля, Оля и тетя Поля .....	43
---	----

<i>Внимание! Конкурс</i> Конкурс № 49 для учащихся .....	43
Итоги конкурсов № 42–46 для учащихся .....	43–46

№ 17 (522)

1–15 сентября 2006



Методическая газета для учителей информатики

# ИНФОРМАТИК



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

## ОБЪЯВЛЯЮТ НАБОР СЛУШАТЕЛЕЙ НА КУРСЫ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ НА 2006/2007 УЧЕБНЫЙ ГОД, ПЕРВЫЙ ПОТОК

Курсы проводятся в режиме дистанционного обучения (взаимодействие со слушателями производится посредством обычной или, при наличии у слушателя возможности, электронной почты). Продолжительность обучения — 7 месяцев, нормативный срок освоения учебного материала — 72 часа. Лекционный материал (8 лекций) и контрольные (2 работы) будут публиковаться на страницах газеты «Информатика» (для курса 07-007) или отправляться по почте (для остальных курсов). Итоговую работу слушатели будут выполнять в своих учебных заведениях.

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.

**В 2006/2007 учебном году мы предлагаем четыре курса по вашей специальности:**

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»
07-007	И.Н. Фалина, В.Ф. Бурмакина. Как готовиться к тестированию по проверке ИКТ-компетенции школьников

**Мы также предлагаем один 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 апреля по 30 сентября 2006 г. После регистрации вам будет выслан комплект документов с правилами обучения и счетом для оплаты. Вы оплатите счет лишь в том случае, если вас устроят предлагаемые условия (факт подачи заявки ни к чему не обязывает).

**ЗАЯВКА** Прошу выслать мне комплект документов для зачисления на курсы повышения квалификации. 07-17

**ФАМИЛИЯ**

**ИМЯ**

**ОТЧЕСТВО**

**ИНДЕКС**

**АДРЕС**

Телефон (с кодом города): ( ) \_\_\_\_\_

Электронный адрес (если есть): \_\_\_\_\_

Место работы: \_\_\_\_\_

Должность: \_\_\_\_\_ Стаж работы по специальности: \_\_\_\_\_

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

**Я хочу пройти обучение по курсам (укажите коды выбранных вами курсов):**

— 
     
  — 
     
  — 
     
  —

Если вы обучались в 2005/2006 году на наших курсах, укажите, пожалуйста, ваш идентификатор:

**Заявки следует направлять по адресу: ул. Киевская, д. 24, г. Москва, 121165,  
 Педагогический университет «Первое сентября». Справки по тел.: (499) 249-47-82**

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



**БУРМАКИНА Вероника Федоровна** — руководитель проекта "Разработка инструмента для оценки ИКТ-компетентности школьников", доцент кафедры компьютерной графики и дизайна МПГУ.

**ФАЛИНА Ирина Николаевна** — кандидат педагогических наук, доцент кафедры информатики СУНЦ МГУ им. М.В. Ломоносова, автор учебных и методических пособий по информатике.



### Концепция курса

И как часто мы слышим: стоит ли тратить время на изучение конкретного текстового редактора, если через несколько лет нынешнему ученику придется работать с другим редактором?!

Вопрос в такой формулировке возникает в связи с тем, что мы учим конкретным навыкам работы с конкретным программным средством. И действительно, нужны ли нам сейчас те навыки и умения, которые мы приобрели при работе в Lexicon'e? Но если мы будем учить тому, как представить одну и ту же информацию, например, таблицу изменения цен по коммунальным платежам для разных групп населения, то акцент будет сделан на способах подачи материала. Конечно, ученик в этом случае должен будет знать тот же объем материала об инструментах текстового редактора, но... Текстовый редактор будет выступать в роли инструментария, учить школьника будут другому — способам передачи информации в зависимости от аудитории. Этот пример показывает, что необходимо переходить от подхода выработки ЗУНов к компетентностному подходу.

В рамках проекта "Информатизация системы образования" (ИСО) предусмотрена регулярная (ежегодная) оценка изменения уровня компетентности школьников в области использования ИКТ в регионах проекта ИСО (Красноярский край, Республика Карелия, Пермская область, Челябинская область, Калужская область, Ставропольский край, Хабаровский край).

Цель нашего курса — познакомить учителей информатики с методикой обучения компетентностному подходу овладения информационно-коммуникационными технологиями.

### Учебный план

№ газеты "Информатика"	Учебные материалы
17/2006	<b>Лекция 1.</b> Цели и задачи проекта по оценке ИКТ-компетентности девятиклассников. Цели и метод тестирования. Описание когнитивных компетентностей, оцениваемых тестом. Структура теста.
18/2006	<b>Лекция 2.</b> Описание познавательных деятельности, составляющих компетентность "определение информации". Как научить школьников строить информационные запросы, направленные на достижение конкретной цели?
19/2006	<b>Лекция 3.</b> 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	<b>Лекция 4.</b> 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? <b>Контрольная работа № 1.</b>
21/2006	<b>Лекция 5.</b> 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	<b>Лекция 6.</b> 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	<b>Лекция 7.</b> 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. <b>Контрольная работа № 2.</b>
24/2006	<b>Лекция 8.</b> 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?
<b>Итоговая работа.</b> Итоговая работа должна быть отправлена в Педуниверситет "Первое сентября" до 28 февраля 2007 г.	

## Лекция 1.

# Цели и задачи проекта по оценке ИКТ-компетентности девятиклассников

Отличительной чертой современного общества становится все возрастающая изменчивость окружающего мира. Впервые в истории человечества поколения вещей, процессов, идей сменяются быстрее, чем поколения людей. При этом изменчивость проявляется через небывалое прежде многообразие, затрагивающее практически все стороны общественной жизни. В качестве примеров можно привести:

- 1) изменение функциональности, мощности, внешнего вида ЭВМ;
- 2) изменение функциональных характеристик телефона (или того предмета, который все еще называется телефоном);
- 3) изменение методов медицинской диагностики;
- 4) новые технологии в стоматологии.

Все эти примеры обладают по крайней мере одной общей характеристикой: изменения стали возможны (или были вызваны) в результате передачи и обработки специалистами в сжатые сроки больших объемов информации.

Динамичные изменения, происходящие в обществе, во всей окружающей человека среде, породили различные наименования современного общества: “постиндустриальное общество”, “постцивилизация”, “технотронное общество” и т.п. Однако наиболее широко используется термин “информационное общество”.

### Необходимость специальной подготовки человека к жизни в информационном обществе

Вхождение человеческой цивилизации в информационное общество предъявляет принципиально новые требования к системе образования. Один из ведущих специалистов по информационной культуре Н.И. Гендина (директор НИИ информационных технологий, Кемеровский государственный университет культуры и искусств) [2] пишет, что характер изменений, происходящих в системе образования, отражается в понятии “новая парадигма образования”. Новая образовательная парадигма — это своего рода стратегия “образования для будущего”. Суть новой парадигмы образования характеризуется следующими факторами:

- смещение основного акцента с усвоения значительных объемов информации, накопленной впрок, на овладение способами непрерывного приобретения новых знаний и умения учиться самостоятельно;

- освоение навыков работы с любой информацией, с разнородными, противоречивыми данными, формирование навыков самостоятельного (критичного), а не репродуктивного типа мышления;

- дополнение традиционного принципа “формировать профессиональные знания, умения и навыки” принципом “формировать профессиональную компетентность”.

Анализ характеристик, присущих информационному обществу, позволяет выделить проблему *специальной подготовки человека к жизни в информационном обществе*, что, в свою очередь, требует кардинальных изменений в системе образования. Одним из путей решения этой проблемы в нашей стране является проект “Информатизация системы образования” (ИСО), проводимый Правительством Российской Федерации. Реализация проекта возложена на Национальный фонд подготовки кадров (НФПК).

Проект ИСО направлен на поддержку реализации Концепции модернизации российского образования на период до 2010 года и содействие в обеспечении доступности, качества и эффективности образовательных услуг в системе общего и начального профессионального образования на основе использования информационных и коммуникационных технологий (ИКТ).

Одним из индикаторов успешности проекта ИСО является его влияние на формирование информационной и коммуникационной компетентности выпускников девятого класса, которые завершают обучение по программе основной школы. Этот индикатор позволяет косвенно оценить влияние изменений в содержании и методах учебной работы в основной школе, которые связаны с внедрением ИКТ в практику учебной работы, с реальным уровнем использования и востребованности новых информационных технологий в образовательном процессе. Для определения информационной и коммуникационной компетентности девятиклассников в рамках проекта ИСО было выделено новое направление — “Разработка инструмента для оценки компетентности школьников в области ИКТ”. Рабочая группа проекта начала свою деятельность в ноябре 2005 года. Руководителем проекта является Бурмакина Вероника Федоровна, научным консультантом — Марк Зелман. Концепцию проекта и направления его развития определяют следующие нормативные документы:



1. Отчет “ИКТ-компетентность в мировой практике. Показатель ИКТ-компетентности учащихся и работников образования как индикатор результативности Проекта ИСО”, подготовленный экспертно-аналитическим центром НФПК.

2. Отчет “Report on ICT Literacy and Standards”, подготовленный консультантом МБРР Марком Зелманом в апреле 2004 г.

3. Концепция проекта “Информатизация системы образования”. Методические материалы, 2006 г.

В рамках нового проекта предусмотрена регулярная (ежегодная) оценка уровня компетентности девятиклассников в области использования ИКТ в регионах проекта ИСО (Красноярский край, Республика Карелия, Пермская область, Челябинская область, Калужская область, Ставропольский край, Хабаровский край). Первая проверка уровня ИКТ-компетентности была проведена в апреле—мае 2006 года. В семи вышеперечисленных регионах было протестировано около 4 тысяч девятиклассников.

### Цели проекта по оценке ИКТ-компетентности девятиклассников

Информационно-коммуникационная компетентность<sup>1</sup> школьников в рамках проекта “Информатизация системы образования” определяется как способность учащихся использовать информационные и коммуникационные технологии для доступа (*access*) к информации, ее опознавания-определения (*define*), организации (*integrate*), обработки (*manage*), оценки (*evaluate*), а также ее создания-производства (*create*) и передачи-распространения (*communicate*), которая достаточна для того, чтобы успешно жить и трудиться в условиях информационного общества, в условиях экономики, которая основана на знаниях (*knowledge based economy*).

Особо отметим, что формирование информационно-коммуникационной компетентности рассматривается не только (и не столько) как формирование технологических навыков. Одним из результатов процесса информатизации школы должно стать появление у учащихся способности использовать современные информационные и коммуникационные технологии для работы с информацией как в учебном процессе, так и для иных потребностей.

При определении компетентности школьников в области использования ИКТ акцент делается прежде всего на оценке сформированности соответствующих обобщенных познавательных навыков (умственных навыков высокого уровня). Для оценки сфор-

мированности таких навыков необходим специализированный инструмент, который позволяет оценить демонстрируемые школьниками способности работать с информацией в ходе решения специально подобранных задач (в контролируемых условиях), автоматизировать процедуру оценки уровня ИКТ-компетентности учащихся.

Таким образом, одной из главных задач нового проекта является разработка программного инструмента, позволяющего оценивать, как девятиклассник умеет работать с информацией, представленной в электронном виде. В этом программном инструменте можно выделить 3 составные части:

1) специально разработанная тестовая среда, имитирующая работу почтовой программы, браузера, текстового редактора и т.п.;

2) банк заданий, позволяющих проверять ИКТ-компетентность;

3) система сбора, хранения и анализа результатов тестирования.

Процедура проведения измерений ИКТ-компетентности называется тестированием. В ходе этой процедуры учащиеся выполняют последовательность контрольных заданий, которые в совокупности образуют тест. Тексты (или описания) заданий естественно называть контрольно-измерительными материалами (КИМ). Тест состоит, как правило, из нескольких типов заданий. Будучи встроены в программную оболочку инструмента, задания превращаются в автоматизированный тест.

На основании результатов тестирования учащимся предполагается выдавать сертификат установленного образца, который будет фиксировать уровень ИКТ-компетентности, достигнутый школьником. В соответствии с распорядительными документами, принятыми в регионах проекта ИСО, сертификат будет учитываться в том числе при поступлении учащегося в учреждения профессионального образования с ИКТ-направленностью, а также при продолжении профильного обучения в классах с ИКТ-направленностью.

Для достижения основной цели проекта — разработки специализированного инструмента для автоматизированной оценки ИКТ-компетентности — были поставлены следующие задачи: анализ аналогичных исследований достижений учащихся, проводимых в России и за рубежом; уточнение понятия ИКТ-компетентности и на основе его уточнение спецификаций на разрабатываемые контрольные (тестовые) задания.

### Мониторинг достижений учащихся в России и других странах

Гигантский поток информации, которую сегодня предоставляют системы информации и телекомму-

<sup>1</sup> Детализация понятия ИКТ-компетентности дана в статье В.Ф. Бурмакиной, И.Н. Фалиной “Начало проекта по оценке ИКТ-компетентности девятиклассников”, Информатика № 2/2006.

никации, обостряет проблему подготовки учащихся к жизни в информационном обществе. Осознавая это, во многих странах придается приоритетное значение подготовке учащихся для работы с информацией. Для оценки результатов проводимой работы в некоторых странах уже проводятся исследования, подобные проверке ИКТ-компетентности девятиклассников.

### США, страны Европы

С 1999 года в США компания ETS проводит оценивание ИКТ-компетентности (ИКТ-грамотности в зарубежной терминологии) у студентов колледжей (16–17 лет). Аналогичное тестирование проводится в Бельгии.

При обсуждении проблемы ИКТ-компетентности зарубежными специалистами широко используется термин “грамотность” (*literacy*). Отечественные специалисты используют перекрывающиеся по содержанию понятия “грамотность”, “культура”, “компетентность”. Термин “грамотность” в широком смысле слова претерпел кардинальные изменения по своему содержанию. Если раньше “грамотным” был человек, который научился писать и читать, то сегодня этот термин употребляется гораздо шире. В современном обществе уметь читать и писать явно недостаточно, и требования как к юным гражданам, так и ко взрослым все возрастают. Уметь учиться в течение всей жизни становится главным умением и качеством (основой) новой грамотности.

### Российские библиотеки

Проблема подготовки учащихся к жизни в веке информации связана не только с приобретением ими новой “информационной” грамотности. Она значительно сложнее. В России традиционно развитием и мониторингом информационной грамотности (информационной культуры) занимались и занимаются библиотеки. В.П. Чудинова (Российская государственная детская библиотека) [1] отмечает значительные изменения таких понятий, как “читательская грамотность”, “медиаграмотность”, “компьютерная” и “информационная” грамотности, что с необходимостью приводит к новым формам работы специалистов библиотечного дела.

### Международные исследования PISA

Значимым примером программы оценивания общей компетентности учащихся является “Программа международной оценки обучающихся” (PISA).

Как помнят многие из нас, Советский Союз был довольно закрытым государством. В связи с тем, что наша страна не участвовала в ряде сравнительных международных исследований, многие аспекты нашей жизни было невозможно оценить.

С начала 90-х годов Министерство образования РФ совместно с Российской академией образования принимает участие в сравнительных международных исследованиях качества образования (тестировались девятиклассники в 2000-м и 2003 гг.). Проведение второго цикла международного исследования PISA в 2003 году было направлено на получение данных, необходимых для:

- сравнительной оценки функциональной грамотности 15-летних учащихся разных стран в области математики, чтения, естествознания и решения проблем;
- оценки динамики состояния функциональной грамотности учащихся в области математики, чтения и естествознания за последние три года;
- выявления факторов, позволяющих объяснять различия в результатах учащихся.

Отметим, что в тестировании PISA девятиклассникам предлагаются задания по математике, естествознанию и функциональному чтению. При этом задания по чтению (работа с текстом) проверяют сформированность основных познавательных деятельностей, необходимых для работы с информацией.

Результаты этих исследований показывают [7], что уровень подготовки школьников по естественно-математическим предметам устойчиво превышает средние международные показатели. Российские школьники также не имеют существенных различий в общеобразовательной подготовке со своими сверстниками в большинстве развитых стран мира<sup>2</sup>.

Вместе с тем в этих исследованиях выявлено, что при наличии достаточно высоких предметных знаний и умений российские школьники испытывают затруднения в применении этих знаний в ситуациях, близких к повседневной жизни, а также в работе с информацией, представленной в различной форме, характерной для средств массовой информации.

Анализ результатов исследования PISA показал, что в практике обучения российские учащиеся не встречаются (или встречаются крайне редко, особенно в рамках одной проверочной работы):

- с заданиями, содержащими большой объем как текстовой информации, так и информации, предъявляемой в виде таблиц, диаграмм, графиков, рисунков, схем;
- с заданиями, составленными на материале из разных предметных областей, для правильного выполнения которых надо интегрировать разнообразные знания, использовать общеучебные умения, отбирать и использовать адекватные описываемой ситуации способы размышления, анализа, обоснований, коммуникации и т.п.;

<sup>2</sup> Исключение составляет лидирующая в мире группа стран Юго-Восточной Азии и Тихоокеанского региона (Япония, Южная Корея, Сингапур).

- с заданиями, в которых неясно, к какой области знаний надо обратиться, чтобы определить способ действий или информацию, необходимые для постановки и решения проблемы;

- с заданиями, требующими привлечения дополнительной информации (в том числе выходящей за рамки описанной в тексте задания ситуации), или, напротив, с заданиями, содержащими избыточную информацию и “лишние данные”;

- с комплексными или структурированными заданиями, состоящими из нескольких взаимосвязанных вопросов;

- с большим числом заданий разной тематики и разных форматов, требующих разных форм записи ответа (выбора ответа, записи слова или числа, краткого или развернутого обоснования и т.п.), в одной работе, которую надо выполнить за ограниченное время.

#### Тестирование ECDL и Microsoft

О современном состоянии ИКТ-компетентности в России информации недостаточно. Программы тестирования ECDL и Microsoft по определению уровня ИКТ-грамотности позволяют говорить об оценке только технологических навыков владения компьютерными технологиями и знаний конкретных программных продуктов.

#### Определение ИКТ-компетентности

Развитие проекта по оценке ИКТ-компетентности должно начинаться с понимания, что, прежде чем ИКТ-компетентность можно будет эффективно

измерить, надо четко представлять, что же мы хотим измерять, т.е. надо дать четкое определение ИКТ-компетентности.

Вначале рассмотрим, что стоит за термином “компетентность”, который начинает активно использоваться в отечественной педагогике и в официальных документах Министерства образования и науки. Круг компетентностей, которым следует учить сегодняшних школьников, не определен окончательно, как и само понятие компетентности [9]. За основу понятия “компетентностный человек” взяты способность человека брать на себя ответственность при решении возникающих проблем, проявлять самостоятельность в постановке задач и их решении. Для каждого предмета вырабатывается свое понятие компетентности.

Определение ИКТ-компетентности вводится на базе определения ИКТ-грамотности [9].

*ИКТ-грамотность* — это использование цифровых технологий, инструментов коммуникации и/или сетей для получения доступа к информации, управления ею, ее интеграции, оценки и создания для функционирования в современном обществе.

Под *ИКТ-компетентностью* подразумевается уверенное владение учащимися всеми составляющими навыками ИКТ-грамотности для решения возникающих вопросов в учебной и иной деятельности, при этом акцент делается на сформированность обобщенных познавательных, этических и технических навыков.

Структуру ИКТ-компетентности составляют следующие познавательные навыки (когнитивные действия):

<b>Определение</b> (идентификация)	Умение точно интерпретировать вопрос
	Умение детализировать вопрос
	Нахождение в тексте информации, заданной в явном или в неявном виде
	Идентификация терминов, понятий
	Обоснование сделанного запроса
<b>Доступ</b> (поиск)	Выбор терминов поиска с учетом уровня детализации
	Соответствие результата поиска запрашиваемым терминам (способ оценки)
	Формирование стратегии поиска
	Качество синтаксиса
<b>Управление</b>	Создание схемы классификации для структурирования информации
	Использование предложенных схем классификации для структурирования информации
<b>Интеграция</b>	Умение сравнивать и сопоставлять информацию из нескольких источников
	Умение исключать несоответствующую и несущественную информации
	Умение сжато и логически грамотно изложить обобщенную информацию
<b>Оценка</b>	Выработка критериев для отбора информации в соответствии с потребностью
	Выбор информационных ресурсов согласно выработанным или указанным критериям
	Умение остановить поиск

<b>Создание</b>	Умение вырабатывать рекомендации по решению конкретной проблемы на основании полученной информации, в том числе противоречивой
	Умение сделать вывод о нацеленности имеющейся информации на решение конкретной проблемы
	Умение обосновать свои выводы
	Умение сбалансированно осветить вопрос при наличии противоречивой информации
	Структурирование созданной информации с целью повышения убедительности выводов
<b>Сообщение (передача)</b>	Умение адаптировать информацию для конкретной аудитории (путем выбора соответствующих средств, языка и зрительного ряда)
	Умение грамотно цитировать источники (по делу и с соблюдением авторских прав)
	Обеспечение в случае необходимости конфиденциальности информации
	Умение воздерживаться от использования провокационных высказываний по отношению к культуре, расе, этнической принадлежности или полу
	Знание всех требований (правил общения), относящихся к стилю конкретного общения

### Пример тестового задания “Языки программирования”

Приведем текст 5-минутного задания на проверку сформированности навыков “сообщение” (“передача”) и “оценка”.

*Полный сценарий* (его ученик видит на экране)

Ваша любимая учительница по информатике пригласила вас сделать доклад перед семиклассниками на тему “Современные языки программирования”. Вы очень серьезно подошли к поручению, прочита-

ли “горы” литературы, но, как всегда, для подготовки хорошего доклада не хватает нескольких часов. Вы разослали краткий текст своего доклада трем друзьям и попросили сделать один (но хороший!) слайд, который вы планируете вставить в свою презентацию. Друзья не подвели, они прислали по электронной почте ответы. Ваша задача — выбрать лучший слайд, который будет понятен ученикам 7-го класса и поможет вам сделать хороший доклад.

Далее на экране представлено окно электронной почты

**От:** kalabazz@aha.ru

**Кому:** Serg\_ferst@mail.ru, Тре-Ска@yandex.ru, oform@arg.ru

**Тема:** нужен слайд

**Текст письма**

Ребята, зашиваюсь, помогите. Нужен один слайд для презентации по языкам программирования. Ниже краткое содержание моего доклада, вот к этой-то части и нужен слайд.

В настоящее время существует большое число языков программирования, которое не перестает расти, хотя надо заметить, что данный процесс замедлился. Языком программирования принято называть формально определенный язык для записи программы (т.е. содержащий средства описания данных и алгоритмов). Языки программирования делятся на два класса: процедурные и непроедурные.

Процедурные (императивные) языки — это языки операторного типа. Описание алгоритма на этом языке имеет вид последовательности операторов. Характерным для процедурного языка является наличие оператора присваивания (БЕЙСИК, ПАСКАЛЬ, СИ).

Непроцедурные языки разделяются на три группы языков программирования: функциональные, логические и объектно-ориентированные.

Функциональные языки основаны на идее лямбда-исчисления и теории рекурсивных функций. Программа, написанная на функциональном языке, состоит из неупорядоченного набора уравнений, определяющих функции и значения, которые задаются как функции от других значений (ЛИСП, РЕФАЛ).

Логические языки основаны на использовании символической логики. Наиболее широкоиспользуемый язык — ПРОЛОГ.

Объектно-ориентированные языки — это языки, в которых понятия процедуры и данных, используемых в обычных системах программирования, заменены понятиями “объект” и “сообщение”. В отличие от процедуры, которая описывает, как должна выполняться обработка, сообщение только определяет, что желает выполнить отправитель и что должно произойти. Наиболее развитым языком программирования считается СМОЛТОК, к этому классу языков относятся также Java, C++, Object Pascal.

Best regards,  
kalabazz

mailto: kalabazz@aha.ru



Смена экрана (На экране представлены три раскрытых окна электронной почты)

### Письмо 1

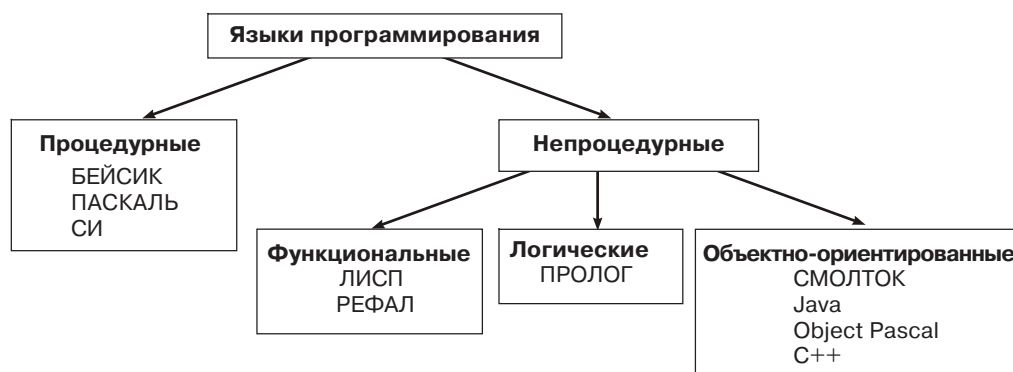
От: *Serg\_ferst@mail.ru*  
 Кому: *kalabazz@aha.ru*  
 Тема: Re [2]: нужен слайд  
 Текст письма



### Письмо 2

От: *Тре-Ска@yandex.ru*  
 Кому: *kalabazz@aha.ru*  
 Тема: удачи!  
 Текст письма

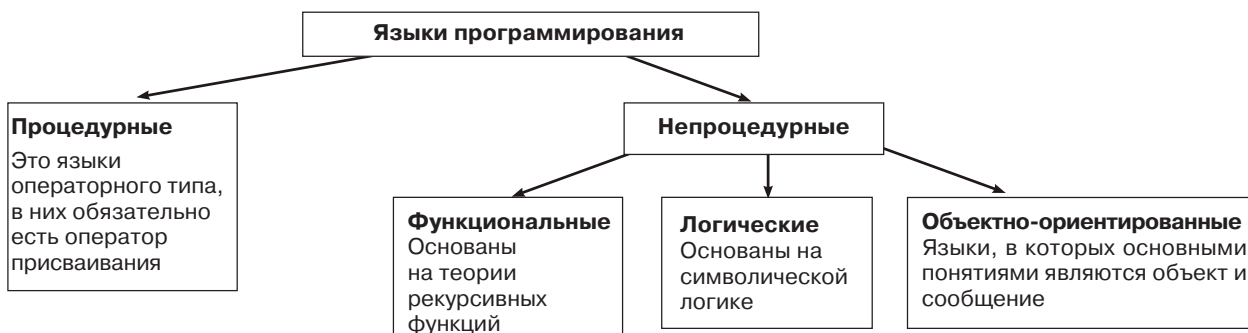
#### КЛАССЫ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ



### Письмо 3

От: *oform@arg.ru*  
 Кому: *kalabazz@aha.ru*  
 Тема: Re [2]: нужен слайд  
 Текст письма

#### КЛАССЫ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ



**Далее на экране**

Выберите вариант слайда и обоснуйте свой ответ.

- Слайд из письма 1
- Слайд из письма 2
- Слайд из письма 3

- Информация представлена полно
- Информация понятна ученикам 7-го класса
- Информация изложена научным языком
- Информация представлена кратко
- Заголовок слайда отражает цель и содержание слайда

**Конец задания**

Попробуйте выполнить это задание, оцените, смогли ли ваши ученики выполнить это задание успешно. При выполнении теста ученик видел это задание как бы в среде почтовой программы. От него требовались технические навыки работы с почтовой программой (специально разработанная среда имитирует наиболее распространенные почтовые программы, но более дружелюбна; в ней нет “лишних” кнопок: даже если ученик первый раз работает с “почтовой” программой, то он сможет достаточно быстро понять, как она устроена).

Но цель данного задания (как, впрочем, и всех остальных) — проверить, как ученик может работать с информацией в “цифровом” формате. От него требуется так представить информацию, чтобы семиклассники поняли, о чем идет речь. Кроме того, от него требуется не только выбрать подходящий способ представления информации (слайд), но и обосновать свой выбор. Заметим, что школьники, как правило, с этим заданием справлялись успешно.

### Результаты первого проведенного тестирования по оценке ИКТ-компетентности девятиклассников

В соответствии с приведенным выше определением ИКТ-компетентности были разработаны тестовые задания, проверяющие одно или несколько когнитивных умений. Тест был рассчитан на 2 часа, он состоял из 14 вопросов: одиннадцать 5-минутных вопросов (на проверку одного умения), два 15-минутных вопроса (на проверку двух или трех умений) и один 30-минутный вопрос. Примеры вопросов вы можете также посмотреть в [8].

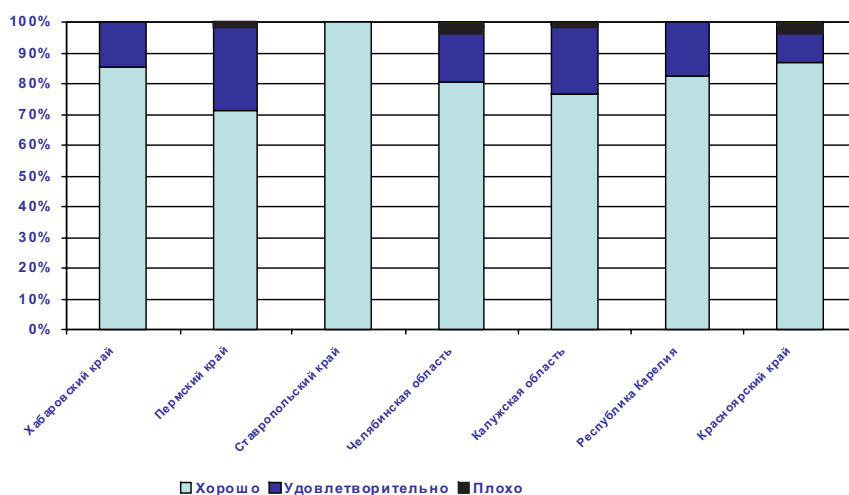
Тестирование было проведено в семи пилотных регионах проекта ИСО, протестировано было 4324 девятиклассника.

Регион	Количество центров	Количество учащихся
Калужская область	19	389
Красноярский край	19	841
Пермский край	42	793
Республика Карелия	5	294
Ставропольский край	6	523
Хабаровский край	16	336
Челябинская область	18	1148
ИТОГО:	125	4324

Тестирование проходило в помещениях ММЦ, в которых проводится и ЕГЭ. На выходе, после тестирования, члены рабочей группы проводили опрос учащихся. Школьникам задавали следующие вопросы:

- понравилось ли им участвовать в таком тестировании;
- устали они или нет;
- сложные были задания или нет;
- хотели бы они, чтобы их протестировали еще раз?

В целом ребятам участвовать в таком необычном тестировании понравилось. На диаграмме 1 приведена усредненная оценка, данная школьниками самим себе по их субъективным ощущениям, т.е. это усредненный ответ на вопрос “Как вы выполнили данный тест?”.



Однако реальные результаты по оценке ИКТ-компетентности не так радужны. Часто школьники говорили, что задания в тесте были простыми (легкими), но при обсуждении конкретных заданий через систему наводящих вопросов выяснялось, что ученик не понял сути вопроса. На первый план выступает та же причина, что и в неутешительных результатах программы PISA: наши школьники не умеют работать с информацией.

Отметим, что наиболее часто при опросе на выходе с тестирования школьники давали следующие комментарии: “Я никогда не читал так много”, “Очень много приходилось читать”, “Устал читать, часть заданий вообще не читал, пытался угадать ответ” и т.д.

На диаграмме ниже приведены средние результаты всех протестированных школьников с разделением по когнитивным умениям. Каждое умение оценивалось по трехбалльной шкале: low, medium, high (заметим, что в структуре ИКТ-компетентности проверяемые навыки расположены в порядке возрастания когнитивной сложности). Относительно высокие результаты, показанные учащимися по таким умениям, как управление и создание информации, отчасти объясняется тем, что на каждое из этих умений было предложено 1–2 задания, и эти задания оказались легче или понятнее остальных.

### Вопросы и задания

1. Приведите примеры идей, и в частности в образовании, которые изменились при вашей жизни.

2. Приведите примеры предметов материальной культуры человека, которые функционально изменились при вашей жизни.

3. Приведите примеры образцов поведения людей, которые изменились в типичных устойчиво-бытовых ситуациях.

4. Существует ли такая жизненная ситуация, в которой для человека принципиально важно (вопрос жизни и смерти) владение ИК-технологиями.

5. Какими технологическими навыками должен обладать школьник для успешного выполнения задания “Языки программирования”? Владеют ли ваши ученики (девятиклассники) этими навыками? Эти навыки при выполнении задания используются явно или косвенно?

### Литература

1. Чудинова В.П. Медиа и грамотность личности: процессы и проблемы. Российская государственная детская библиотека. М.: Школьная библиотека, 2004.

2. Сборник материалов “Научно-исследовательский институт информационных технологий социальной сферы Кузбасса, региональный центр реализации программы ЮНЕСКО “Информация для всех”. Сост.: Н.И. Гендина, Н.И. Колкова. М.: Школьная библиотека, 2005.

3. Ковалева Г.С., Красновский Э.А., Краснокутская Л.П., Краснянская К.А. Изучение знаний и умений учащихся в рамках Международной программы PISA. Общие подходы. ИОСО РАО, 1999.

4. Knowledge and skills for life. First results from PISA 2000. OECD, 2001.

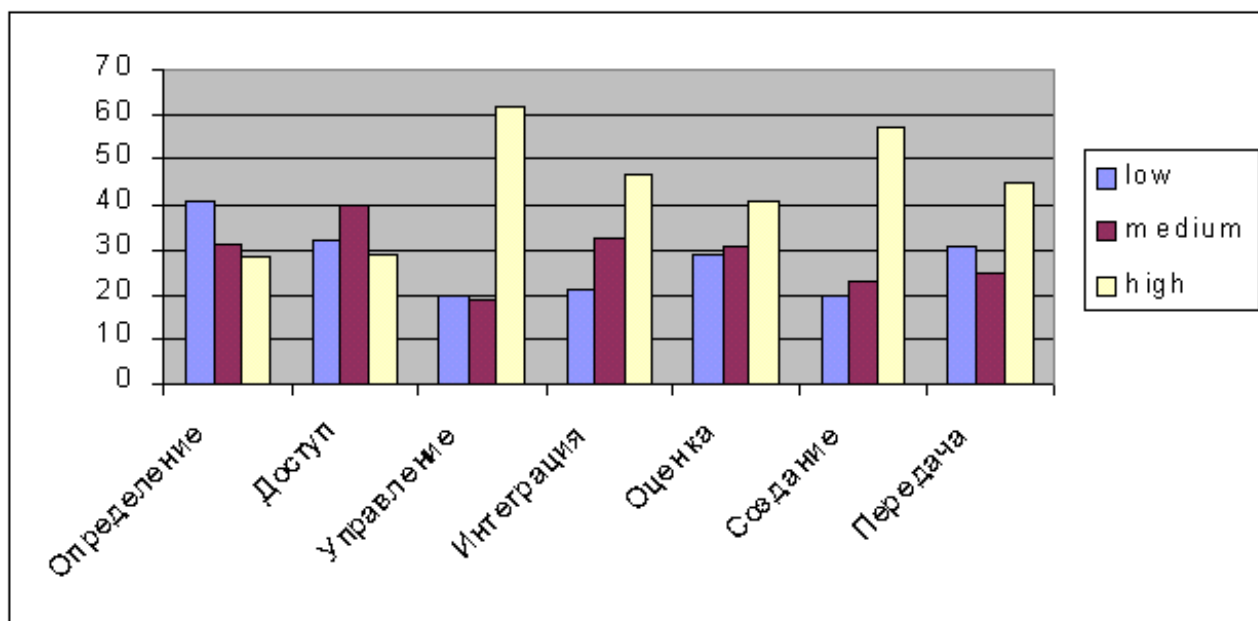
5. Knowledge and skills for life. First results from PISA 2000. Executive summary. OECD, 2001.

6. Measuring student knowledge and skills. A new Framework for Assessment. OECD, 1999.

7. [www.centeroko.ru](http://www.centeroko.ru) (материалы по PISA).

8. Бурмакина В.Ф., Фалина И.Н. Начало проекта по оценке ИКТ-компетентности девятиклассников. Информатика № 2/2006.

9. Фалина И.Н. Компетентностный подход в обучении и стандарт образования по информатике. Информатика № 7/2006.



# Азы информатики. Выходим в Интернет

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

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

Краткое содержание нового учебника:

1. Компьютерные сети.
2. Как работает Интернет.
3. Электронная почта.
4. Сервисы Web.
5. Web-дизайн и основы сайтостроения.

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

Бумажные учебники выпускает издательство “БХВ-Петербург” (алгоритм заказа для школ по специальным низким ценам приводится на роботландском сайте по адресу [www.botik.ru/~robot/ru/bhv.htm](http://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/azinfo.zip> (6,3 Мб).

Отзывы пользователей можно посмотреть на странице: [www.botik.ru/~robot/sale/reasons.htm](http://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).

25 марта 2006 года в Санкт-Петербурге прошел Фестиваль курса “Азы информатики” (см. “Информатику” № 9/2006). В нем приняли участие 45 команд из 9 районов города, всего более 170 человек. По нижеприведенной ссылке расположена заставка с описанием Фестиваля, условиями и решениями заданий: <http://robotland.narod.ru/pub/festival.zip> (3,7 Мб).

В Роботландском сетевом университете ([www.botik.ru/~robot/ru](http://www.botik.ru/~robot/ru)) открыт набор на 2006/2007 учебный год (см. с. 48). В этом учебном году будет работать и новый курс 16. “Азы информатики. Выходим в Интернет” (по представляемым здесь материалам).

## Часть I. Компьютерные сети

### Слово к читателю

На суд доброго читателя выносится первый фрагмент учебника с тайной надеждой на радушный прием.

Дело в том, что мне самому этот материал нравится: нет-нет, да и снова перелистываю его страницы для собственного удовольствия! ☺

Оказывается, компьютерные сети — страшно интересная вещь! Как с точки зрения алгоритмических основ, так и с точки зрения содержательных идей. Эти базовые идеи каким-то образом залетели в головы сетевых пионеров в далекие 70-е годы и успешно работают до сих пор без радикальных изменений!

### Введение

Содержание уроков по компьютерным сетям разделено по трем уровням. Каждый следующий

уровень уточняет (детализирует) материал предыдущего и расширяет его:

1. **Читальный зал.** Компьютерные сети в упрощенном изложении. Этот материал читайте ниже.

2. **Академия-1.** Общая информация по сетям (более подробные, по сравнению с “Читальным залом”, сведения о компьютерных сетях). Этот материал читайте в следующем номере.

3. **Академия-2.** Локальные сети Ethernet (детальное описание принципов и алгоритмов работы локальных сетей на примере сетей Ethernet). Этот материал будет опубликован в № 19.

В соответствии с поставленными задачами обучения можно ограничиться только материалом *Читального зала*, можно продвинуться далее к *Академии-1* или углубиться в конкретные подробности построения сетей Ethernet на базе материала *Академии-2*.



## Читальный зал

В этом разделе представлен рассказ о компьютерных сетях в упрощенном изложении.

### В доме Куков появляется паутина

Интернет сегодня стал таким же обычным средством связи людей, как телефон, но возможности Интернета гораздо шире. Опутывая своей сетью малые и большие предприятия, он постепенно приходит в каждый дом.

Понятно, что семье Куков, в которой живут “продвинутые” компьютерные пользователи, было начертано попасть в сетевую паутину еще тогда, в далеком 1969 году, когда в США появился первый росток нынешней Всемирной сети. Даже удивительно, что Интернет у Куков появился только сейчас!



Рис. 1.1. Паутина в доме у Куков

Послушаем, как братья (Петя и Вася) справляются с “наступившим” на них радостным событием.

— Вася! У меня для тебя хорошая новость: папа наконец подключил наш домашний компьютер к сети Интернет.

— Здорово! Теперь можно этот Интернет “потрогать” своими руками. Столько разговоров вокруг: Интернет, Интернет... Ах, какие у него возможности! Интернет может все! Последнее время я чувствовал себя как Маленький Мук без своих волшебных туфель. Когда мы начнем разбираться с этой штуковиной?

— Почему бы не приступить прямо сейчас! Вот наш компьютер, к нему уже подсоединен модем, с нашим провайдером все отношения улажены, можно начинать работу!

— Двумя руками — за! Только по порядку. Модем, провайдер — что это такое?

— Разберемся сначала с самим Интернетом.

— Интернет — это Всемирная компьютерная сеть. Так говорят, но я не слишком хорошо представляю себе, как она устроена.

Сети бывают рыболовные для ловли трески — это понятно. Паук плетет свою сеть для бедных цоко-

тук, это тоже понятно. А что такое компьютерная сеть? Что ловят этими сетями?

— Компьютерными сетями ловят информацию! Как ты обмениваешься картинками со своей подружкой Машей Пушковой?

— Очень просто! Записываю картинку на диск и несу Маше. Маша переписывает с диска на свой компьютер и готово!

— Можно соединить ваши компьютеры проводами, и тогда не нужно будет бегать с дисками.

— Красивое решение! Компьютеры умеют передавать информацию по проводам?

— Почему же нет? Ведь информация от твоей старой мышки, например, в компьютер так и передается: по проводу. Ты прямо сейчас прекрасно видишь этот электрический мышинный хвостик.

— Верно! Для подключения мышки в компьютере предусмотрено специальное гнездо, в него я вставляю разъем, которым оканчивается мышинный хвост. А как соединить проводами два компьютера? Для этого есть гнезда, как для мышки?

— Для соединения двух компьютеров можно обойтись теми же гнездами и разъемами, которые в нем уже есть (рис. 1.2).

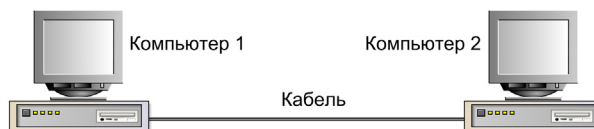


Рис. 1.2. Соединение двух компьютеров

Только кабель, используемый для соединения, не должен быть слишком длинным — ведь компьютерный сигнал очень слабый и быстро угасает на расстоянии.

— Маша живет в другом конце города, это расстояние не слишком большое для соединения наших компьютеров?

— Увы, таким образом вы с Машей не свяжетесь, да и как вы будете по улице прокладывать свой кабель? А вот компьютеры, находящиеся в соседних комнатах, соединить кабелем можно. И тогда они будут образовывать маленькую компьютерную сеть.

— А как же тогда компьютеры связываются друг с другом в разных странах?

— Для такой связи можно использовать телефонные линии и модемы.

Модем — это специальное электронное устройство: оно может вставляться прямо в корпус компьютера (*внутренний модем*) или располагаться подобно принтеру на столе рядом с компьютером и соединяться с ним проводами (*внешний модем*). Отдельным шнуром модем подключается к обычной телефонной розетке.

Когда модем работает на передачу, он усиливает и кодирует компьютерный сигнал для передачи по телефону, а когда на прием — декодирует телефонный сигнал и ослабляет его. Схема соединения показана на рис. 1.3:

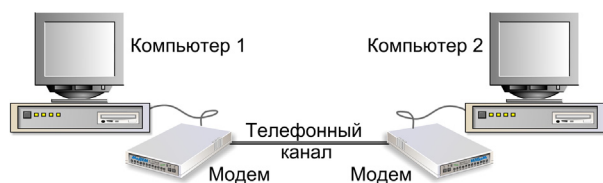


Рис. 1.3. Соединение двух компьютеров по телефону с помощью модемов

Замечу, что модем является не единственным *сетевым адаптером* — так называют электронные устройства, преобразующие компьютерную информацию в форму, пригодную для передачи по каналу связи и обратно.

Модемом называют сетевой адаптер, рассчитанный на телефонный канал связи, в отличие, например, от сетевого адаптера, рассчитанного на соединение компьютеров при помощи радио.

— Если два человека говорят по телефону, они могут говорить одновременно, и разговор не получается. Не возникает ли такая проблема при “общении” компьютеров?

— Ты прав! “Разговор” компьютеров по телефону не может быть таким же, как разговор людей. Обмен компьютерной информацией нужно управлять, и эту координацию выполняет *сетевое программное обеспечение* — специальные программы, которые должны быть установлены на связанных между собой компьютерах.

— Значит, мы с Машей можем установить на своих компьютерах модемы, сетевое программное обеспечение и обмениваться файлами по телефону?

— Да, конечно. Но лучше подсоединиться к серверу Интернета. Тогда вы сможете общаться не только между собой, но и со всеми другими пользователями этой Всемирной сети.

— А что такое сервер?

— *Сервер* — это компьютер, который при помощи специального программного обеспечения способен предоставлять пользователям сетевой сервис (например, обмен файлами, сообщениями). Слово “сервер” — английское, оно обозначает объект, который оказывает услуги (от слова *serve* — служить).

Например, сервер может обслуживать компьютеры школьной *локальной сети*, которые подсоединены к нему при помощи кабельной связи. На рис. 1.4 показан один из возможных вариантов соединения компьютеров в локальную сеть:

Сервер локальной сети



Рис. 1.4. Вариант соединения компьютеров в локальную сеть

Все узлы сети, показанной на рис. 1.4, связаны между собой при помощи одного общего кабеля. Такой способ соединения компьютеров в сеть, или, как говорят, *топология сети*, носит название *общая шина*.

Рядовые компьютеры, подключенные к локальной сети, называют *рабочими станциями*.

При помощи сервера в сети организуются разные полезные службы, доступные пользователям рабочих станций: обмен файлами, электронная переписка, служба новостей, доска объявлений, общение в режиме реального времени, видеоконференции, поддержка коллективных проектов...

Сервер хранит на своем жестком диске многочисленные файлы (наборы данных и программы), которые могут использовать рабочие станции. Кроме того, рабочие станции могут использовать оборудование, подключенное к серверу (принтер, сканер, факс, модем...).

Устройства могут подключаться к сети самостоятельно, без компьютера. Например, популярны сетевые принтеры: они имеют свой встроенный компьютер.

— Значит, сеть в школьном классе — это и есть Интернет?

— Интернет связывает компьютеры по всему миру. В вашей школе между собой связаны только 20 компьютеров. Какой же это Интернет! Сеть является частью Интернета, если она связана с другими сетями (или отдельными компьютерами), входящими в состав Интернета.

— Ты меня совершенно запутал! Получается, что сеть — часть Интернета, если она — часть Интернета. Масло масляное! Что же такое Интернет? Где он находится, кто у него хозяин?

— Ладно, ладно! Сейчас все станет понятно, но придется затронуть новейшую историю.

## История Интернета

В 60-х годах прошлого века Министерство обороны США задумало создать надежную систему связи между компьютерами своих штабов. Обмен данными не должен был прекращаться даже тогда, когда отдельные участки сети выходили из строя.

Исследования, проведенные в рамках этого проекта, были профинансированы подразделением Пентагона — Агентством перспективных научных разработок США (*Advanced Research Project Agency, ARPA*), и в 1969 году система была создана. Она получила соответствующее название — ARPANET (сеть ARPA).

Надежность работы сети обеспечивалась за счет:

- отсутствия у нее единого центра управления;
- множественных (паутиных) связей между отдельными узлами.

Если бы сеть управлялась из единого центра, то она прекратила бы свою работу при поломке головного компьютера или его каналов связи (рис. 1.5).

Центр управления

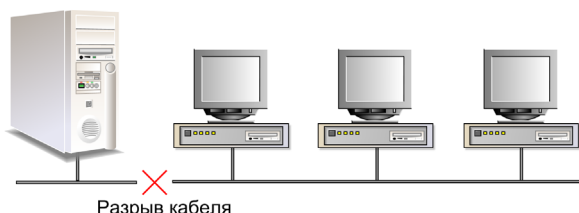


Рис. 1.5. Разрыв кабеля в центре управления вывел бы из строя всю сеть

Если бы компьютеры были соединены в цепочку, то сеть также была бы ненадежна. Скажем, выход из строя компьютера 3 привел бы к тому, что компьютеры 1 и 2 не смогли бы связаться с компьютерами 4 и 5 (рис. 1.6).



Рис. 1.6. Поломка компьютера в цепочке вывела бы из строя всю сеть

В надежной сети один компьютер может связаться с другим многими способами, и тогда выход одного узла из строя не повлияет на работу других.

Например, пять компьютеров можно соединить так, как показано на рис. 1.7.

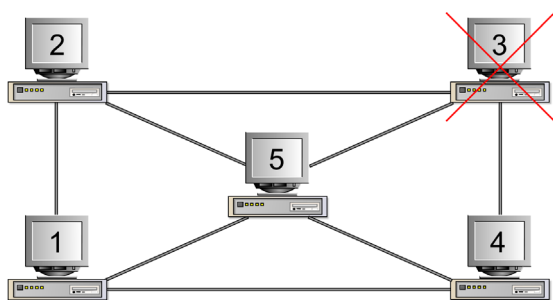


Рис. 1.7. Паутинная связь компьютеров в надежной сети

Если узел 3 выйдет из строя, это не повлияет на работу остальной части сети. Узел 4 может соединиться с узлом 2 либо через узел 1, либо через узел 5.

Говорят, что такие сети имеют *ячеистую топологию*. В ячеистой сети сообщение от одного узла к любому другому может проходить по нескольким маршрутам.

Идея компьютерных коммуникаций оказалась очень привлекательной. Сеть ARPANET, которая объединяла сначала 2 компьютера (один из которых был в Лос-Анджелесе, а другой — в Пало-Альто), потом 4 компьютера, стала зародышем Всемирной компьютерной сети под названием “Интернет”.

Со временем к этой сети подсоединялось все больше пользователей (отдельные компьютеры и целые сети). Остановить этот процесс было уже невозможно. Даже если бы ARPANET уничтожила все свои компьютеры и все свои каналы связи, то оставшаяся часть сетевого сообщества продолжала бы работать, даже не заметив потери.

Замечу к слову, что примерно так и произошло: в 1989 году исходная сеть ARPANET прекратила свое существование, но Интернет остался!

В 2005 году количество пользователей Интернета достигло миллиарда.

Согласно оценкам компании Morgan Stanley, 36% пользователей Интернетом сейчас проживают в Азии и 24% — в Европе. Лишь 23% пользователей живут в Северной Америке, т.е. там, где в 1969 году все это началось с сети ARPANET.

Для привлечения первого миллиарда пользователей Интернет трудился 36 лет. По прогнозам второй миллиард добавится уже к 2015 году (т.е. за 10 лет), затем темпы роста числа пользователей замедлятся, и третий миллиард будет добавлен только к 2040 году (за 25 лет).

У Интернета нет центра управления, а значит, нет и хозяина. Каждый узел Интернета сам заботится о том, чтобы иметь связь с другими узлами сети.

### На что уходят наши денежки

Вася немного помолчал, “переваривая” рассказ Пети, а потом задал коварный вопрос:

— Ты говоришь, что у Интернета нет хозяина. Значит, мы будем пользоваться им бесплатно? Как воздухом и солнечным светом?

— Мы будем платить деньги своему провайдеру. *Провайдер* — это организация, которая подключает пользователей к своему серверу, связанному с Интернетом.

— За что же мы будем платить деньги?

— Сервер провайдера — это компьютер, который должен работать 24 часа в сутки. Он обслуживает компьютеры своей собственной сети (своих *клиентов*), он передает и принимает внешние сообщения Интернета, в том числе и транзитные, то есть те, которые следуют через сервер нашего провайдера к своему адресату (рис. 1.8).

Компьютер-сервер надо обслуживать, поддерживая его в рабочем состоянии, за каналы связи с другими серверами надо платить владельцам этих каналов. Нужно платить за помещение, в котором находится сервер, за электроэнергию, зарплату обслуживающему персоналу... Вот на что тратятся наши денежки! Мы платим их не за Интернет, а за обслуживание.



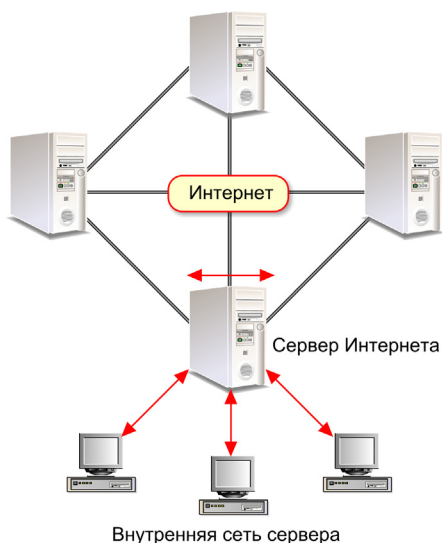


Рис. 1.8. Подключение к Интернету через сервер провайдера

### Локальные и глобальные сети

— Значит, компьютерная сеть — это либо локальная сеть, либо сеть Интернет?

— Классификацию компьютерных сетей можно проводить по разным признакам (материал раздела *Академия-1*). Если в качестве критерия деления рассматривать территориальный признак, то сети разделяют на локальные и глобальные.

*Локальная сеть* — соединение компьютеров, как правило, внутри здания или в пределах небольшой территории в одну группу для совместного использования информации, устройств и услуг.

*Глобальная сеть* — соединение компьютеров на большом географическом пространстве.

Интернет — глобальная сеть, но не единственная. Существуют и другие глобальные (в том числе и мировые) компьютерные сети, например, сеть Фидонет (Fido — так в США называют дворняг). Было бы неплохо, если бы ты самостоятельно нашел и прочитал описание этой сети.

### Как работает Интернет

Вася — не глупый мальчик. Если предмет ему интересен, схватывает на лету!

— Вот что в итоге я понял, — сказал Вася. — Сети бывают локальные и глобальные. Интернет — это глобальная всемирная компьютерная сеть. Она появилась в США в 1969 году под названием ARPANET. Сначала сеть содержала несколько компьютеров, затем к ней стали подсоединяться все новые и новые пользователи (отдельные компьютеры и целые сети).

На компьютере, подсоединенном к Интернету (*сервере Интернета*), должны быть установлены

специальные программы, которые обеспечивают прием и передачу данных по единым для всего Интернета правилам.

— Правила, по которым компьютеры в сети обмениваются информацией, называются *сетевым протоколом*, — заметил Петя.

— Интернет не имеет хозяина, не имеет единого центра управления, — продолжил Вася.

Каждый сервер Интернета организует передачу и прием данных для своих пользователей. Кроме того, серверы — хорошие товарищи. Они работают не только на своих клиентов, но принимают и передают транзитные сообщения.

— Ты все понял правильно, — подбодрил Петя брата. — Вот посмотри, на *рис. 1.9* я условно изобразил путь, который проделало сообщение, посланное с нашего компьютера на компьютер Грегга:



Рис. 1.9. Прохождение сообщения в Интернете

— На *рис. 1.9* показан не один путь, — заметил Вася, — а целых два.

— Это не ошибка, тропинок для одного сообщения может быть и более двух! Дело в том, что сообщение перед передачей в Интернет разделяется на части-пакеты. Каждый пакет путешествует самостоятельно, и путь его зависит от текущей «дорожной» ситуации, которая в Интернете меняется постоянно: перегрузка или временное отключение одних каналов заставляет пакет двигаться по другим. В пункте назначения отдельные пакеты собираются в исходное сообщение.

— Теперь твой рисунок понятен!

— К серверу Интернета подсоединяются не только отдельные компьютеры (как, например, наш домашний), но и локальные сети (как, например, ваша школьная сеть), — дополнил Петя рассказ Васи (*рис. 1.10*).





Рис. 1.10. К Интернету подсоединяются локальные сети и отдельные компьютеры

## Вопросы и ответы Читального зала

### Объясните смысл терминов

#### 1. Компьютерная сеть

**Ответ.** Компьютерная сеть — соединение компьютеров при помощи каналов связи для обмена информацией и совместного использования ресурсов и услуг.

#### 2. Локальная сеть

**Ответ.** Локальная сеть — соединение компьютеров внутри здания или в пределах небольшой территории для совместного использования информации, устройств и услуг.

#### 3. Глобальная сеть

**Ответ.** Глобальная сеть — соединение компьютеров на большой географической территории.

#### 4. Топология сети

**Ответ.** Топология сети — способ физического соединения компьютеров в сеть каналами связи.

#### 5. Интернет

**Ответ.** Интернет — глобальная всемирная компьютерная сеть. Интернет появился в США в 1969 году (ARPANET). Сначала сеть содержала несколько компьютеров, затем к ней стали подсоединяться новые пользователи по всему миру (отдельные компьютеры и сети).

#### 6. Сервер

**Ответ.** Сервер — компьютер, подключенный к сети (и выполняющаяся на нем программа), обслуживающий запросы клиентов на получение информационных услуг.

Сервер предлагает клиентам место на своем жестком диске, принтеры, сканеры, факсы, другие подключенные к нему устройства, организует различные сетевые службы (почта, файловые архивы, тематические страницы, доски объявлений, новостные группы, форумы, чаты, конференции...).

#### 7. Сервер Интернета

**Ответ.** Сервер Интернета — сервер, связанный с другими серверами Интернета. Сервер Интернета обслуживает клиентов и выполняет транзитную передачу сообщений от других серверов к адресатам.

#### 8. Клиент

**Ответ.** Клиент — компьютер (и программа), имеющий доступ к услугам сервера.

#### 9. Рабочая станция

**Ответ.** Рабочая станция — рядовой компьютер локальной сети.

#### 10. Провайдер

**Ответ.** Провайдер сетевых услуг — организация, предоставляющая сетевые услуги через свой сервер.

#### 11. Сетевой протокол

**Ответ.** Правила, по которым компьютеры в сети обмениваются информацией, называются сетевым протоколом.

Более подробный ответ: сетевой протокол — согласованный и утвержденный стандарт, содержащий описание форматов данных, правил приема и передачи. Протоколы служат для синхронизации работы сети.

#### 12. Сетевой адаптер

**Ответ.** Сетевой адаптер — электронное устройство (как правило, компьютерная плата) — преобразователь информации, поступающей от компьютера в форму, пригодную для передачи по каналу связи и обратно.

#### 13. Модем

**Ответ.** Модем — сетевой адаптер для передачи данных по телефонному каналу связи.

### Ответьте на вопросы

#### 1. Какие задачи решает компьютерная сеть?

**Ответ.** Компьютерная сеть позволяет объединять программные, информационные и электронные ресурсы компьютеров и предоставляет пользователям различные информационные услуги (электронная почта, работа с файловыми архивами, поиск информации, дистанционные конференции, электронная коммерция...).

#### 2. Чем глобальная сеть отличается от локальной сети?

**Ответ.** По территориальному признаку. Локальная сеть соединяет компьютеры внутри здания или в пределах небольшой территории. Глобальная сеть — на большом географическом пространстве.

### 3. Как появился Интернет?

**Ответ.** В 1969 году Агентством перспективных научных разработок (*Advanced Research Project Agency*, ARPA — подразделение Министерства обороны США) была создана компьютерная сеть ARPANET, из которой вырос современный Интернет.

Основной задачей этой разработки было получение надежной компьютерной сети, которая продолжала бы работу даже тогда, когда отдельные узлы выходили из строя.

### 4. Какие задачи решает сервер Интернета?

**Ответ.** Сервер Интернета работает на своих клиентов: передает и принимает для них информацию, а также выполняет транзитные пересылки информации, следующие через него к своему пункту назначения.

Основной режим работы сервера — ожидание запросов от клиентов или других серверов. Когда запрос получен, сервер либо сам выполняет его, либо передает запрос другим серверам.

### 5. Как информация передается в Интернете?

**Ответ.** Отправитель (сетевое программное обеспечение) разделяет сообщение на части-пакеты. Каждый пакет снабжается адресом получателя и отправляется соседу в сети по пути следования. В зависимости от текущей обстановки в сети пакеты могут попадать к адресату разными путями. В пункте назначения из полученных пакетов сообщение восстанавливается.

6. Назовите две основных причины, которые обеспечивают надежность Интернета.

**Ответ.** Надежность Интернета обеспечивается за счет:

- отсутствия единого центра управления;
- множественных (паутинных) связей между отдельными узлами.

## Задания

1. Исследуйте условия подключения к Интернету у провайдеров вашего населенного пункта.

На что следует обратить внимание:

- ✓ Тип подключения.
- Через аналоговый модем по телефонной линии (DIAL-UP). Самый медленный тип подключения (до 56 Кбит/с).
- Через цифровой модем по телефонной линии (ADSL). До 8 Мбит/с в режиме приема и до 1,5 Мбит/с в режиме передачи.
- Подключение на передачу в режиме DIAL-UP (до 56 Кбит/с), на прием через спутниковую антенну (до 5 Мбит/с).
- По кабелю (до 10 000 Мбит/с).
- Другим способом.

- ✓ Стоимость доступа.

Большинство провайдеров предлагают, как минимум, два тарифных плана: дневной (более дорогой) и ночной (более дешевый).

Оплата взимается либо за трафик (количество переданных и полученных мегабайт), либо за время работы в сети. Кроме трафика (или времени), может назначаться фиксированная абонентская плата.

Трафик может подразделяться на внутренний (в пределах локальной сети провайдера), российский и зарубежный. Внутренний трафик может быть бесплатным.

- ✓ Пропускная способность канала, соединяющего провайдера с Интернетом.

Измеряется в килобитах в секунду и определяет, сколько клиентов провайдера одновременно могут получить доступ в Сеть. Например, при пропускной способности 560 Кбит/с канал способен “пропустить” в сеть одновременно 10 человек, работающих по модему на скорости 56 Кбит/с.

- ✓ Удобство дозвона по модему.

Зависит от числа входных телефонных линий у провайдера. Скажем, для одновременной работы 10 человек необходимо 10 линий. Важно именно количество входных линий, а не число номеров (их может быть меньше).

Результаты исследования представьте в виде таблицы.

2. Исследуйте локальную сеть своего класса (школы). Подготовьте отчет, в котором укажите:

- ✓ Стандарт, по которому построена сеть (Ethernet, Fast Ethernet, Gigabit Ethernet, 10G Ethernet, Token Ring, FDDI).
- ✓ Быстродействие сети.
- ✓ Оценку реальной скорости передачи данных в вашей сети при малой, средней и максимальной нагрузке (подумайте, как организовать такие уровни нагрузок и как выполнить замеры скоростей).
- ✓ Топологию сети.
- ✓ Типы используемых каналов связи (коаксиальный кабель, витая пара, оптоволокно).
- ✓ Укажите используемые коммуникационные устройства (повторители, хабы, коммутаторы, маршрутизаторы) и их паспортные характеристики (производитель, количество портов, дополнительные свойства). Назначение и принципы работы этих устройств описаны в разделах *Академия-1* и *Академия-2*.
- ✓ Сколько разделяемых сред (доменов коллизий) в сети (см. *Академию-1* и *Академию-2*).
- ✓ Опишите функции сервера вашей локальной сети (предоставляемые пользователям устройства и услуги, в том числе связь с Интернетом).
- ✓ Нарисуйте полную схему сети.
- ✓ Кто обслуживает локальную сеть?
- ✓ Какое программное обеспечение используется для обеспечения работоспособности сети?

✓ Оцените стоимость сети (укажите цены коммуникационных устройств, кабелей, разъемов, адаптеров, стоимость монтажа и обслуживания сети).

3. Используя Интернет и другие источники информации, подготовить доклад на одну из перечисленных тем:

- Принципы построения небольшой локальной сети внутри квартиры или дома.
- Развитие Интернета в численном отношении, начиная с 1969 года.
- Развитие Интернета в России.
- Возникновение глобальной сети Фидонет и принципы ее работы.

### Вопросы зачета (с ответами)

В каждом задании Зачетного класса отметьте *все* правильные высказывания.

1. Компьютерная сеть:
  - 1.1. множество компьютеров внутри одного здания
  - 1.2. множество компьютеров на большом географическом пространстве
  - 1.3. множество компьютеров по всему миру
  - 1.4. множество компьютеров, соединенных каналами связи
  - 1.5. множество компьютеров внутри одного предприятия

*Правильные ответы:* 4.
2. Локальная сеть:
  - 2.1. компьютерная сеть на небольшой территории
  - 2.2. компьютерная сеть на большом географическом пространстве
  - 2.3. множество компьютеров на небольшой территории
  - 2.4. множество компьютеров на большом географическом пространстве

*Правильные ответы:* 1.
3. Глобальная сеть:
  - 3.1. множество компьютеров на небольшой территории
  - 3.2. множество компьютеров на большом географическом пространстве
  - 3.3. компьютерная сеть на небольшой территории
  - 3.4. компьютерная сеть на большом географическом пространстве

*Правильные ответы:* 4.
4. Топология сети:
  - 4.1. способ соединения компьютеров в сеть
  - 4.2. количество каналов связи компьютерной сети
  - 4.3. тип каналов связи компьютерной сети
  - 4.4. скоростные характеристики компьютерной сети

*Правильные ответы:* 1.
5. Интернет:
  - 5.1. сеть ARPANET
  - 5.2. мировая сеть с выделенным управляющим сервером
  - 5.3. локальная сеть
  - 5.4. глобальная сеть

5.5. международная сеть

5.6. компьютерная сеть

*Правильные ответы:* 4, 5, 6.

6. Сервер:

6.1. компьютер в сети

6.2. рядовой компьютер в сети

6.3. организация, поставщик сетевых услуг

6.4. компьютер, обслуживающий запросы клиентов

*Правильные ответы:* 1, 4.

7. Клиент:

7.1. компьютер, обслуживающий запросы клиентов

7.2. компьютер в сети

7.3. рядовой компьютер в сети

7.4. сетевой адаптер

*Правильные ответы:* 2, 3.

8. Сервер Интернета:

8.1. сервер, управляющий работой Интернета

8.2. компьютер, работающий по протоколу Интернета

8.3. компьютер, управляющий работой сервисной станции

8.4. сервер, связанный с другими серверами Интернета

*Правильные ответы:* 2, 4.

9. Рабочая станция:

9.1. рядовой компьютер локальной сети

9.2. пункт обслуживания компьютеров

9.3. компьютер, управляющий работой локальной сети

*Правильные ответы:* 1.

10. Сетевой протокол:

10.1. правила передачи данных в сети

10.2. правила работы пользователей сети

10.3. отчет о сетевой активности компьютеров

10.4. отчет о сетевой активности компьютерной сети

*Правильные ответы:* 1.

11. Две основные причины надежности Интернета:

11.1. разветвленные связи

11.2. дублирование информации

11.3. защита от вирусов

11.4. равноправие всех узлов

11.5. наличие единого управляющего центра

*Правильные ответы:* 1, 4.

12. Сетевой адаптер:

12.1. преобразователь сигналов

12.2. ускоритель сигналов

12.3. руководство по работе в сети

12.4. тренажер по работе в сети

*Правильные ответы:* 1.

13. Модем:

13.1. устройство для подключения компьютера к телефонной линии

13.2. телефонный аппарат, подключенный к компьютеру

13.3. сетевой адаптер

13.4. преобразователь сигналов

*Правильные ответы:* 1, 3, 4.

# Как компьютер читает файл

Е.А. ЕРЕМИН,

г. Пермь

Представление пользователя о файловой системе как об иерархически организованном множестве информационных объектов имеет мало общего с порядком хранения файлов на диске. Файл, имеющий образ цельного, непрерывающегося набора байт, на самом деле очень часто разбросан “кусочками” по всему диску, причем это разбиение никак не связано с логической структурой файла...

В.Г. Олифер, Н.А. Олифер [1]

Программа, скрывающая истину об аппаратном обеспечении и представляющая простой список поименованных файлов, которые можно читать и записывать, и является операционной системой. ...Абстракция, предлагаемая операционной системой, намного проще и удобнее в обращении, чем то, что может предложить нам непосредственно основное оборудование.

Э.Таненбаум [2]

Практически каждый пользователь умеет открывать файл, используя стандартный диалог операционной системы Windows. Именно с этого действия чаще всего начинается работа с компьютером: загрузить текст в MS Word, рисунок в Corel Draw и т.д. Наиболее “продвинутые” — те, кто знакомился с учебной литературой, дополнительно знают, что диск состоит из отдельных дорожек и секторов, в которых хранится информация файла. И лишь очень немногие имеют представление о том, как эти самые сектора и дорожки ставятся в соответствие с указанным нами именем файла.

Типичный ответ на вопрос, как ПК по имени файла считывает с диска его содержимое, до неприличия прост: это делает установленная на компьютере операционная система. Ответ этот по-своему замечателен: он короткий, понятен и абсолютно правилен. Единственный его недостаток состоит в том, что непосредственного понимания сути происходящего информационного процесса он почти не добавляет. Грех не вспомнить здесь замечательную цитату из книги наших известных фантастов братьев Стругацких “Пикник на обочине”:

“Как насчет того, что человек, в отличие от животных, существо, испытывающее непреодолимую потребность в знаниях? Я где-то об этом читал.

— Я тоже... Вся беда в том, что человек, во всяком случае массовый человек, с легкостью преодолевает эту свою потребность в знаниях. По-моему, у него такой потребности и вовсе нет. Есть потребность понять, а для этого знаний не надо. Гипотеза о Боге, например, дает ни с чем не сравнимую возможность абсолютно все понять, абсолютно ничего не узнавая... Дайте человеку крайне упрощенную систему мира и толкуйте всякое событие на базе этой упрощенной модели. Такой подход не требует никаких знаний. Несколько заученных формул плюс так называемая интуиция, так называемая практическая сметка и так называемый здравый смысл”.

Я ни в коей мере не пытаюсь осудить тех, кому приведенный выше ответ об операционной системе достаточен. Вполне допускаю, что таких пользователей даже большин-

ство, и, к моему личному сожалению, их едва ли заинтересует данная статья (в то же время, я затрудняюсь придумать случай, когда понимание того, что такое кластер и почему следует регулярно производить дефрагментацию диска, кому-то повредит). Но я старался, потратив на обдумывание, подготовку и написание данной статьи много сил и времени, ради тех, кто не удовлетворен приведенным выше кратким ответом. Ради тех, кто не утратил естественного природного человеческого любопытства, стремления дойти до сути и во всем разобраться. Это лучшая часть людей, и именно они открывают законы природы, создают принципиально новые технологии и неизвестные ранее устройства. Ради них стоит постараться.

Итак, данная статья дает развернутый ответ на сформулированный в заголовке вопрос о том, как именно компьютер читает файл, имя которого мы указали. Несмотря на подробность и логическую строгость статьи, я очень старался, чтобы она читалась легко и не требовала какой-либо глубокой предварительной подготовки. Я надеюсь, что уровень ее изложения вполне доступен рядовому старшекласснику. В то же время, данная статья не претендует на право быть энциклопедически полным собранием всех технических подробностей процесса. Напротив, материал, который в ней приводится, отобран по степени важности и типичности, а все второстепенные детали отброшены.

И последнее замечание, прежде чем мы начнем. В силу сложившихся традиций в нашей стране операционные системы от Microsoft являются доминирующими. Кроме того, в большинстве изданных ранее книг подробно рассматривается файловая система FAT, хотя она отнюдь не является единственной даже в среде Windows. Подобное единообразие хотя и удобно с точки зрения самообразования (берешь любую книжку и читаешь), но приводит к некоторой ограниченности мышления. В такой ситуации начинает казаться, что операционные системы могут быть только такими, что, разумеется, неверно не только теоретически, но и практически. Маленький пример такого сорта. Согласно идеологии



всех ОС Microsoft, каждый конкретный файл принадлежит *одному и только одному каталогу*. Казалось бы, а как еще может быть? А вот в операционных системах семейства Unix любой файл может *одновременно принадлежать произвольному числу каталогов*. Скажем, имеется набор административных документов, которые используются в течение нескольких лет. Тогда в Unix вы можете их поместить в каталог каждого года, причем реальный файл на диске будет только один. Справедливости ради следует сказать, что в Windows тоже есть средство для решения рассматриваемой проблемы, хотя и не такое естественное, — использовать ярлыки (специальные небольшие файлы, ссылающиеся на требуемый файл).

**Общая схема процесса**

Рассмотрим подробным образом, что происходит в компьютере от момента, когда мы выбрали в диалоге открытия файла (см. рис. 1) тот, который нам нужен, до момента, когда данные из этого файла попадут в необходимую область ОЗУ.

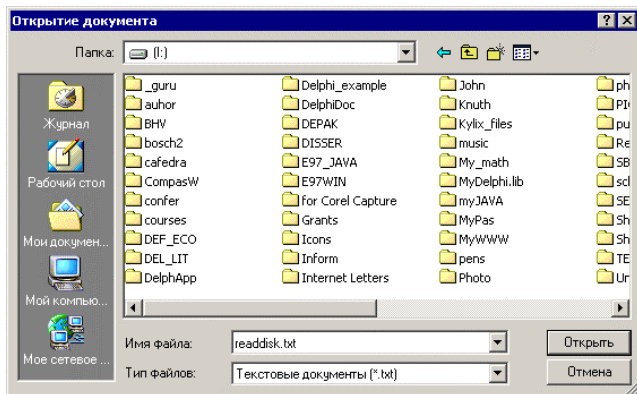


Рис. 1. Стандартный диалог открытия файла в Windows

Особо подчеркнем, что диалог открытия файла фактически стал стандартным компонентом в среде Windows, поэтому он выглядит практически одинаково для всех прикладных программ. Его назначение состоит в том, чтобы через удобный графический интерфейс помочь пользователю в сложной иерархии папок найти и выбрать необходимый файл, а затем вернуть программе правильный полный путь к этому файлу. Далее программа обычно вызывает стандартные функции Windows для работы с файлами, которые просто переадресуют все обращения операционной системе.

Параметрами, которые прикладная программа сообщает ОС для загрузки файла, являются его *полное имя* и *адрес памяти*, куда необходимо прочитать содержимое файла (или его части). Таким образом, роль самой прикладной программы в чтении невелика и состоит в

обращении к операционной системе с необходимыми параметрами<sup>1</sup>. Отсюда следует, что чтение файлов прикладными программами происходит единым стандартным образом, который определяется операционной системой. В дальнейшем мы будем в основном говорить для краткости о чтении файла *операционной системой*, хотя, точнее говоря, данное чтение обычно производится “по заказу” некоторой прикладной программы.

Хотя это и не обязательно, для единообразия рассуждения везде будем предполагать, что считывается текстовый файл, а его полное имя имеет общий вид

$$D:\ p \backslash f.txt$$

где *D* — имя диска (устройства), *p* — путь к файлу, а *f.txt* — его имя. Разумеется, вместо любого из однобуквенных компонентов можно подставить произвольные корректные значения, например, путь по папкам нескольких уровней.

Полная схема процесса приведена на рис. 2.

Обратимся к левому верхнему углу рисунка, где изображен вызванный некоторой прикладной программой диалог. Пусть пользователь с его помощью выбрал некоторый файл, полное имя которого мы договорились обозначать *D:\ p \ f.txt*. Приняв строку с этим именем, операционная система прежде всего выделяет из него имя устройства *D:* и устанавливает с ним связь. Физическая природа используемого устройства может быть весьма разнообразной: жесткий диск, CD-ROM, сетевой диск и т.п. В любом случае ОС анализирует правильность<sup>2</sup> ссылки на устройство и разрешение на доступ к нему для данного пользователя и организует канал обмена с ним. Сюда входит целый комплекс мероприятий технического характера, начиная от активизации устройства до выделения в ОЗУ буфера определенного размера для чтения данных — конкретными деталями ведает драйвер устройства. В результате ОС получает воз-



Рис. 2. Полная схема процесса считывания файла

<sup>1</sup> Данное обстоятельство очень удобно для программистов, которые пишут прикладное ПО.

<sup>2</sup> Если устройство взято из стандартного диалога, то проверка будет излишней, но в общем случае защита от ошибок может оказаться полезной.

можность работать с устройством и переходит к следующей задаче — загрузке необходимого каталога, в котором находится файл *f.txt*.

На данном этапе ОС расшифровывает следующий элемент имени файла — путь. Поскольку современные операционные системы позволяют создавать иерархические системы с большой степенью вложенности каталогов, данный шаг может оказаться достаточно сложным. Если не вдаваться в детали, то общая стратегия выглядит примерно так. С диска считывается корневой каталог, в котором делается попытка найти папку самого верхнего уровня. Например, для пути

*Тексты\Документы\1*

это будет папка *Тексты*. Далее производится считывание следующего каталога пути *Документы*, и т.д. Теоретически путь может оказаться неправильным, о чем операционная система уведомит программу и прекратит дальнейшие действия. В нашем случае, когда выбор пути осуществлялся из диалогового окна, ошибка исключена (т.е. любой выбранный путь реально существует). Последовательно расшифровав весь путь *p*, ОС считает последний каталог (в приведенном выше конкретном примере это будет каталог *1*), в котором найдет запись об интересующем нас файле.

**Примечание.** Современные ОС производят так называемое *кэширование* диска, которое заключается в том, что читаемые сектора диска и используемые каталоги запоминаются в ОЗУ; в случае, если они понадобятся повторно, обращение к диску уже не потребуется. В нашем примере наверняка так и будет: в процессе выбора в диалоге открытия файла система уже прочитала необходимый каталог, так что повторно его искать на диске уже не стоит.

Найденная в каталоге запись будет сохранена в определенной области ОЗУ и проанализирована с целью извлечения полной первичной информации о положении файла *f.txt* на диске. Слово “первичной” мы вынуждены употреблять потому, что для внутреннего удобства организации файловой системы ОС оперирует не секторами и дорожками, которые в конечном счете позволят найти на диске необходимую информацию, а так называемыми **кластерами** (в Unix используется термин **блоки**) — некоторыми минимальными порциями, которыми информация читается или пишется на диск. Главной причиной использования кластеров является необходимость сохранять информацию о каждом доступном в системе блоке диска, следовательно, таких блоков не должно быть чрезмерно много. Даже на первых жестких дисках требовалось объединять в кластер по несколько последовательно располагающихся секторов, а про современные объемы винчестеров и говорить не приходится<sup>3</sup>.

Способы хранения номеров кластеров, образующих файл, в различных операционных системах различны. Например, в первой ОС для микрокомпьютеров CP/M в записях каталога хранилась вся информация о распо-

ложении файла. А в файловой системе FAT, которая возникла в MS-DOS, но до сих пор поддерживается Windows, в каталоге хранится номер только самого первого кластера; цепочка остальных образующих файл кластеров извлекается из так называемой “FAT-таблицы”<sup>4</sup>. Таким образом, теоретически для получения полной информации обо всех образующих файл кластерах тех данных, которые прочитаны из каталога, может оказаться недостаточно, и потребуются дополнительное обращение к диску. Наличие кэширования, о котором говорилось в приведенном выше примечании, практически исключает такую необходимость.

Итак, в результате анализа каталога и, возможно, некоторых дополнительных системных таблиц ОС способна извлечь последовательность чисел, каждое из которых обозначает номер кластера, принадлежащего файлу. Пока этой информацией еще нельзя непосредственно воспользоваться для обращения к диску, и необходима ее дальнейшая обработка.

Первый этап такой обработки довольно прост: система переходит от номеров кластеров к номерам логических секторов. **Логические сектора** — это виртуальные блоки, емкость которых равна емкости реальных (**физических**) секторов, но пронумерованные последовательным образом и не имеющие непосредственной привязки к секторам и дорожкам диска. Проще говоря, считается, что все логические сектора образуют некоторую линейную структуру — этакую гигантскую единую дисковую дорожку (см. *Эксперимент 3* в следующем номере).

Пересчет из одних виртуальных номеров в другие, т.е. из кластеров в логические сектора, осуществляется довольно просто: в принципе для получения номера самого первого логического сектора в кластере достаточно просто умножить номер кластера на константу, равную числу секторов в кластере; номера следующих секторов получают последовательным прибавлением единицы. На практике формула пересчета может выглядеть чуть-чуть сложнее, поскольку приходится согласовывать значения начальных номеров: так, например, в системе FAT кластеры нумеруются, начиная с 2, а логические сектора — с 0.

Зато следующий пересчет получается заметно сложнее. Зная номер логического сектора, его необходимо перевести в три стандартные координаты: номера головки (считается, что у каждого диска есть 2 стороны и каждая из них обслуживается собственной магнитной головкой; благодаря объединению нескольких дисков на одной оси возможно большее число головок<sup>5</sup>), дорожки и сектора. В расчете используются некоторые константы, например, количество секторов на дорожке и общее количество головок накопителя. Более подроб-

<sup>4</sup> FAT расшифровывается как *File Allocation Table* — таблица расположения файлов.

<sup>5</sup> Интересно, что число головок у некоторых винчестеров было нечетным: одна поверхность использовалась для размещения *серво-меток*, обеспечивающих навигацию блока головок (*DSS — Dedicated Servo Surface*).

<sup>3</sup> На моем домашнем жестком диске емкостью 100 Гб размер кластера составляет 8 Кб, т.е. 16 секторов.

но примеры формул для пересчета будут приведены в подробном описании соответствующего этапа.

Наконец, вместо имени файла система получила физические координаты тех секторов диска, которые содержат информацию файла *f.txt*. Остается их прочитывать в ОЗУ.

Чтением секторов в системе координат головка—дорожка—сектор занимается ROM-BIOS — специальная системная программа, находящаяся в ПЗУ компьютера IBM PC. Она способна прочитать физические сектора (один или несколько, но в пределах физической дорожки) в служебную область ОЗУ, которая называется **буфером**. Остается перекопировать данные из буфера в то место ОЗУ, где программа их ожидает увидеть.

Копирование — это последняя операция в цепочке чтения файла. Процессор легко и быстро умеет копировать любые объемы данных, если известны начало исходного и результирующего массивов, а также их длина. Подчеркнем, что реальная длина файла в большинстве случаев не равна в точности объему целого количества секторов. В результате самый последний сектор чаще всего получается неполным: лишь его начальные байты относятся к данным файла, а конец может быть заполнен абсолютно произвольными кодами. Поскольку читать можно только весь сектор целиком, то в буфер излишние данные все же попадают, но копированию они не подлежат.

Блок копирования на *рис. 2* показан пунктиром по следующей причине. Операционные системы обычно позволяют управлять положением буфера считывания данных в ОЗУ. Поэтому прикладная программа “при желании” может совместить буфер ОС с той областью ОЗУ, куда требуется прочитать файл. Понятно, что в такой ситуации перепись уже не требуется.

Если файл имеет небольшой размер, то процесс чтения на этом заканчивается. Тем не менее при загрузке большого текста чтение придется продолжать (возвращаемся по пунктирной стрелке на *рис. 2*). После записи в память очередного блока файла происходит возврат на этап 3: из сохранившейся в ОЗУ информации каталога извлекаются номера очередных кластеров и все описанные действия снова повторяются до тех пор, пока не будет достигнут конец файла (EOF на рисунке как раз и означает *End Of File* — конец файла).

## Основные этапы процесса

Теперь, когда общая логика процесса стала нам ясна, рассмотрим каждый из его этапов более детально.

### Этап 1. Выбор устройства

**Вход:** полное имя файла вида *D:\p\f.txt*.

**Выход:** активированное устройство, на котором находится файл; остаток строки *p\f.txt*.

**Содержание:** выделить имя устройства из полной строки и проверить его правильность; проверить право доступа; активировать устройство и проверить его состояние; выделить в ОЗУ область под хранение информации о канале и буфер для приема данных.

**Ошибки:** неправильное имя устройства; доступ к нему запрещен; устройство не готово.

Первой частью длинной строки, определяющей местоположение файла, является указание устройства, на котором данный файл нужно искать. Учитывая широкий ассортимент устройств внешней памяти современного компьютера, пользователь имеет большой выбор среди допустимых источников данных.

Прежде всего упомянем традиционные дисковые накопители — один из разделов жесткого диска или все реже используемые на практике дискеты. Не менее распространены в настоящее время накопители на оптических дисках (CD-ROM) и на базе флэш-памяти. Гораздо реже используются дисковые устройства типа ZIP или JAZ, а также магнитооптические диски. Подчеркнем, что, помимо “встроенных” в корпус компьютера, все большее распространение получают внешние накопители, как правило, подключаемые к компьютеру через разъем USB (для примера на *рис. 3* показан внешний флоппи-дисковод от ноутбука). И хотя это физически такие же самые устройства, и на их дисках имеется такая же самая файловая система, но доступ к данным через USB-интерфейс осуществляется компьютером совсем иначе.



*Рис. 3.* Внешний накопитель на гибких магнитных дисках

Но и это еще не все возможные варианты. Современные операционные системы предоставляют необычайно удобную возможность создания так называемых *сетевых дисков*, работа с которыми для пользователя ничем не отличается от работы с перечисленными выше локальными устройствами; тем не менее данные таких дисков находятся на другом компьютере, возможно, расположенном достаточно далеко от вашего.

Современные цифровые портативные устройства (фотокамеры, сотовые телефоны) при подключении к компьютеру также отображаются в виде дисковых устройств (см. *рис. 4* на с. 24).

Наконец, рабочие данные, по крайней мере теоретически, могут временно записываться на RAM-диск, организуемый из выделенной для этой цели части ОЗУ. Подобный диск имеет определенные недостатки (не сохраняет информацию при выключении питания, уменьшает объем доступной памяти компьютера), зато работает необычайно быстро. Когда-то давно, когда винчестеры еще только появились и все дисковые накопители передавали данные раздражающе медленно, RAM-диски были очень полезны. Со временем их роль, разуме-



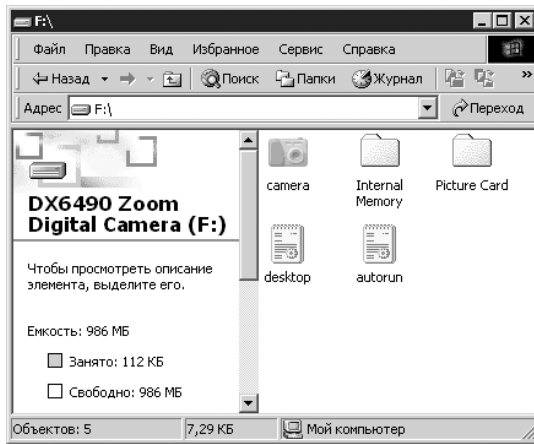


Рис. 4. Доступ к фотокамере как к диску

ется, существенно уменьшилась. Тем не менее резкое увеличение объемов устанавливаемого в компьютерах ОЗУ, наблюдающееся в последнее время, мягко намекает нам, что не следует хоронить данную возможность раньше времени.

Весь этот, согласитесь, впечатляющий перечень устройств действительно использует файловую систему. Тем не менее даже к устройствам, не имеющим таковой, можно обращаться как к файловым. Начиная с операционной системы Unix, где было введено понятие *специального файла*, существует возможность ввода данных с клавиатуры и вывода их на символьный дисплей или печатающее устройство теми же самыми методами, что и при работе с файлом.

“*Специальный файл*, называемый также виртуальным файлом, связан с некоторым устройством ввода-вывода и представляет его для остальной операционной системы и прикладных процессов в виде неструктурированного набора байт, то есть в виде обычного файла. Однако в отличие от обычного файла специальный файл не хранит статические данные, а является интерфейсом к одному из аппаратных драйверов ОС.

Использование специальных файлов во многих случаях существенно упрощает программирование операций с внешними устройствами. Со специальным файлом можно работать так же, как и с обычным, то есть открывать, считывать из него или же записывать в него определенное количество байт, а после завершения операции закрывать. Для этого используются привычные многим программистам системные вызовы для работы с обычными файлами: `open`, `create`, `read`, `write` и `close`” [1]

Те, кто умеет немного программировать, я думаю, поняли, о чем идет речь. Для всех остальных приведу пример из практики работы с системой MS-DOS или ее традиционной оболочкой Norton Commander<sup>6</sup>. Если при копировании в качестве имени результирующего файла указать *PRN*: (символическое имя принтера, со стандартным для всех имен устройств двоеточием), то файл

будет нормальным образом напечатан. Подчеркнем, что собственно принтер не имеет внутри себя файловой системы и, разумеется, не сохраняет распечатанные файлы в электронном виде. Иначе говоря, обсуждаемый пример показывает, что можно работать “файловыми” методами с абсолютно “нефайловым” устройством.

Таким образом, существует множество устройств, которые поддерживаются современными операционными системами. “Главный вопрос операционной системы — как сделать так, чтобы все устройства ввода-вывода и драйверы выглядели более или менее одинаково. Если диски, принтеры, клавиатуры и т.д. требуют различных интерфейсов, при появлении каждого нового устройства будет требоваться переделка операционной системы, что определенно не является удачной мыслью”. [2]

Каждое устройство в системе имеет свое имя, которое можно поставить в полное имя интересующего нас файла. Во многих (хотя и не во всех) ОС, начиная с CP/M, дисковые устройства принято обозначать буквой латинского алфавита, например, гибкий диск A: или локальный жесткий диск C: (чтобы данное имя система не спутала с однобуквенным именем файла, после него необходимо ставить двоеточие).

После того как система выделила имя устройства, указанное нами при выборе файла, и убедилась, что оно корректно, необходимо дополнительно проделать еще одну операцию — проверить наличие разрешения на доступ к данному устройству. Дело в том, что в многопользовательских системах различные пользователи (а значит, и запускаемые ими программы) имеют индивидуальные права доступа к тем или иным устройствам. Разграничение доступа является темой отдельного разговора, поэтому мы не станем углубляться в детали.

Итак, пусть система произвела все перечисленные выше проверки и не обнаружила причины прервать обмен (при задании имени файла с помощью диалога открытия в подавляющем большинстве случаев именно так и будет). Теперь необходимо проконтролировать состояние (статус) требуемого устройства и, если оно еще не было активировано ранее другими задачами, подготовить его к работе. Это весьма специфический этап, он существенным образом зависит от физической организации устройства. Например, для накопителя на гибких магнитных дисках необходимо включить мотор и дождаться его раскрутки, а также дополнительно проверять наличие дискеты в дисководе (случаются же рассеянные или вредные пользователи!). Диск на базе флэш-памяти таких операций, разумеется, не требует, но он подключается через интерфейс USB, который имеет свои особенности инициализации. А еще при неблагоприятном стечении обстоятельств могут возникнуть аварийные ситуации, о чем необходимо известить пользователя. По счастью, все эти технические тонкости учитывают драйверы низкого уровня, и при нашем уровне детальности рассказа их можно пропустить.

Подчеркнем, что, помимо подготовки к обмену аппаратной части файлового устройства, операционная система принимает еще целый ряд организационных мер по

<sup>6</sup> Достойный Windows-наследник NC — программный менеджер FAR — также поддерживает эту возможность.



его поддержке. Поскольку в современной ОС разрешен одновременный доступ нескольких заданий к дисковому устройству, система должна сохранять информацию о состоянии каждого канала обмена — следовательно, необходимо выделить и инициализировать некоторую служебную область ОЗУ, которая будет обслуживать чтение нашего файла с выбранного устройства. Кроме параметров текущего состояния канала обмена, в памяти должно быть зарезервировано место под буфер, в котором будут накапливаться данные по мере считывания. Упомянем также о кэшировании дисковых устройств, возможность которого система также должна предусмотреть при первом обращении к устройству.

Таким образом, после осуществления комплекса перечисленных выше, а также некоторых неназванных служебных действий операционная система готовит устройство к работе. Результатом этапа 1 является создание возможности прочитать каталог файлового устройства, что составляет содержание следующего этапа.

## Этап 2. Загрузка каталога (путь к файлу)

**Вход:** имя файла вида *p/f.txt*.

**Выход:** каталог с требуемым файлом; остаток строки *f.txt*.

**Содержание:** выделить путь из строки и проверить его правильность; проверить право доступа; последовательно пройти цепочку указанных каталогов и загрузить последний, в котором содержится запись о нужном файле.

**Ошибки:** неправильное имя в пути; доступ запрещен.

Когда-то очень давно, когда единственными магнитными дисками на ПК были гибкие (в самом прямом смысле слова) дискеты в бумажных конвертах диаметром 5 или даже 8 дюймов, вся служебная информация о файлах хранилась в едином каталоге. Тем не менее появление жестких дисков весьма скромной емкости в несколько мегабайт(!) существенным образом изменило ситуацию: на такой диск уже могли поместиться сотни файлов, так что найти в их длинном списке на экране нужный стало проблематично.

Выход из возникшего затруднения был найден довольно быстро, в результате чего файловая система приобрела иерархическую структуру. Препятствие осталось; он стал называться **корневым**, но в нем теперь могли храниться не только файлы, но и каталоги более низкого уровня; те, в свою очередь, помимо файлов, могли содержать новые подкаталоги (subdirectory) и т.д.

**Примечание.** В пользовательской литературе по ОС Windows принято использовать термин **папки**, который почти совпадает с классическим понятием каталогов. Ради общности рассуждений мы не будем конкретизировать систему и переходить на терминологию Windows.

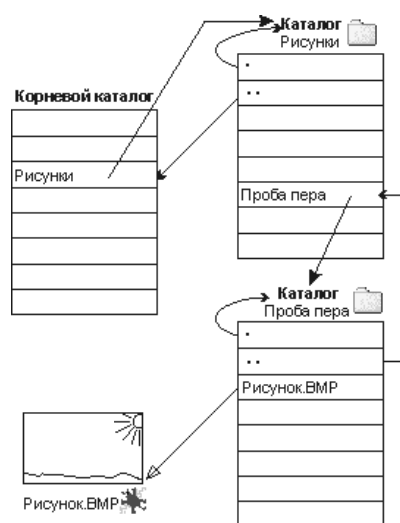


Рис. 6. Способ организации иерархии папок

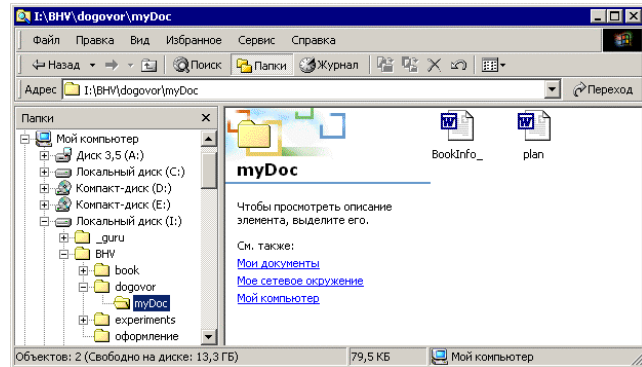


Рис. 5. Пример иерархии каталогов в окне Проводника Windows

В итоге при такой сложной системе “добраться” до нужного файла стало самостоятельной задачей. Перечень каталогов, которые необходимо пройти от корневого каталога до интересующего нас файла, получил довольно естественное название **путь** (path).

Иерархическая система вообще часто применяется человеком при хранении информации, так что, несмотря на некоторые технические трудности, пользователи ее быстро признали. Следует особо подчеркнуть, что указанная организация файловой системы применяется не только в распространенных у нас операционных системах, разработанных фирмой Microsoft, но и других современных системах, в том числе многочисленных Unix-подобных ОС. В качестве примера приведем цитату из книги [3]:

“В Linux есть даже оригинальная файловая система, позволяющая монтировать отдельные файлы, как если бы они были физическими устройствами.

Файловые системы присоединяются к файловому дереву с помощью команды `mount`. Эта команда связывает каталог существующего файлового дерева, называемый точкой монтирования, с корневым каталогом новой файловой системы. ...Например, команда

```
mount /dev/hda4 /users
```

монтирует на устройстве `/dev/hda4` файловую систему `/users`. По окончании монтирования можно с помощью команды `ls /users` просмотреть содержимое файловой системы”. [3]

Хотя иерархическая структура диска описывается в любой компьютерной книжке, все же рискнем рассмотреть ее еще раз, правда, несколько подробнее, чем в многочисленных пособиях для “чайников”.

Пусть на некотором диске создан каталог *Рисунки*, в который вложен еще один каталог — *Проба пера*, и в нем, в свою очередь, хранится наш рисунок под названием *Рисунок.BMP*. Как это выглядит на диске, иллюстрирует рис. 6.

В корневом каталоге среди прочих записей (чтобы не загромождать ри-

сунок, они не показаны) имеется служебная справочная информация о каталоге *Рисунки*. Она позволяет найти на диске некоторый специфический служебный файл, представляющий собой совокупность подробных записей о содержимом каталога *Рисунки*. Среди записей данного файла мы находим название интересующего нас вложенного каталога *Проба пера*, которое аналогичным образом указывает на очередной каталог. В последнем расположена ссылка на наш рисунок

*Рисунки\Проба пера\Рисунок.ВМР*

(как обычно требуют принятые Microsoft правила, элементы пути разделяются знаком обратной косой черты).

Отметим еще один важный момент, который облегчает ОС работу с многочисленными вложенными каталогами. Каждый из них (и в Windows, и в Unix) начинается двумя служебными записями, которые используются ОС для облегчения поддержания иерархической структуры и навигации в ней. Первая запись, обозначенная точкой, хранит ссылку на данный каталог, а следующая, название которой состоит из двух точек, — ссылку на “родительский” каталог. Такие записи имеются в любом файле с каталогом; в корневом каталоге они, разумеется, отсутствуют.

Приведенная на нашем рисунке организация относится к файловой системе FAT. Постепенно вытесняющая ее более сложная, но вместе с тем более надежная и способная обслуживать огромные диски система NTFS хранит каталоги как единое целое лишь до тех пор, пока они невелики. Когда объем информации в каталоге превысит некоторый предел, ОС выделяет под продолжение каталога дополнительное пространство в другом месте диска; если потребуется, процедуру выделения дополнительного места под расширение каталога можно повторять. Первоначальная (резидентная) часть каталога во всех случаях реорганизуется так, чтобы указывать уже не на файлы каталога, а на блоки его расширения, где, в свою очередь, теперь находится информация о расположении файлов. Важно отметить, что в отличие от системы FAT, где записи в каталоге хранятся в порядке возникновения, в NTFS каталог упорядочивается в соответствии с лексикографическим (алфавитным) порядком имен файлов. Благодаря данному приему ОС даже при нескольких дополнениях каталога может быстро определить, в каком из них должен находиться файл с нужным именем.

Иерархия файлов и каталогов может быть изображена в виде графа. Возможные варианты принципиально разных организаций представлены на рисунке, заимствованном из книги [1].

Здесь на *рис. 7а* показана структура файловых систем с единым каталогом (CP/M, RT-11, первые версии MS-DOS и др.), на *рис. 7б* — иерархическая структура каталогов более поздних версий MS-DOS и Windows, а на *рис. 7с* — иерархия ОС семейства Unix. Принципиальное отличие двух последних рисунков заключается в том, что в системах Microsoft всегда имеет место однозначная связь между каталогом и файлом, а в Unix это необязательно: некоторые файлы (на *рис. с* они закра-

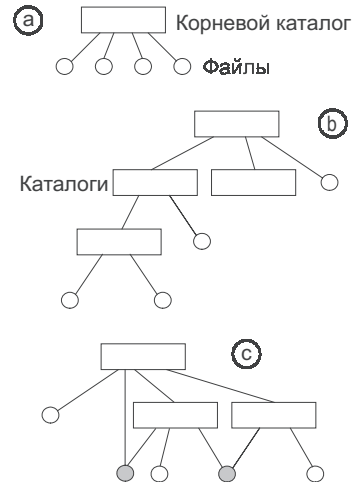


Рис. 7. Варианты иерархии файловой системы

шены серым цветом) могут одновременно принадлежать к нескольким каталогам и даже в них по-разному называться! Заметим, что граф *рис. 7б* в математике принято называть *деревом*, а на *рис. 7с* — *сетью*.

Интересно отметить, что способ организации каталогов может зависеть от конкретного вида устройств. В частности, рассмотренный ранее *рис. 6* относится к накопителям на гибких магнитных дисках, для которых характерно постоянное обновление состояния каталогов. На оптических дисках CD-ROM ситуация иная. Каталоги там обычно создаются единовременно (в ходе так называемой “сессии записи”), а длина всех файлов, которые будут записываться на диск, заранее известна. Поэтому для ускорения доступа к файлам на CD-дисках вся каталоговая информация сгруппирована в одном месте, а каждый файл размещен на диске непрерывно, так что однозначно определяется началом и длиной (отпадает необходимость отслеживать расположение фрагментов файла).

Итак, мы видели, каким образом система шаг за шагом проходит путь к затребованному пользователем файлу, находя среди многочисленных связанных каталогов тот единственный, в котором этот файл расположен.

### Этап 3. Определение положения файла

**Вход:** имя файла *f.txt*; каталог с требуемым файлом.

**Выход:** последовательность номеров кластеров, в которых находится файл.

**Содержание:** найти в каталоге информацию о файле; проверить право доступа; с помощью записи каталога о нужном файле расшифровать первичную информацию о его расположении на диске.

**Ошибки:** файл не найден; доступ запрещен; файл занят.

На предыдущем этапе был найден тот конкретный каталог, в котором содержится информация о требуемом файле. Теперь по имени надо найти относящуюся к нужному файлу запись каталога и ее содержимое расшифровать для определения, в каких частях диска располагаются фрагменты файла. Иначе говоря, на данном этапе обработки происходит преобразование имени тре-

буемого нам файла в информацию о **кластерах** — блоках, которыми оперирует система при чтении и записи информации.

Выбор размера блока в файловой системе является непростой задачей. С одной стороны, система обязана следить за состоянием каждого блока, следовательно, их желательно иметь не слишком много. Поскольку объем диска фиксирован, то это немедленно означает, что блоки не стоит делать слишком маленькими (физический сектор 512 байт для этой цели явно не годится). Но, с другой стороны, чем больше выбран размер блока, тем больше потери дискового пространства. В самом деле, ОС будет записывать информацию целыми блоками, а это значит, что какая-то часть последнего блока будет, как правило, оставаться незаполненной. Особенно плохо обстоит дело с маленькими файлами, такими, например, как стандартные файлы значков (иконки) по 766 байт: при размере блока 8 Кб такой файл использует менее 10%(!) объема блока, а остальное просто заполняется ничего не значащими кодами. Мы видим, что должен существовать некий оптимальный (не слишком большой и не слишком маленький) размер кластера; создатели ОС должны очень внимательно подходить к выбору данной константы. Читателям, которые заинтересуются данной проблемой, стоит посмотреть книги [2, 4].

Реально, как уже говорилось ранее, размер кластера выбирается от единиц до нескольких десятков килобайт.

Для решения задач этапа 3 система последовательно просматривает записи выбранного каталога, сравнивая в каждой из них соответствующее поле с тем именем, которое задано. При правильно указанном пользователем имени поиск будет успешным, в противном случае после просмотра всех записей и отсутствия совпадения на экране появится в той или иной форме знаменитое “file not found” (“файл не найден”).

Затем, после проверки прав доступа, ОС приступает к расшифровке найденной записи каталога. Механизм этой расшифровки существенным образом зависит от используемой файловой системы.

Наиболее просто рассматриваемая процедура выглядит в ОС CP/M. Там все кластеры, которые принадлежат файлу, заносились непосредственно в каталог, так что системе оставалось просто их извлечь из предназначенных для этого полей записи. Единственная трудность состояла в том, что одной записи для длинного файла могло не хватить; тогда система заводила в каталоге дополнительные записи.

Несколько сложнее устроен каталог в Unix. Помимо заданного числа полей (обозначим его  $K$ ) для номеров входящих в файл блоков (кластеров), предусмотрены три дополнительных (последних) поля для случаев, когда файл велик и места под номера его блоков непосредственно в записи каталога недостаточно. Расширение происходит следующим способом. Когда все возможные поля с номерами блоков в каталоге уже заполнены, в поле  $K+1$  указывается ссылка на специальный кластер, где хранится список номеров следующих кластеров файла. В отличие от CP/M,

где дополнительная информация заносилась в каталог, здесь используется пространство данных; очевидно, что размер кластера существенно больше объема дополнительной записи каталога! Но и это еще не все. Если файл существенно больше, чем это обеспечивает механизм простой косвенной адресации с одним дополнительным кластером, включается новый механизм — двойная косвенная адресация: в поле  $K+2$  заносится ссылка на кластер, в котором находятся ссылки на кластеры, где перечислены номера кластеров файла. Наконец, поле  $K+3$  позволяет реализовать тройную косвенную адресацию. В последнем случае данное поле ссылается на кластер, в котором хранится список кластеров второго уровня, каждый из которых, в свою очередь, содержит ссылки на кластеры с расширениями первого уровня, а те, наконец, — на кластеры со ссылками на кластеры данных. (Очень напоминает детскую сказку о доме, который построил Джек.) Полученный механизм косвенной адресации обеспечивает рост допустимого количества кластеров в файле по степенному закону, что дает возможность (по расчетам, приведенным в [1]) при 8-килобайтовом блоке адресовать файл размером более  $7 \cdot 10^{13}$  байт, что примерно в 700 раз больше, чем объем 100 Гб диска!

В файловой системе FAT каталог хранит только одну ссылку — на самый первый кластер файла. Для определения остальных ОС обращается к специальной таблице FAT (*File Allocation Table* — таблица размещения файлов; именно по ее аббревиатуре и называется данная файловая система). В этой таблице каждому кластеру соответствует свое число, описывающее состояние кластера. Если оно равно нулю, кластер свободен и может быть распределен для записи файла. Наличие ненулевого значения чаще всего говорит о том, что кластер входит в состав файла; само число является ссылкой на следующий кластер. Таким образом, имея из каталога первый кластер файла, мы обращаемся к FAT и извлекаем из нее число, соответствующее второму кластеру. Аналогичным образом снова обращаемся к таблице и получаем номер третьего кластера и т.д. Если в качестве номера следующего кластера таблица возвращает максимально допустимое число, то это значит, что кластер был последним и файл закончился<sup>7</sup>. Описанным способом ОС MS-DOS и Windows, если используется диск с FAT-системой, способны получить набор чисел, являющихся полным списком номеров кластеров файла.

В другой файловой системе, разработанной специально для Windows NT — NTFS (*NT File System*), запись каталога расшифровывается по-другому. Базовой единицей распределения дискового пространства в NTFS является не кластер, а непрерывная группа последовательных кластеров, которая называется **отрезком**. Отрезок однозначно характеризуется началом и длиной; в

<sup>7</sup> Аналогичным образом некоторая “большая” константа (т.е. заведомо большая, чем количество кластеров на диске) означает плохой блок диска, который система при записи будет игнорировать.



случае, если файл не сильно фрагментирован, его описание с помощью отрезков становится необычайно компактным (в предельном случае дефрагментированного файла потребуется всего один отрезок). Когда число отрезков невелико, информация о них помещается внутри одной записи. В этом случае получить список кластеров, образующих файл, чуть сложнее, чем в CP/M. Однако если файл большой, то вместо ссылок на отрезки файла в запись помещаются ссылки на новые записи. Допускается даже двойная косвенная адресация отрезков, когда запись ссылается на другие записи, в которых лежат ссылки на отрезки файла.

**Примечание.** Если файл очень мал (менее 1,5 Кб), то NTFS вообще размещает его содержимое непосредственно внутри информации о файле!

Сложность NTFS заключается не столько в сложности устройства ее файлов и каталогов, сколько в большом разнообразии вариантов. Еще одно усложнение, которое заслуживает упоминания, состоит в некотором стирании граней между каталогами и файлами: если в FAT-системе в кластерах хранились исключительно данные (корневой каталог хранился до кластерного пространства и в него не входил), то в NTFS весь диск состоит из кластеров. Но, так или иначе, по имени файла ОС Windows находит в каталоге запись о файле и однозначно предопределенным способом расшифровывает ее, преобразуя в последовательность номеров требуемых для чтения файла кластеров.

Приведенный выше краткий обзор механизмов обработки информации из каталога по замыслу автора должен убедить читателя в необычайном разнообразии этих механизмов (и еще в том, что Windows — это далеко не все, что было и есть в компьютерном мире!). Тем не менее все механизмы достаточно просты для того, чтобы их можно было запрограммировать, а значит, компьютер не встретит никаких трудностей при расшифровке требуемых для чтения файла номеров кластеров.

#### Этап 4. Пересчет номеров кластеров в номера логических секторов

**Вход:** номера кластеров, в которых находится файл.

**Выход:** последовательность номеров логических секторов, в которых находится файл.

**Содержание:** по несложной формуле пересчитать номера одних виртуальных блоков в другие.

Данный этап является одним из наиболее простых. Каждый из имеющихся номеров кластеров необходимо пересчитать в номера образующих его логических секторов. Напомним, что кластер — это совокупность определенного количества последовательно расположенных секторов.

Логические сектора — это тоже некоторые виртуальные блоки информации, о которых предполагается, что они располагаются линейно и нумеруются друг за другом, обычно начиная с нуля. Их размер совпадает с размером настоящих, физических секторов, но они лишены конкретного расположения на физических дорожках реального диска. Различие между логическими и физическими секторами поясняет рис. 8.

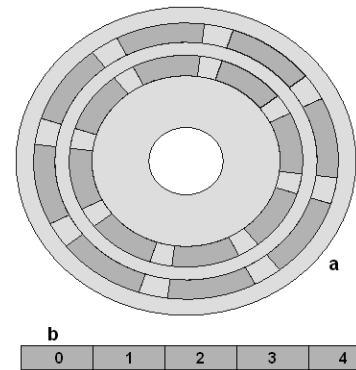


Рис. 8. Физические (а) и логические (б) сектора диска

Использование абстракции логических секторов вместо физических не есть простая прихоть создателей операционных систем. Дело в том, что такой подход позволяет программно описать *аппаратно независимую* файловую систему, никак не связанную с конкретной структурой дисков. И лишь на самых последних этапах (в нашем случае начиная с этапа 5), когда потребуется непосредственно прочитать сектора, осуществляется переход на реальный физический уровень. Описанная структура делает ОС легко переносимой на любое новое оборудование.

Сам пересчет номеров на логическом уровне производится очень просто. Поскольку каждый кластер содержит фиксированное число секторов (будем обозначать его  $M$ ), то для вычисления номера самого первого логического сектора в кластере достаточно умножить номер последнего на  $M$ . При этом, правда, требуется учесть, с каких значений начинается нумерация кластеров и секторов. Например, в файловой системе FAT логические сектора нумеруются с 0, а кластеры, по некоторым чисто техническим причинам, — с 2. Следовательно, номер самого первого логического сектора в кластере с номером  $C$  будет равен

$$L = M \cdot (C - 2)$$

Последующие сектора в кластере  $C$  будут иметь номера

$$L + 1, L + 2, \dots, L + (M - 1)$$

В частности, при  $M = 8$  в самый первый кластер с номером 2 войдут сектора 0–7, в следующий — 8–15 и т.д.

#### Этап 5. Пересчет номеров логических секторов в координаты физических

**Вход:** последовательность номеров логических секторов, в которых находится файл.

**Выход:** последовательность физических координат каждого сектора.

**Содержание:** пересчитать по формулам номера логических секторов в реальные координаты секторов на диске: головка, дорожка, сектор.

Данный этап по смыслу близок к предыдущему, только расчеты производятся по более сложным формулам.



Известно, что каждый диск обслуживается несколькими магнитными **головками** (если речь идет о дискете, то у нее 2 стороны, если на оси диска находится 2 диска, то головок уже 4 и т.д.). Каждая головка, перемещаясь *пошагово* вдоль своей поверхности, обслуживает концентрические “полосы” с информацией — **дорожки**. Наконец, каждая дорожка делится на **сектора** (см. рис. 8а).

В основе пересчета на этапе 5 лежат следующие допустимые любому нормальному человеку логические соображения: если имеется 40 секторов, а на дорожку диска входит по 18 секторов, то каким по счету будет последний из секторов на дорожке? Рассуждаем, например, так. Поскольку 36 секторов — это 2 полные дорожки, то оставшиеся 4 сектора пойдут на следующую дорожку и “наш” сектор будет четвертым. В математике для компактного решения подобных задач предусмотрены специальные операции: деление нацело (часто обозначается DIV) и остаток от деления (MOD). В частности, для нашего примера полных дорожек будет  $40 \text{ DIV } 18 = 2$ , а секторов на последней, неполной дорожке  $40 \text{ MOD } 18 = 4$ .

Помимо данной простой арифметики, необходимо дополнительно учесть следующие особенности физического устройства диска:

- исторически сектора принято нумеровать с 1, а дорожки и головки — с 0;
- количество головок и количество секторов на дорожке<sup>8</sup> для данного диска являются константами;
- для обеспечения большей скорости доступа сначала заполняется информацией вся дорожка (т.е. для каждой из головок) и только потом происходит переход к следующей дорожке.

Введем обозначения двух характерных констант:

ST — количество секторов на дорожке,

NH — количество головок.

Тогда формулы для вычисления номеров сектора S, дорожки T и головки H будут иметь вид (см., например, с. 237 в [5]):

$$S = 1 + L \text{ MOD } ST$$

$$T = L \text{ DIV } (ST * NH)$$

$$H = (L \text{ DIV } ST) \text{ MOD } NH$$

Рассмотрим числовой пример. Пусть  $ST = 18$ , а  $NH = 2$ , что соответствует дискете. Тогда для сектора  $L = 39$  (следует помнить, что “на самом деле” это сороковой сектор, поскольку счет у компьютера идет с нуля!) получим

$$S = 1 + (39 \text{ MOD } 18) = 4$$

$$T = 39 \text{ DIV } 36 = 1$$

$$H = (39 \text{ DIV } 18) \text{ MOD } 2 = 0$$

Полученные числа “на человеческом языке” можно объяснить так. Первые 36 секторов, имеющие номера от 0 до 35, заполняют обе стороны нулевой дорожки, а

оставшиеся 4 сектора с номерами 36–39 будут являться секторами 1–4 на дорожке с номером 1.

В заключение отметим, что описанная выше система координат — головка, дорожка, сектор — первоначально возникла на гибких магнитных дисках. Первые жесткие диски также имели параметры S, T, H, соответствующие реальному размещению секторов на поверхности диска (часто использовался термин *геометрия диска*). Однако позднее реальность геометрии постепенно утратилась.

Первое, что очевидным образом нерационально в строении диска, это постоянное число секторов на дорожке. Еще раз обратимся к рис. 8а, на котором отчетливо видно, что линейный размер секторов на внешних дорожках диска больше, чем на центральных. Можно предложить два выхода из этой ситуации: либо сохранить одинаковое число секторов, но записывать информацию с разной плотностью, либо на разных дорожках делать разное число секторов. На дискетах использовался первый подход<sup>9</sup>, поэтому для них номера секторов и дорожек вполне реальны. Зато на винчестерах ради увеличения емкости был выбран второй путь: диск разбили на зоны, в каждой из которых число секторов было свое.

Вторая причина ухода от реальной геометрии дисков состояла в резком возрастании объема винчестеров. При обращении к диску координаты S, T, H “запаковывались” в регистры микропроцессора, где им отводилось определенное число бит. С ростом объема дисков эти биты заполнялись неравномерно, и, например, оказывалось, что одну координату уже наращивать нельзя, а у другой, как ни обидно, наоборот, имеются неиспользуемые биты. Подобные проблемы возникали, когда емкость винчестеров превысила 528 Мб, а затем таких барьеров появлялось еще несколько (см., например, [6] или [7]). В результате всех усовершенствований координаты S, T, H стали формальными и не соответствуют более реальному устройству диска. Контроллер жесткого диска получает эти абстрактные числа и, зная конкретное устройство “своего” диска, пересчитывает формальные координаты в настоящие номера дорожек и секторов. Тем не менее по традиции за координатами секторов до сих пор сохраняются их первоначальные названия.

Таким образом, хотя на рассмотренном этапе 5 имеется множество различных деталей, для компьютера он не представляет абсолютно никакой сложности: произвести вычисления по трем формулам — задача для него даже слишком простая.

## Этап 6. Чтение секторов

**Вход:** физические координаты нужных секторов.

**Выход:** содержимое секторов в буфере.

<sup>9</sup> Кстати, именно по этой причине каталог дискеты находится на внешней дорожке: там меньше плотность записи и больше вероятность записать и считать информацию без ошибок.

<sup>8</sup> Последнее утверждение неочевидно и будет обсуждено позднее.

**Содержание:** по заданным координатам производится чтение информации из секторов диска в ОЗУ.

**Ошибки:** сектор не находится или не читается; не совпала контрольная сумма.

В IBM-совместимых компьютерах чтением секторов с диска обычно занимаются подпрограммы, располагающиеся в ПЗУ (ROM BIOS). Именно здесь производится учет всех конкретных особенностей дисковых накопителей. Не случайно поэтому при существенном изменении жестких дисков в связи с увеличением их емкости приходится перепрограммировать BIOS, т.е. обновлять содержимое микросхемы ПЗУ, изготавливаемой в настоящее время на основе флэш-технологии.

Взаимодействие подпрограмм BIOS с дисководом является темой для самостоятельной публикации. Данное ПО ведет диалог не только с интеллектуальным контроллером дисковода, но и с контроллером прямого доступа к памяти, обеспечивающим прием байтов из секторов без участия центрального процессора. Обсуждение столь сложных материй явно выходит за рамки данной статьи, поэтому здесь ограничимся только ссылкой на раздел 5.4.1 в известной книге [8], где очень удачно и без излишних технических подробностей обсуждается проблема низкоуровневого чтения данных с дискеты.

Разумеется, никто не запрещает программисту написать альтернативную программу обмена с диском, но это очень кропотливый и квалифицированный труд, который оправдан лишь при написании сложных дисковых утилит или антивирусного ПО. Вот что говорится, например, по этому поводу в книге [2]:

“Основными командами являются команды read и write (чтение и запись). Каждая из них требует 13 параметров, упакованных в 9 байт. Эти параметры определяют такие элементы, как адрес блока на диске, который нужно прочитать, количество секторов на дорожке, физический режим записи, расстановку промежутков между секторами. Они же сообщают, что делать с меткой адреса данных, которые были удалены. Если вы не можете сразу это осмыслить, не волнуйтесь — полностью это понятно лишь посвященным. Когда выполнение операции завершается, чип контроллера возвращает упакованные в 7 байт 23 параметра, отражающие наличие и типы ошибок. Но этого недостаточно, и программист при работе с гибким диском должен постоянно знать, включен двигатель или нет. Если двигатель выключен, то его следует включить (с длительным ожиданием запуска) прежде, чем данные будут прочитаны или записаны. Двигатель не может оставаться включенным слишком долго, так как гибкий диск изнашивается. Программист вынужден выбирать между длинными задержками во время загрузки и изнашивающимися гибкими дисками (с вероятностью потери данных на них).

Даже если не вдаваться глубже в подробности этого процесса, становится ясно, что обычный программист

вряд ли захочет столкнуться с такими деталями при работе с гибким диском (или жестким диском, работа с ним не менее сложна)”.

Отсюда понятно, что в подавляющем большинстве случаев доступ к чтению секторов осуществляется путем вызова стандартных подпрограмм ROM BIOS.

Особо подчеркнем, что BIOS, будучи ПО более низкого уровня, никак не зависит от загружаемой операционной системы. Напротив, именно BIOS иницирует чтение с диска загрузочного сектора, обеспечивая начало ее загрузки. И именно к подпрограммам BIOS в конечном счете (на завершающем этапе чтения) обращается ОС при работе с дисками.

Посмотрим для примера, как выглядит обращение к процедуре ROM BIOS для чтения одного сектора дискеты. Ниже приводится простейшая программа на Паскале (версия Turbo Pascal), которая содержит внутри себя вставку на ассемблере; она читает заданный сектор с диска A:. Переменная buf организует в ОЗУ буфер для приема 512 байт, содержащихся в секторе, и поступившие в нее в ходе считывания байты можно обрабатывать (или напечатать) обычным путем; для простоты эту часть программы мы не будем разрабатывать.

```
VAR buf: ARRAY [0..511] OF CHAR;
    sector, track, head: BYTE;
BEGIN Writeln('Введите десятичные номера:');
      Write('сектора: '); Readln(sector);
      Write('дорожки: '); Readln(track);
      Write('головки: '); Readln(head);
      ASM lea bx,buf {адрес буфера}
          mov cl,sector {сектор}
          mov ch,track {дорожка}
          mov dh,head {головка}
          mov dl,0 {диск A:}
          mov al,1 {количество секторов}
          mov ah,2 {функция «читать сектора»}
          int 13h {вызов функции BIOS}
      END
END.
```

**Примечание.** Функция BIOS INT 13, как и функция DOS INT 25 для чтения логических секторов, способна прочитать несколько последовательных секторов. Но функция BIOS по сравнению с функцией DOS менее универсальна и годится только для секторов *в пределах одной физической дорожки*; функция DOS лишена подобных ограничений.

Поскольку ввод данных из файла заключается в чтении большого количества секторов, на этапе 6 большую роль играет опережающее чтение секторов, осуществляемое ОС. Если файл не сильно дефрагментирован, то данный метод существенно ускоряет процесс.

Теперь у нас есть некоторое общее представление о том, каким образом происходит чтение секторов с дискеты и их содержимое попадает в ОЗУ. Остается рассмотреть последний этап, в ходе которого данные попадают в нужное прикладной программе место ОЗУ.

## Этап 7. Копирование прочитанного

**Вход:** содержимое секторов в буфере.

**Выход:** его копия в области ОЗУ, принадлежащей прикладной программе.

**Содержание:** содержимое заполненной информацией части буфера копируется в ту область ОЗУ, адрес которой задала прикладная программа.

Совершенно очевидно, что процесс чтения сектора, особенно когда используется механически движущийся диск, с позиций компьютера весьма медленный. Отсюда следует, что данные всего сектора не могут появиться в ОЗУ моментально, а, напротив, какое-то время накапливаются<sup>10</sup>. Для обеспечения этого процесса ОС выделяет специальную область памяти, которую принято называть **буфером**.

Буфер, как правило, выделяется в системной области. Когда чтение с диска завершается, прочитанная информация копируется в то место ОЗУ, где она требуется прикладной программе. Как уже упоминалось ранее, адрес ОЗУ, начиная с которого программа “хочет” иметь информацию, является еще одним (кроме имени файла) параметром, который ПО сообщает системе.

Наличие буфера дает возможность решить и еще одну проблему — чтение произвольного количества данных, некратного сектору. В самом деле, предположим, нас интересует некоторый файл размером 120 байтов, причем программа выделила под него в памяти ровно такой объем. В то же время, минимальной порцией информации, которую способен прочитать накопитель, является сектор, т.е. 512 байт. Отсюда совершенно очевидно, что прочитать данные в непосредственно выделенные 120 байт ОЗУ принципиально невозможно. При помощи буфера проблема решается элементарно: сектор, как обычно, читается в буфер целиком, а затем из прочитанных 512 байт копируется только 120.

Хочется также сказать несколько слов о положении буфера в ОС. Многие функции ОС, особенно функции высокого уровня, которые работают не с отдельными секторами, а с файлом в целом, для уменьшения числа параметров пользуются стандартным буфером. В MS-DOS такой буфер носит название **DTA (Disk Transfer Area)**. Пользовательские программы с помощью специальной функции ОС могут изменять положение DTA. Следовательно, если это потребуются, в некоторых случаях можно область DTA совместить с той областью ОЗУ, куда наша программа производит чтение файла. Естественно, данные тогда сразу же попадут “на свое место” и копирование просто не потребуются. Именно поэтому этап 7 на рис. 2 изображен пунктиром как не всегда обязательный.

Таким образом, мы подробно разобрали, как именно происходит чтение файла. Мы видели, что исключительно

ную роль в данной процедуре играет операционная система. Благодаря ей путем целой серии действий среди служебной информации диска находится запись, описывающая требуемый файл, и затем в конечном счете она заменяется физическими координатами секторов, входящих в состав файла. После этого остается считать необходимые сектора в ОЗУ и переписать в необходимую область памяти.

## Что изменится при записи

Интересно подумать над вопросом, что изменится в описанной выше процедуре, если файл будет не читаться, а, наоборот, сохраняться на диск. Многие этапы при этом останутся прежними, хотя, конечно, на определенных этапах изменения все же возникнут.

В частности, **первые два этапа** (выбор устройства и загрузка каталога) никак не изменятся.

**Этап 3** в соответствии с логикой действия будет иной. Файл с указанным именем при этом будет проверен, и, если окажется, что он есть, ОС потребует подтверждение операции сохранения, поскольку она приведет к удалению старой версии файла. При положительном ответе дополнительно будет произведено удаление файла. Данная процедура во многом похожа на чтение файла, только вместо передачи цепочки номеров кластеров на следующие этапы все они будут освобождены. Например, в случае файловой системы FAT во все элементы таблицы, соответствующие кластерам файла, будут занесены нули. Запись каталога будет помечена как удаленная: согласно традициям MS-DOS первая буква имени файла заменяется некоторым фиксированным характерным кодом. Важно понимать, что сектора с содержимым файла остаются неизменными, так что, несмотря на удаление, информация не исчезла и, по крайней мере теоретически, ее возможно восстановить.

Еще более существенным отличием данного этапа при записи будет подбор незанятых информацией кластеров для нового файла и их пометка как входящих в файл. Наиболее понятно, как это делать при наличии FAT-таблицы: надо искать в ней нулевые значения, т.к. именно они соответствуют свободным кластерам. Что касается других ОС, например, UNIX, то там для этой цели предусмотрена битовая карта занятости кластеров: каждому свободному кластеру соответствует нулевой бит, а занятому — единичный. Учитывая, что в системе команд процессора предусмотрены логические операции над отдельными битами, анализ данной карты существенной трудности не представляет.

Интересно подумать над вопросом о “политике” заполнения диска информацией. Допустим, у нас имеется диск, который давно находится в эксплуатации, т.е. на нем много файлов и многие старые файлы подвергались удалению. Для простоты рассуждений будем считать, что пользователь нашего гипотетического диска не обременяет себя применением необычайно по-

<sup>10</sup> Еще раз напомним, что этот процесс в современных ПК происходит без непосредственного участия центрального процессора под руководством контроллера прямого доступа к памяти.



лезной для диска процедуры дефрагментации. В таких условиях на диске файлы чередуются с освобожденными от данных местами. Пусть теперь мы хотим записать некоторый файл. Куда? Возможен целый ряд алгоритмов.

Первый, самый простой, состоит в максимальном заполнении начальной области диска. Для этого перебираем кластеры в порядке возрастания номеров и заполняем все попавшиеся свободные. Очевидно, что данный алгоритм имеет, как минимум, два недостатка: сильную фрагментацию создаваемых файлов и повышенный износ начальной области диска. Второй способ состоит в проверке всех свободных отрезков (порядка расположенных свободных кластеров) и заполнении того из них, который ближе всего (в идеале равен) длине нового файла. Такой метод трудноприменим, если размер файла заранее неизвестен, например, когда текстовый файл набирается с клавиатуры. Наконец, третий способ состоит в том, что система находит самый последний из занятых кластеров и записывает новый файл после него, игнорируя пустые места в начале. В таком “неэкономичном” алгоритме, пока диск не заполняется до конца, все новые файлы вообще не содержат фрагментации, а поверхность диска нагружается данными более равномерно. В принципе возможны и другие стратегии, а также сочетание любых из них.

В книге [9], например, по этому поводу говорится следующее:

“DOS заполняет диск от внешнего края внутрь. ...DOS просматривает ТРФ<sup>11</sup>, находит свободный кластер на цилиндре 1 и назначает его файлу. Затем ищет следующий свободный кластер и находит его где-то ближе к центру диска, скажем, на цилиндре 10. Новый файл оказывается фрагментированным.

В идеале DOS должна бы просматривать таблицу размещения файлов в поисках блока свободных кластеров, достаточного по величине для размещения файлов. Однако такой способ требует много времени, а при заполненном диске его использовать просто нельзя. DOS пришлось бы взять на себя функции дефрагментатора. Некоторые операционные системы так и делают, но DOS не принадлежит к их числу.

**Улучшенный метод.** Однако, начиная с версии 3.0, в DOS были внесены изменения, способствующие уменьшению фрагментации. В процессе заполнения диска DOS фиксирует номер ячейки ТРФ, которая использовалась для назначения последнего кластера. Когда требуется новый кластер, DOS начинает поиск с этого места ТРФ. Иначе говоря, DOS сначала назначает все кластеры по одному разу и лишь после этого возвращается назад и переназначает кластеры, освобожденные в ходе стирания файлов. ...После того как каждый кластер на диске

был однажды использован, DOS переходит в режим версий 2.x назначения кластеров, ближайших к цилиндру 0. Этот режим в дальнейшем уже не меняется. Описанная процедура на какое-то время существенно уменьшает фрагментацию диска”.

Читатели могут самостоятельно поэкспериментировать с дискетой и попытаться определить конкретную стратегию ее заполнения для своей версии ОС.

После того как необходимые свободные кластеры найдены, **этапы 4 и 5** по тем же самым формулам производят пересчет их номеров в физические координаты секторов.

Зато **этапы 6 и 7** меняются местами. Естественно, что сначала сохраняемые данные копируются в буфер, а затем следует их запись на диск. Процедура ROM BIOS для записи на диск секторов выглядит необычайно похоже: она отличается только номером функции в регистре **ah** (см. этап 6), все остальные параметры те же самые. Что касается этапа переписи, то меняется лишь направление перемещения данных: если при чтении данные копировались из буфера в программу, то теперь, наоборот, из программы в буфер. Так что и здесь изменения минимальны.

В итоге мы видим, что схема процесса записи отличается от чтения не столь существенно, как это можно было ожидать. Желющие смогут самостоятельно переделать *рис. 2* для схемы сохранения файла.

#### Литература

1. Олифер В.Г., Олифер Н.А. Сетевые операционные системы. СПб.: Питер, 2002, 544 с.
2. Таненбаум Э. Современные операционные системы. СПб.: Питер, 2004, 1040 с.
3. Немет Э., Снайдер Г., Хейн Т. Руководство администратора Linux. М.: Издательский дом “Вильямс”, 2004, 880 с.
4. Гринзоу А. Философия программирования для Windows 95/NT. СПб.: Символ-Плюс, 1997, 640 с.
5. Нортон П. ПК фирмы IBM и операционная система MS-DOS. М.: Радио и связь, 1992, 416 с.
6. Гук М. Аппаратные средства IBM PC. Энциклопедия. СПб.: Питер, 2003, 923 с.
7. Гук М. Дискровая подсистема ПК. СПб.: Питер, 2001, 336 с.
8. Джордейн Р. Справочник программиста персональных компьютеров типа IBM PC, XT, AT. М.: Финансы и статистика, 1992, 554 с.
9. Нортон П., Джордейн Р. Работа с жестким диском IBM PC. М.: Мир, 1992, 560 с.
10. Еремин Е.А. Популярные лекции об устройстве компьютера. СПб.: ВHV-Петербург, 2003, 272 с.
11. Еремин Е.А. Коллективное проектирование имитатора операционной системы. / Информатика и образование, 2005, № 7, с. 85–90.

<sup>11</sup> В цитате использовано русское сокращение от “таблица размещения файлов”, эквивалентное FAT; подобная практика перевода не прижилась.



# НАЧАЛКА

газета-клуб для всех,  
кто учит информатике  
маленьких детей



№ 11 (1–15 сентября)

## Начальная школа в летнем лагере

Ю.А. Первин,

г. Переславль-Залесский

Многолетняя работа и общение с А.П. Ершовым научили его коллег внимательнее относиться к динамике информационного общества. Например, в начале 80-х годов прошлого века произошло замечательное, но почти никому не замеченное событие: стоимость одного печатного знака в компьютере стала дешевле, чем полиграфическая реализация символа на бумаге — в газете, журнале, книге. На это событие обратил внимание А.П. Ершов, отметив, что именно его и следует считать началом эпохи информационного общества.

В ушедшем учебном году мне дважды довелось наблюдать такие знаковые события. Проводя в начале учебного года традиционные встречи в студенческих группах, я, как всегда, задавал студентам вопросы: есть ли у них доступ к компьютеру, есть ли у них электронный адрес в сети. С каждым годом после этого вопроса в аудитории поднимается все больше и больше рук. И вот в начале 2005–2006 учебного года многие сказали о возможности доступа к компьютеру, а на вопрос об электронной почте почти все без исключения подтвердили такую открытую для них возможность. Возможно, такой новой информации в какой-то степени способствовало то, что беседа эта проходила со столичными студентами, да и речь шла о группе, специализирующейся на кафедре информатики. Но факт остается фактом: такой информации достаточно, чтобы конструктивно менять методику преподавания информатики в вузах. Но это отдельный, хотя и вполне назревший повод для того, чтобы поговорить еще об одном вехе новой эпохи.

А начало каникулярного лета — самый конец июня — мне пришлось (уже не первый год) проводить в детском летнем компьютерном лагере сельских школ Переславского

района Ярославской области в селе Нагорье<sup>1</sup>. Нагорьевская школа — самая большая в Переславском районе. И в известном смысле — одна из самых хороших. Она стала единственной школой района среди тех, кто выиграл президентский грант, о котором в этом году так много говорят в школах. Когда три года тому назад лагерь только замыслился, казалась разумной, обоснованной и реальной перспектива того, что принимать гостей со всего района на неделю летних каникул будут по-очереди разные сельские школы района. Но жизнь показала, что руководители Нагорьевской школы смогут организовать подобное мероприятие лучше, чем любая другая школа. Так или иначе (впрочем, и этот, по-своему интересный вопрос надо рассматривать отдельно, не в этой заметке), но в традиции лагеря уже прочно вошли не только компьютеры и программы, кафедры и конференции, но и кабинеты информатики именно Нагорьевской школы. В школе — два компьютерных класса (на 600 учеников), но школьники из других уголков района приезжают в лагерь со своими (школьными) машинами. На всех хватает!

Дети разделяются (в соответствии с их желаниями) на несколько кафедр, число которых растет год от года. Те, кто постарше, увлеченно работают на кафедре сайтостроения. Ребятам из пятого-шестого класса интересны кафедры презентаций и компьютерной графики. Многих привлекают кафедры с манящими названиями — “Английский с компьютером”, “Немецкий с компьютером”. А совсем “молодым”, учащимся начальной школы, всегда предлагалась кафедра, за которой устоялось название “Здравствуй, компьютер!”. Сюда рекомендуют отправлять новичков, вообще говоря, независимо от возраста. Новичками всегда считались ученики, которых следовало учить работать с клавиатурой, “щелкать” кнопками мыши, редактировать текст, копировать символы и фрагменты текста, перетаскивать по экрану ярлыки информационных объектов... Кафедра “Здравствуй, компьютер!” была организована и на этот раз, но... о, чудо, среди новичков этой кафедры — школьников из начальных классов сельских школ района, удаленного и от столицы, и от област-

<sup>1</sup> Следует признать, что “чистота эксперимента” (в этот лагерь в принципе могут попасть только ученики-сельские) объясняется не дидактическими и не социальными аргументами, а носит исключительно политический (точнее, локально-политический) характер: Переславский район — единственный в Ярославской области район, где в силу амбиций местных руководителей существуют две отдельные, малодружественные власти: городская (город Переславль-Залесский) и сельская (Переславский сельский муниципальный округ).

ного центра, на этот раз не оказалось ни одного, кому были бы нужны такие простейшие знания: все они свободно чувствовали себя за компьютером. Причина, конечно, не в том, что их родители обладают компьютерами (таких в районе пока еще совсем немного, увы). К сожалению, причина и не в том, что во всех сельских школах грамотно поставлены курсы раннего обучения информатике (это тоже пока не так). Дело в том, что имеющиеся в сельских школах компьютерные классы не пусты. Правда, далеко не всегда у экрана компьютера в школьном кабинете информатики сидят усердные отличники, готовящие задания по информатике, или кружковцы, занимающиеся в дистанционном обучающем центре (хотя надо признать, что ежемесячные семинары сельских учителей информатики по сайтостроению и пропедевтическому курсу “Роботландии” не могли не дать результат). Однако ясно, что большая доля коллектива кафедры “Здравствуй, компьютер!” приехала в лагерь с большим опытом бездумных компьютерных игр — стрельбок, догонялок, космических войн и проч. И хотя “игровые” увлечения детей стали в лагере одной из забот для педагогов и воспитателей, все же полученные детьми “игроками” навыки общения с компьютером удалось направить в “мирное русло”.

Если о новой методике занятий со студентами высших учебных заведений еще нужно размышлять, готовить новые учебно-методические материалы и пособия и, возможно, дискутировать, то на кафедре “Здравствуй, компьютер!” летом 2006 года надо было идти на уроки с новой методикой буквально “с колес”.

Например, пришлось отказаться от практики, проверенной за первые два года существования лагеря, когда, следуя испытанным рекомендациям великого французского педагога С.Френе и великого русского писателя М.Горького, преподаватели готовили с детьми книгу не только собственного сочинения, но и собственного издания — “Мы из Роботландии”. Такая книга состояла из автобиографических рассказов детей и встроенных в word’овский текст цветных цифровых фотографий-портретов.

Ребенку очень важно прожить лагерную жизнь в многообразной и динамической среде. Включение компьютерного книгоиздательства в программу недельного лагеря оправдывалось тем, что простые навыки компьютерного набора не успевали наскучить за столь короткий недельный срок детям, у которых совмещались знакомство с навыками текстовой обработки информации и их усвоение. Детям, уже владеющим навыками динамического управления компьютером, но сохраняющим тяготение к методу проектов, к завершеному продукту, необходимо иное наполнение учебной программы.

Естественной идеей в такой ситуации оказался пакет ПервоЛого, разработанный группой С.Ф. Сопрунова в Центре образования “Технологии обучения” с сохранением основных идей и задумок С.Пейперта. ПервоЛого отлично совмещается с идеологией метода проектов. Проект включает в себя полный цикл разработки программного продукта от задумки через испытания (отладку) к конечному результату с помощью многопрофильного инструментария. Кроме того, проекты способны обеспечить многообразие сюжетов. Эти качества были очень востребованы в условиях лагеря с короткой сменой.

Но сильная сторона метода проектов — коллективизм, базирующийся на учете индивидуальных способностей и увлечений учащегося, — не могла стать методической основой для работы кафедры: провести обоснованно эффективное разделение детей на микрогруппы по их роли в проекте (проектах) не представлялось возможным из-за краткости срока. Хотя в лагере (в том числе и на кафедре “Здравствуй, компьютер!”) были и “старожилы”, приехавшие сюда во второй или даже в третий раз, в целом группа новичков была новой. Было ясно, что основой для содержания учебного процесса должна стать серия индивидуальных проектов, выполняемых одновременно, синхронно.

В такой серии был выделен один “подготовительный” проект, в котором дети, познакомившись с основными инструментами и возможностями ПервоЛого, придумали, построили и озвучили свои проекты-пейзажи, в большинстве из которых удалось встроить простую анимацию — горизонтальные и вертикальные перемещения моноформ (форм, не меняющихся в динамике выполнения проекта). А основу серии составили три последующие проекта, сгруппированные вокруг одной, главной темы — “Олимпийские игры”. Таким образом, каждый ученик прошел через четыре основных проекта и один дополнительный — изготовление визитных карточек с встроенным фотопортретом.

Самую большую сложность (после выбора инструментария) составил отбор содержания, существенно ограничиваемый временем. Дети провели в лагере шесть неполных дней. Дни приезда и отъезда, разумеется, были нерабочими. Еще один день был целиком “съеден” экскурсионно-развлекательной программой. И еще (примерно половина дня) — традиционной встречей с интересными людьми (в этом году в Нагорье приезжал известный научный сотрудник “Роботландии” А.А. Дуванов) и заключительной конференцией: выступлениями кафедральных команд с рассказом о проделанной работе, награждениями и вручениями. Отбор содержания из ПервоЛого на три с половиной рабочих дня был действительно нелегким. Многие соблазнительные механизмы — семафоры, процедуры, списки... — пришлось, скрепя сердце, отложить до “лучших времен”. Но и отобранный материал оказался велик: инструментарий ПервоЛого; альбом, его сохранение; формы Черепашки, работа с формами; обучение Черепашки; команды перемещения и повтора; одинарное выполнение команд и цикл; текстовое и звуковое сопровождение проекта; редактирование и конструирование команд; параметры команды; механизм случайного выбора значений параметров; рекурсия; реакция Черепашки на цвет...

Вот, что получилось (ниже приводится план; этот план практически полностью реализован):

#### 26 июня

Немного истории.

Сеймур Пейперт и его Черепашка.

Портрет С.Пейперта. Что такое Лого? А что такое ПервоЛого?

Запуск программы.

Как выглядит пустой альбом.

Лист альбома, имя альбома. Меню. Изменение размеров листа.

Рисовалка.

Ящик инструментов.  
Пульт управления, Палитра.  
Кнопка откатки.  
Формы, команды управления.

*Первый учебный проект.* Пейзаж с движущимся автомобилем.

Фон. Контур (выбор параметров карандаша), заливка цветом (выбор цветов).

Рождение Черепашки. Одежда Черепашки — ее формы. Перемещение черепашек по экрану.

Увеличение и уменьшение черепашек. Штамп, рисование леса.

Упражнение: установка деталей пейзажа — дом, клумба, пруд, солнышко (штамповка форм), облака (распылитель), рисование дороги (карандаш).

Чем отличается Черепашка от других форм?

Поворот головы Черепашки перед одеванием.

Основные инструменты — Заливка, Распылитель, Карандаш, Ручка, Прямоугольник и Овал, контурные и полные.

Учим Черепашку двигаться. Ключ из ящика инструментов. Окно обучения. Выбор команды и установки ее параметров. Шаг Черепашки в команде *Иди*. Одинарное выполнение и цикл. Как остановить цикл? Команды *Засни*, *Проснись*, *Выключи все*.

Движение автомобиля с торможением (команда *Часы*). Работа с текстом. Выбор кегля, шрифта и цвета. “Прозрачный” цвет.

Музыкальное сопровождение проекта. Выбор и запуск мелодий.

Как записать музыку? Как ее запустить? Разница между звучанием музыки в команде и в фоне.

Первый самостоятельный проект (допустима работа парами). Дети сами придумывают сюжеты: лесной пейзаж, полевая дорога, морской пейзаж, берег реки, ...

Именованье и сохранение альбомного листа. Выбор (чтение) нужного альбома.

### 27 июня

#### *Проект “Рисующая Черепашка”*

Перо Черепашки, команды *Поднять перо* и *Опустить перо*.

Цепочка учебных команд. Пунктир (команды управления подъемом перья). Изготовление значка формы.

Бесконечное рисование — “рекурсивный” вызов.

Дополнительные команды. Поворот. Угловая мера.

Команда рисования квадрата. “Циклический” квадрат.

Черепашка, движущаяся по кругу. Команда смены цвета пера (и Черепашки). Рисование олимпийской эмблемы.

#### *Проект “Олимпийские игры”*

Постановка географической задачи — перенести олимпийский огонь на самолете из Афин (Олимпийские игры-2004) в Пекин (Олимпиада-2008).

Установка на карте форм точек-городов. Подписывание географических точек. Изготовление самолета — зеркальное отражение готовой формы.

Расчет маршрута (подбор параметров проектируемой команды).

#### *Проект “Гонки”*

Реакция Черепашки на цвет. Программирование поведения мячика, отскакивающего от стены. Замечания по поводу толщины стены.

“Многочерепашие”. Выделение черепашек. Какой черепашке дается команда?

Для чего существуют похожие формы?

Выбор участников гонок. Изготовление форм для участника-собаки.

Рисование интерфейса гонок: финишная черта, стартовая черта, надписи на “стартовых колодках”.

Программирование каждого участника гонок. Случайный выбор шага — игральные кости. Демонстрация отдельных участников.

Общая картина гонок.

Гонки под музыку.

### 29 июня (до обеда)

Самостоятельное творчество детей над проектами, завершение.

#### *Проект “Визитная карточка”*

Обсуждение задачи. Подготовка текста, редактирование и форматирование.

Фотографирование. Перенос фотографий в компьютер. Графические файлы, просмотр фотографий. Размещение портрета на визитной карточке. Печать визиток.

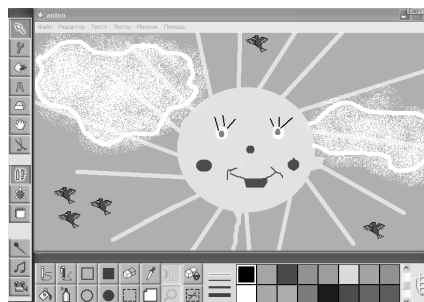
### 30 июня

Рекомендация по подготовке и оформлению. Изготовление значков для создаваемых команд.

Коллективный просмотр проектов, обсуждение их и репетиция выступления на итоговой конференции.

Каждый из “студентов” кафедры (от 8 до 10 лет) подготовил 7 проектов: проект-“пейзаж”, олимпийский флаг, полет на Олимпиаду, проект “Гонки” (в трех версиях-модификациях — четвертьфинал, полуфинал и финал) и визитная карточка. Визитная карточка была показана (в приветственном помахивании) при выступлении на конференции, а остальные 6 проектов дети показывали на экране компьютера в ходе итоговой конференции, которая проходила в информационном центре (одном из кабинетов) Нагорьевской школы. Команда кафедры выступала первой (в силу своего молодого возраста). Каждый ученик-докладчик демонстрировал один из своих проектов самостоятельно, а руководитель сопровождал эти демонстрации своим конферансом.

Три проекта-пейзажа показали Илья Митяков (Купанская школа), Антон Лях (Нагорьевская школа, *рис. 1*) и Маша Федорова (Смоленская школа, *рис. 2*). Маша назвала свой проект “Морское путешествие”, хотя скольжение морского парусника по волнам сопровождалось мелодией И.Штрауса “На прекрасном голубом Дунае”.



*Рис. 1*

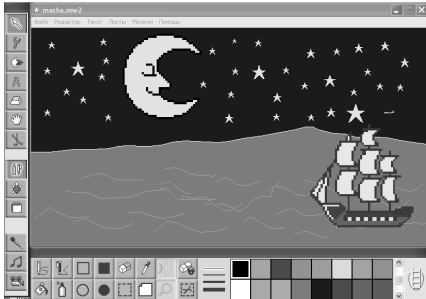


Рис. 2

На занятиях после “пейзажных” проектов прочно заняли свое место разговоры о спорте (столь актуальные в жаркие июньские дни этого года), о всемирном спортивном празднике — Олимпиаде, о ее эмблеме и ее флаге. Пришлось коллективно вспоминать о цвете флага, о кольцах на флаге и об их цвете. С трудом и не по порядку вспомнили цвета колец, еще больше труда доставило соотнесение цветов к континентам Земли. На конференции Настя Докторова (Филимоновская школа) показала созданный ею (и Черепашкой, которую Настя научила рисовать цветные окружности) олимпийский флаг.

“С олимпийским флагом можно уже отправляться в путь. Последней столицей Олимпиады была столица Греции — Афины (совсем непросто было вспомнить этот спортивно-географический факт). А лететь нам предстояло в Пекин — китайскую столицу, где состоятся следующие олимпийские игры (про Пекин с трудом вспомнили, но показать на карте не смог никто). Итак, мы в Афинах, самолет уже подан, и трап поднят. Полетели. За штурвалом автор проекта, штурман 1-го класса Сережа Жарков (из Брембольской школы). Благодаря его мастерству самолет приземлился точно в нужном месте — на аэродроме Пекина”. Кстати, “штурманский” расчет (подбор параметров) и конструирование команды полета Черепашки, на которую надо было надеть форму самолета, предварительно ее ориентируя в нужном направлении, оказались совсем не простым делом (рис. 3).

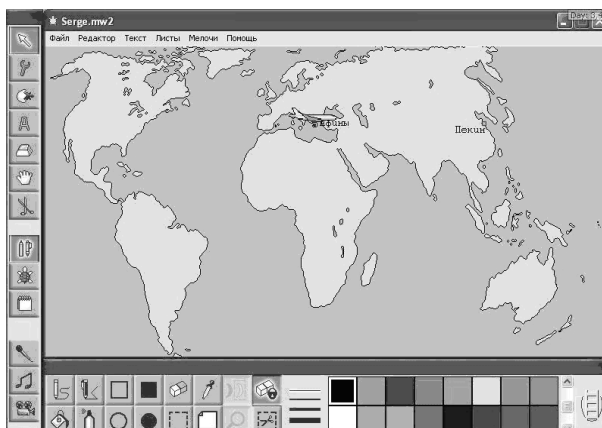


Рис. 3

“Ну вот, мы и в Пекине. Мы не просто смотрели на олимпийские соревнования, но и стали их участниками. В четвертьфинал попали пять спортсменов из нашего лагеря, которых тренирует Вова Жарков (тоже из Брембольской школы, младший брат “штурмана”). В четвертьфинал олимпийского забега он вывел пятерых спортсменов. Вот они (рис. 4): собака Жучка, конь Петя, мальчик Юра, девочка Ира, лев Сережа”.

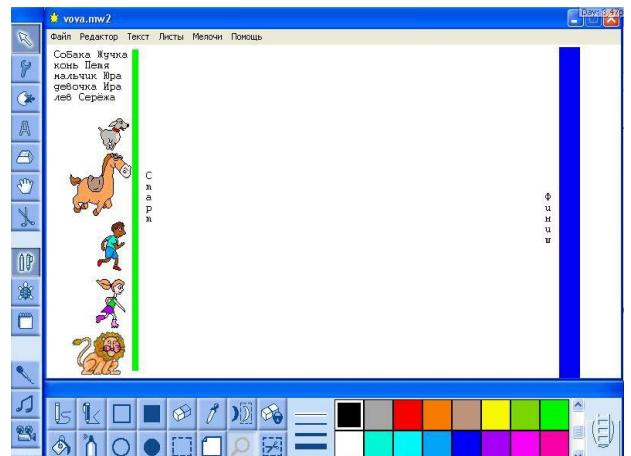


Рис. 4

“Теперь перед вами полуфинальный забег. Судья на старте — Алена Романова (Ивановская школа). В полуфинале бегут три участника: конь Скаунд, девочка Маша и мальчик Петя. Но хотя Алена подобрала в качестве мелодии, сопровождающей этот забег, музыку С.Прокофьева из сказки-оперы “Петя и волк”, мальчик Петя все-таки в финал не попал — он пришел последним.

И вот, наконец, финальный забег. Судьей на финише этого забега стал Андрюша Моисеев (Берендеевская школа). В забеге бегут девочка Галя и пес Шарик (рис. 5). Старт! За кого вы болеете?”

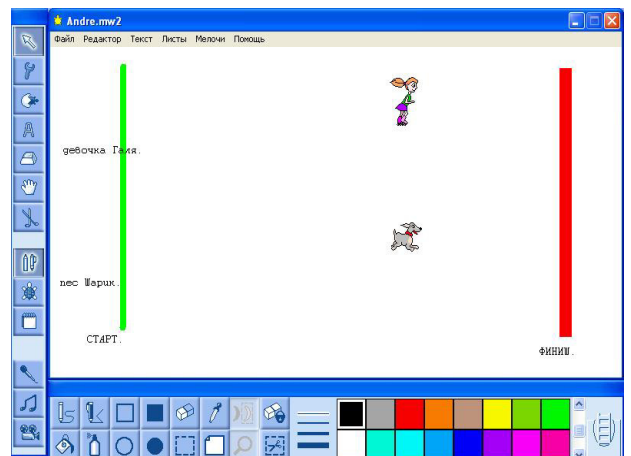


Рис. 5



# В мир информатики

# 78 (1—15 сентября)

Газета для пытливых учеников  
и их талантливых учителей

## Самым активным читателям — Похвальная грамота газеты “Информатика”

За активное участие в конкурсах для учащихся Похвальными грамотами газеты “Информатика” награждены:

— Абдуллин Рамиль, средняя школа деревни Старый Бабич Кармаскалинского р-на Республики Башкортостан, учитель **Абдуллин Р.Ф.**;

— Басыров Дамир и Ялаева Эмилия, г. Уфа, Республика Башкортостан, гимназия № 3, учитель **Колоскова Л.Н.**;

— Агеева Марья, Винакур Валерий, Исаев Николай, Ревтов Андрей, Ротко Мария, Кудряшов Евгений, Малявкин Георгий, Ногтев Сергей, Приходько Ирина, Смирнов Юрий, Харлапанова Злата, Шелюх Василий, пос. Надвоицы, Республика Карелия, школа № 1, учитель **Богданова Л.М.**;

— Анненков Евгений и Чапкевич Михаил, г. Орел, лицей № 4, учитель **Чапкевич И.М.**;

— Ванщикова Наталья, г. Лесосибирск Красноярского края, пос. Стрелка, школа № 8, учитель **Лопатин М.А.**;

— Белкин Максим, г. Тула, гимназия № 11, учитель **Леонова Е.С.**;

— Бережных Анастасия, средняя школа села Качикатцы Хангалусского улуса, Республика Саха (Якутия), учитель **Яковлева М.Д.**;

— Бурцев Анатолий и Крупин Федор, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Видякин Иван, Кирютин Алексей, Лось Олеся и Умряева Ирина, г. Струнино Владимирской обл., школа № 11, учитель **Волков Ю.П.**;

— Гайсина Галия и Гайсин Рашит, г. Уфа, Республика Башкортостан, школа № 18, учитель **Искандарова А.Р.**;

— Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.Л.**;

— Гордиенко Евгений, средняя школа села Гилевка Алтайского края, учитель **Збарах В.Ф.**;

— Горшкова Светлана, Мухарметова Эльвина, Орлов Александр и Шкенева Наталья, г. Стерлитамак, Республика Башкортостан, школа № 1, учитель **Орлова Е.В.**;

— Губанова Марина, Идиятов Владимир и Шалагина Анна, г. Ревда Свердловской обл., школа № 28, учитель **Кольцова Е.М.**;

— Иванова Татьяна и Колокольцева Ксения, г. Сегежа, Республика Карелия, школа № 5, учитель **Меньшиков В.В.**;

— Исакова Олеся, пос. Лимбяяха Новоуренгойского р-на Тюменской обл., школа № 1, учитель **Исакова И.С.**;

— Клипов Игорь, г. Балашов Саратовской обл., гуманитарно-педагогический лицей-интернат, учитель **Сухорукова Е.В.**;

— Копылова Валерия, пос. Жатай, Республика Саха (Якутия), школа № 1, учитель **Копылова Л.Ю.**;

— Кузнецова Елена и Стафеева Елизавета, г. Новоуральск Свердловской обл., школа № 58, учитель **Стафеева Н.А.**;

— Летова Елена и Хаустова Кристина, Вознесенская основная школа Красногвардейского р-на Оренбургской обл., учитель **Гриднев А.Б.**;

— Митин Илья, г. Зеленокумск Ставропольского края, школа № 3, учитель **Ржевский Д.В.**;

— Пудова Наталья, г. Прокопьевск Кемеровской обл., школа № 3, учитель **Шерстнева Л.М.**;

— Силаев Валерий, Ардатовский аграрный техникум, пос. Ардатов Нижегородской обл., преподаватель **Касаткина С.Ю.**;

— Усманов Сергей, село Актаныш Актанышского р-на, Республика Татарстан, школа № 2, учитель **Гилязова Г.М.**;

— Шамсутдинова Альбина, г. Лениногорск, Республика Татарстан, школа № 8, учитель **Кашапова Р.Х.**;

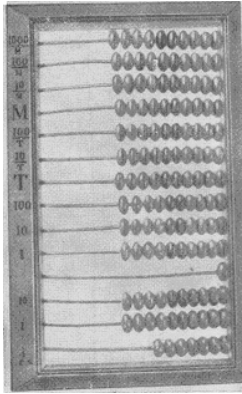
— Щербакова Анастасия, рабочий поселок Яя Кемеровской обл., школа № 2, учитель **Щербакова О.Б.**;

— Ямкина Анна, г. Ульяновск, школа № 37, учитель **Ямкина Е.В.**;

— Яценко Роман, г. Новочеркасск, Суворовское училище МВД РФ, преподаватель **Воронкова О.Б.**

## Усовершенствованные счеты

В статьях [1–2] было рассказано об истории возникновения русских счетов.



Счеты XIX века

Более чем двухвековой опыт работы на счетах в России и недостатки существовавших счетных приборов привели к тому, что в XIX веке в России стали приспособлять счеты к возросшим требованиям вычислительной практики.

Известно, что А.Бок (проживавший в Варшаве) предложил в 1812–1813 гг. некоторый вид счетов. Правда, никаких сведений об этих счетах не сохранилось.

В 1828 г. генерал-майор Ф.М. Свободской предложил свой счетный прибор после многих лет работы на нем. Этот прибор состоял из соединенных в общей раме нескольких счетов, чаще всего употреблялось 12 счетов, но встречались наборы до 30 штук. Для передвижения костяшек служил специальный прут с рукояткой. Кроме четырех арифметических действий, Свободской производил много различных операций, достигая при этом большой скорости. Например, извлечение кубического корня из 21-значного числа занимало 3 минуты. Он вычислял также сложные проценты, возводил числа в различные степени и т.п. Основное условие скорости счисления на счетах Свободского — это строгое соблюдение единообразных правил. Все операции сводились к действиям сложения и вычитания.

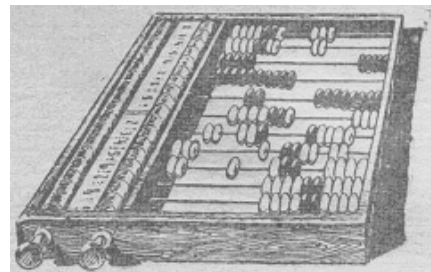
Положительный отзыв на счеты Свободского дала Петербургская академия наук. Преподавание вычислений на счетах по методу Свободского было введено в Петербургском (с 1829 г.), Московском (с 1830 г.) и Харьковском (с 1830 г.) университетах. Вышел ряд книг, посвященных этим счетам, появились статьи в разных журналах. Но, несмотря на все это, к середине XIX в. счеты Свободского были уже почти забыты. Они оказались слишком громоздкими, да и сама идея вернуть счетам их универсальный характер была явно несостоятельна.

В 1861 г. И.Бураков предложил счеты, в которых было 20 полных рядов и один с четырьмя костяшками. Кроме арифметических дей-

ствий, на этих счетах возводили в степень и извлекали корни. На счеты Буракова были похожи счеты, предложенные Марковым.

А.Большаном в 60-х годах были изготовлены свои счеты с 9 косточками в полных рядах. На этих счетах можно было возводить в степень, извлекать корни, вычислять сложные проценты и выполнять четыре арифметических действия.

В 1872 г. известный счетовод, организатор Общества счетоводов Ф.В. Езерский сконструировал счеты с машинкой для умножения и деления. Вдоль длинной планки для умножения и деления. Вдоль длинной планки рамы этих счетов помещены два валика, на которых накручены таблицы. Вращая валики, можно было получать частные произведения, которые затем складывались на счетах.



В 1882 г. Н.Компанейский описал свои двойные счеты. Прибор состоит из счетов и валиков, причем оси валиков шли параллельно проволокам счетов и, кроме того, валики могли передвигаться относительно проволок. Это представляет удобство для правильного сдвига при сложении частных произведений.

При умножении и делении больших чисел на счетах Ф.В. Езерского и Н.Компанейского скорость несколько больше, чем при производстве этих действий на бумаге.

Усовершенствование счетов продолжалось и в XX веке. А.Талалай в 1903 г. издал вспомогательные таблицы, при помощи которых на счетах было удобно умножать.

В 1921 г. Б.Н. Компанейский создал прибор, который представляет собой соединение довольно удобных таблиц умножения с обыкновенными счетами. На этом приборе, кроме четырех арифметических действий, можно было решать задачи на проценты, оперировать с дробями и т.п. Относительно этого прибора техническое бюро Комитета по делам изобретений НТО ВСНХ вынесло 25 января 1921 г. следующее постановление: “Признавая пользу предложенного арифмометра всюду, где требуются точные расчеты, несомненно рекомендовать изобретение Б.Н. Компанейского... в качестве прибора, могущего с успехом заменить механические арифмометры существующих систем”.

Попыток усовершенствовать счеты было много, но создать универсальный счетный прибор на основе счетов все же не удалось.

### Литература

1. Абак в России. / “В мир информатики” № 67, 69 (“Информатика” № 2, 4/2006).
2. Русские счеты. / “В мир информатики” № 75 (“Информатика” № 10/2006).



## Ответы, решения, разъяснения

к заданиям,  
опубликованным в газете  
"В мир информатики"  
№ 70 ("Информатика"  
№ 5/2006)

### 1. Статья "Четыре инженера"

#### Ответы

1. Профессии инженеров: Борисов — химик, Данин — автомеханик, Кириллов — строитель, Савин — радиомеханик.
2. Возрастной порядок (по убыванию возраста): Савин, Борисов, Кириллов, Данин.
3. Порядок по лыжному мастерству: Данин (лучший), Борисов, Савин, Кириллов.
4. Порядок по шахматному мастерству: Савин (лучший), Борисов, Данин, Кириллов.

#### Ответы прислали:

— Агеева Марья, Винакур Валерий, Исаев Николай, Кудряшов Евгений, Малявкин Георгий, Ногтев Сергей, Приходько Ирина, Смирнов Юрий, Харлапанова Злата и Шелюх Василий, пос. Надвоицы, Республика Карелия, школа № 1, учитель **Богданова Л.М.**;

— Вайзерт Владислав и Егорова Кристина, пос. Лимбьяха Новоуренгойского р-на Тюменской обл., школа № 1, учитель **Исакова И.С.**;

— Гайсина Галя и Гайсин Рашит, г. Уфа, Республика Башкортостан, школа № 18, учитель **Искандарова А.Р.**;

— Загайнова Наталья, Несмачных Татьяна, Никешина Ольга и Савинова Мария, г. Белово Кемеровской обл., пос. Краснобродский, школа № 31, учитель **Зайцева Л.А.**;

— Кузнецова Евгения, г. Ярославль, школа № 33, учитель **Головина Л.И.**;

— Лепихин Евгений, Ургушевский лицей, Республика Башкортостан, учитель **Мадияров Ю.М.**;

— Мельник Алексей, г. Лениногорск, Республика Татарстан, школа № 8, учитель **Кашапова Р.Х.**;

— Парунакян Араик, Москва, школа № 1194, учителя **Кулаженова Н.В.** и **Никонорова Л.Н.**;

— Попков Денис и Садыки Роман, г. Новоуральск, Центр внешкольной работы, педагог **Тумашова Е.В.**;

— Сенёва Анастасия и Шмуляев Андрей, г. Струнино Владимирской обл., школа № 11, учитель **Волков Ю.П.**;

— Хузина Эльвира, село Николо-Берёзовка, Республика Башкортостан, Краснокамский р-н, школа № 1, учитель **Ситдикова А.Г.**

### 2. Статья "Зачет по логике"

#### Решение

Каждый из трех студентов оказался в одной и той же ситуации — видел перед собой товарищей в двух белых шапках — и рассуждал следующим образом:

"Допустим, на мне черная шапка. Тогда кто-то из двух других студентов видит одну белую и одну черную шапку и думает над цветом своей шапки. Допустим, он рассуждает так:

"Если на мне (думает он о себе) черная шапка, то третий студент сразу же определил бы цвет своей шапки — белый. Но он молчит, значит, на мне шапка белого цвета».

Но так как этот второй студент молчит, значит, он не видит черную шапку, следовательно, на мне тоже белая шапка".

Так рассуждал каждый из трех студентов, видя перед собой товарищей в белых шапках, и каждый пришел к выводу о белом цвете своей шапки, причем сделали студенты это одновременно.

#### Ответы прислали:

— Абдуллин Рамиль, средняя школа деревни Старый Бабич Кармаскалинского р-на Республики Башкортостан, учитель **Абдуллин Р.Ф.**;

— Агеева Марья, Винакур Валерий, Исаев Николай, Кудряшов Евгений, Малявкин Георгий, Ногтев Сергей, Приходько Ирина, Смирнов Юрий, Харлапанова Злата и Шелюх Василий, пос. Надвоицы, Республика Карелия, школа № 1, учитель **Богданова Л.М.**;

— Винникова Олеся и Лобода Андрей, г. Старый Оскол Белгородской обл., школа № 24, учитель **Винникова О.Е.**;

— Гайсин Рашит, г. Уфа, Республика Башкортостан, школа № 18, учитель **Искандарова А.Р.**;

— Кузнецова Евгения, г. Ярославль, школа № 33, учитель **Головина Л.И.**;

— Ларионова Анна, г. Новосибирск, лицей № 22 "Надежда Сибири", учитель **Гора С.Г.**;

— Мельник Алексей, г. Лениногорск, Республика Татарстан, школа № 8, учитель **Кашапова Р.Х.**;

— Рашитов Сергей, г. Ленск, Республика Саха (Якутия), школа № 3, учитель **Захарченко Е.А.**;

— Яценко Роман, г. Новочеркасск, Суворовское училище МВД РФ, преподаватель **Воронкова О.Б.**

### 3. Статья "Модель колеса"

*Виды кривых, имеющих имена:* 1. Архимедова спираль. 2. Астроида. 3. Генератриса. 4. Гипербола. 5. Гиперболическая спираль. 6. Гипотрохоида. 7. Гипоциклоида. 8. Декартов лист. 9. Кардиоида. 10. Квадратриса. 11. Клотоида (спираль Корню).

12. Конхоида Никомеда. 13. Косекансоида. 14. Косинусоида. 15. Котангенсоида. 16. Кохлеоида. 17. Кубическая парабола. 18. Лемниската. 19. Лемниската Бернулли. 20. Логарифмика. 21. Логарифмическая спираль. 22. Локон Аньези. 23. Локсодрома. 24. Овал Кассини. 25. Окружность. 26. Окружность Апполония. 27. Парабола. 28. Полукубическая парабола. 29. Секансоида. 30. Синусоида. 31. Строфоида. 32. Тангенсоида. 33. Трактриса. 34. Улитка Паскаля. 35. Цепная линия. 36. Циклоида. 37. Циссоида. 38. Эвольвента. 39. Эллипс. 40. Эпициклоида.

Перечень привела Кузнецова Елена, г. Новоуральск Свердловской обл., школа № 58, учитель **Стафеева Н.А.** Редакция решила наградить Елену дипломом. Поздравляем!

*Ответы,  
решения,  
разъяснения*

**к заданиям,  
опубликованным в газете  
"В мир информатики"  
№ 71 ("Информатика"  
№ 65/2006)**

### 1. Статья "Квадратная рамка из костей домино"

Сумма очков всех сторон искомого квадрата должна равняться  $44 \times 4 = 176$ , т.е. на 8 больше, чем сумма очков на косточках полного набора домино (168). Происходит это, конечно, оттого, что числа очков, занимающих вершины квадрата, считаются дважды. Сказанным определяется, какова должна быть сумма очков на вершинах квадрата — 8. Это несколько облегчает поиски требуемого расположения костей домино, хотя нахождение его довольно трудоемкое. Трудностей не испугались (☺) и нашли правильное решение:

— Морозова Дина, Республика Татарстан, г. Казань, школа № 22, учитель **Осипова А.А.**:

- 1) верхняя строка (слева направо):  
0, 0-6, 6-6, 6-5, 5-1, 1-1, 1-3, 3-0;
- 2) правая строка (сверху вниз):  
0, 0-4, 4-1, 1-6, 6-4, 4-4, 4-2, 2-2;
- 3) нижняя строка (слева направо):  
6-3, 3-3, 3-5, 5-5, 5-0, 0-1, 1-2, 2;
- 4) левая сторона (сверху вниз):  
0-0, 0-2, 2-5, 5-4, 4-3, 3-2, 2-6, 6;

— Пехов Олег, пос. Белый Яр Верхнекетского р-на Томской обл., школа № 1, учитель **Слезко П.С.**:

- 1) верхняя строка (слева направо):  
0, 0-3, 3-4, 4-0, 0-6, 6-6, 6-2, 2-2;
- 2) правая строка (сверху вниз):  
2, 2-1, 1-1, 1-5, 5-6, 6-3, 3-3, 3-2;
- 3) нижняя строка (слева направо):  
4-5, 5-2, 2-4, 4-6, 6-1, 1-0, 0-2, 2;
- 4) левая сторона (сверху вниз):  
0-0, 0-5, 5-5, 5-3, 3-1, 1-4, 4-4, 4.

Молодцы!

Имеются и другие решения, причем не только те, которые получаются при "поворотах" описанных рамок.

В ряде присланных ответов были использованы не все кости полного набора домино.

### 2. Статья "Какой может быть цифра?"

Напомним, что в задаче предлагалось определить, какой *может быть* последняя цифра числа  $A$  с учетом известных условий.

*Решение.* Прежде всего установим, какими цифрами оканчиваются квадраты натуральных чисел. Из табл. 1 следует, что такими цифрами являются 0, 1, 4, 5, 6 и 9.

Таблица 1

Последняя цифра натурального числа $x$	Последняя цифра числа $x^2$
0	0
1	1
2	4
3	9
4	6
5	5
6	6
7	9
8	4
9	1

В табл. 2 приведены данные о том, какой может быть последняя цифра числа  $A$ , чтобы соответствовать каждому из двух утверждений в условии задачи. При заполнении этой таблицы учтена информация из табл. 1.

Таблица 2

Возможная последняя цифра	Может ли быть такая цифра у числа $A$ ?	
	Из утверждения 1	Из утверждения 2
0	Да	Нет
1	Нет	Нет
2	Нет	Да
3	Да	Да
4	Да	Да
5	Да	Нет
6	Нет	Нет
7	Нет	Да
8	Да	Да
9	Да	Да

Так как последняя цифра числа  $A$  должна быть такой, чтобы соответствовать как утверждению 1, так и утверждению 2, то, используя операцию логического умножения, или дизъюнкции (как же обой-



тись без логики — ☺), получим, что последней цифрой числа  $A$  могут быть 3, 4, 8 и 9.

Правильные ответы прислали:

— Агеева Марья, Винакур Валерий, Исаев Николай, Кудряшов Евгений, Малявкин Георгий, Ногтев Сергей, Приходько Ирина, Смирнов Юрий, Харлапанова Злата и Шелюх Василий, пос. Надвоицы, Республика Карелия, школа № 1, учитель **Богданова Л.М.**;

— Александров Алексей, г. Новочеркасск, Суворовское училище МВД РФ, преподаватель **Воронкова О.Б.**;

— Бурцев Анатолий, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Винникова Олеся, г. Старый Оскол Белгородской обл., школа № 24, учитель **Винникова О.Е.**;

— Гайсин Рашид, г. Уфа, Республика Башкортостан, школа № 18, учитель **Искандарова А.Р.**;

— Клипов Игорь, г. Балашов Саратовской обл., гуманитарно-педагогический лицей-интернат, учитель **Сухорукова Е.В.**;

— Кузнецова Евгения, г. Ярославль, школа № 33, учитель **Головина Л.И.**;

— Морозова Дина, Республика Татарстан, г. Казань, школа № 22, учитель **Осипова А.А.**;

— Пехов Олег, пос. Белый Яр Верхнекетского р-на Томской обл., школа № 1, учитель **Слезко П.С.**;

— Холод Максим, г. Холмск Сахалинской обл., лицей “Надежда”, учитель **Романькова С.Н.**

Некоторые читатели провели более детальное исследование и определили, что из перечисленных *возможных* цифр в числе  $A$  последней является цифра 4.

### 3. Статья “Городские слухи”

Ответы прислали:

— Агеева Марья, Винакур Валерий, Исаев Николай, Кудряшов Евгений, Малявкин Георгий, Ногтев Сергей, Приходько Ирина, Смирнов Юрий, Харлапанова Злата и Шелюх Василий, пос. Надвоицы, Республика Карелия, школа № 1, учитель **Богданова Л.М.**;

— Баженов Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Бурцев Анатолий, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Гайсин Рашид, г. Уфа, Республика Башкортостан, школа № 18, учитель **Искандарова А.Р.**;

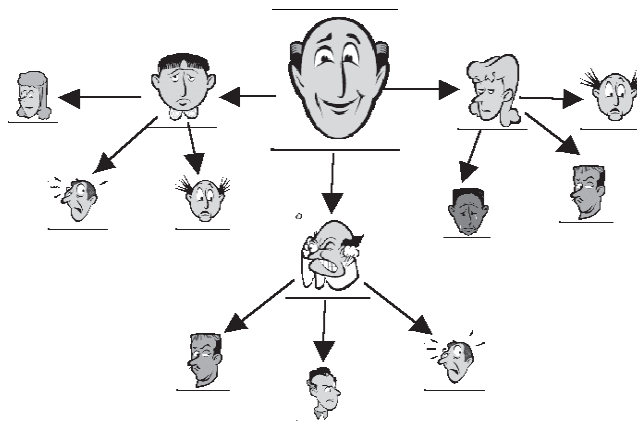
— Деминцев Борис, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Иванченко Любовь, средняя школа рабочего поселка Пинеровка, Саратовская обл., Балашовский р-н, учитель **Пичугин В.В.**;

— Кузнецова Евгения, г. Ярославль, школа № 33, учитель **Головина Л.И.**;

— Невидимый Сергей, Москва, гимназия № 1530, учитель **Шамшев М.В.**;

— Юргин Андрей, основная школа села Кубайка Красноярского края, учитель **Семеновский В.А.**



Задача может быть решена несколькими способами. В статье предлагалось решить ее с использованием компьютера. Это можно сделать

1) с применением электронной таблицы Microsoft Excel:

	А	В	С
1	Время	Количество людей, узнавших новость	Общее количество людей, знающих новость
2	8:00	1	1
3	8:15	3	4
4	8:30	9	13
5	8:45	27	40
6	9:00	81	121
7	9:15	243	364
8	9:30	729	1093
9	9:45	2187	3280
10	10:00	6561	9841
11	10:15	19683	29524
12	10:30	59049	88573
13			

Примечание. Формулы для расчета запишите самостоятельно.

2) с помощью компьютерной программы, которая на школьном алгоритмическом языке выглядит так:

```

алг Задача_о_городских_слухах
нач цел узнают_вновь, всего_будут_знать,
      номер_шага
узнают_вновь := 1
всего_будут_знать := 1
номер_шага := 0
нц
  номер_шага := номер_шага + 1
  узнают_вновь := узнают_вновь * 3
  всего_будут_знать := всего_будут_знать +
    узнают_вновь
кц_при всего_будут_знать >= 50000
вывод номер_шага
кон

```

**Примечание.** По программе определяется величина номер\_шага — количество интервалов времени по 15 мин., через которые общее количество знающих новость превысит 50 000. Зная эту величину, легко найти соответствующее время.

Из таблицы следует, что ранее чем в половине одиннадцатого утра поголовно все жители Компьютеринска будут осведомлены о новости, которая в 8 часов утра была известна только одному человеку. Ясно, что такой же результат будет получен и при решении другими способами.

Несколько способов решения этой задачи описала Любовь Иванченко (в статье использован рисунок из ее письма). Редакция решила наградить Любовь дипломом. Поздравляем!

Решения всех задач, рассмотренных выше, представил Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.А.** Он же прислал результаты проведенного исследования, предложенного в статье “Блуждания случайного жука” (см. газету-вкладку “В мир информатики” № 66 / “Информатика” № 1/2006). По данным Дмитрия:

— вероятность падения жука при  $T = 60$  сек. равна 0,88;

— среднее расстояние, на которое удалится живой жук от центра через  $T = 60$  сек., равно 3,31 см.

Ответы на задания, связанные с разработкой так называемых “N — S-диаграмм” (им была посвящена статья в газете-вкладке “В мир информатики” № 68 / “Информатика” № 3/2006), прислали:

— Агеева Мария, Винакур Валерий, Исаев Николай, Кудряшов Евгений, Малявкин Георгий, Ногтев Сергей, Приходько Ирина, Смирнов Юрий, Харлапнова Злата и Шелюх Василий, пос. Надвоицы, Республика Карелия, школа № 1, учитель **Богданова Л.М.**;

— Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.А.**;

— Нигматуллин Руслан, село Раевский Альшеевского р-на Республики Башкортостан, школа № 2, учитель **Евграфова Н.Ф.**

Учащиеся из Надвоицкой школы и Дмитрий Глижинский представили также правильные решения следующих задач:

- 1) “Кто изображен на портрете?”;
- 2) “Школьный турнир по футболу”;
- 3) “В магазине”,

а также заданий для самостоятельной работы, предложенных в статье “О троичной системе счисления”.

Кроме того, эти учащиеся правильно решили задачи из статьи “Системы счисления и их применение”, опубликованной в “основной” газете, а не в газете-вкладке “В мир информатики”. Спасибо вам, ребята!

Правильное решение задачи “Однофамильцы в поезде” прислали также Бочкова Елена и Мирхабудинова Регина, г. Белебей, Республика Башкортостан, школа № 17, учитель **Дремина Е.Е.**

Результаты исследования особенностей использования клавиш для работы с регистром памяти в программе “Калькулятор” — стандартном прило-

жении операционной системы Windows — представила Копылова Валерия, пос. Жатай, Республика Саха (Якутия), школа № 1, учитель **Копылова Л.Ю.**

Правильное решение буквенного ребуса, опубликованного в газете-вкладке “В мир информатики” № 67 (“Информатика” № 2/2006), прислали также Кузьмина Элина, Малюжанцева Лилия и Цоктоева Адиса, Республика Бурятия, село Кижинга, Кижингинский лицей, учитель **Дашиева Д.Р.** Адиса и Элина представили также правильный ответ на вопрос о том, кто изображен на портрете, а Элина, кроме того, — решение задачи о разделе семи хлебов между восемью людьми.

Бурцев Анатолий, г. Ярославль, школа № 33, учитель **Ярцева О.В.**, представил правильное решение задачи об “умном” словаре гномиков и задачи “В магазине”.

Решение задачи, опубликованной в статье “Школьный турнир по футболу”, прислал также Пряхин Даниил, пос. Лимбьяха Новоуренгойского р-на Тюменской обл., школа № 1, учитель **Исакова И.С.**

За участие в подготовке ответов на задание, связанное с определением “лишнего термина”, дипломом награждены также Копылов Иван и Судовых Ольга, учащиеся средней школы села Кирицы Спаского р-на Рязанской обл., учитель **Мосарыгина О.В.**

\* \* \*

## Шестнадцать студентов

Шестнадцать студентов возвращались после зимних каникул в Москву. Оказалось так, что среди них было четыре жителя Тулы: А., Б., В. и Г., четыре — из г. Орел: Д., Е., Ж. и З., четыре — из Курска: И., К., Л. и М. и четыре — из Белгорода: Н., О., П. и Р.

Оказалось еще, что А., Д., И. и Н. недавно исполнилось 20 лет; Б., Е., Л. и О. — 21 год; В., Ж., М. и П. — 22 года и Г., З., М. и Р. — 23 года.

Среди них было четыре математика, четыре химика, четыре геолога и четыре биолога, причем любые четверо студентов одной специальности были из разных городов и разного возраста.

Четверо студентов учились на первом курсе, четверо — на втором, четверо — на третьем и четверо — на четвертом, причем любые “однокурсники” были из разных городов, разного возраста и разных специальностей.

Наконец, четверо оказались ярыми футболистами, четверо — боксерами, четверо — волейболистами и четверо — шахматистами, и каждые четыре любителя одного вида спорта были из разных городов, разного возраста, разных специальностей и учились на разных курсах.

Установите для каждого студента факультет, курс и любимый вид спорта, если известно, что И. — волейболист, Е. — футболист, В. — биолог, Г. — математик первого курса, шахматист, Ж. — химик второго курса, шахматист, и К. — геолог третьего курса, шахматист.

Ответы присылайте в редакцию (адреса указаны в рубрике “Внимание! Конкурс”).



## Игра Баше

Ю.П. Волков,  
учитель информатики школы № 11  
г. Струнино Владимирской обл.

В 1612 году во французском городе Лионе вышла в свет книга “Приятные и занимательные задачи”. Ее автор — французский математик, поэт и переводчик Гаспар Клод Баше де Мезариак (1581—1638). Испытывая чувство зависти, сообщим читателям о том, что Баше де Мезариак прекрасно владел французским, итальянским, латинским и греческим языками. Но нас интересуют не его успехи в области изучения языков, а игра, описанная в его книге. Вот один из вариантов игры.

На столе выложены  $n$  предметов (спичек, монет, камешков и т.п.). Играют двое. Они поочередно забирают несколько предметов, причем заранее договорено, что число забранных предметов не превышает  $k$  ( $0 < k \leq n$ ). Проигрывает тот, кто своим ходом вынужден забрать последний предмет<sup>1</sup>.

Имеется выигрышная тактика для игрока, берущего предметы вторым. Читателям предлагаем определить, в чем она заключается. Всегда ли возможно реализовать эту тактику, чтобы обязательно выиграть?

Читателям, владеющим каким-либо языком программирования, предлагаем составить программу, моделирующую игру человека и компьютера. Программа должна предусматривать ввод с клавиатуры значений  $n$  и  $k$  ( $k \leq n/3$ ), а также разыгрывать случайным образом первый ход. Компьютер должен выбирать ходы в соответствии с выигрышной тактикой (см. чуть выше).

### Указания по решению

Определите, какие ситуации являются проигрышными для игрока, делающего очередной ход. Такой ситуацией, в частности, является положение, когда на столе остался 1 предмет. Найдите другие такие ситуации и установите общее правило, определяющее, что ситуация проигрышна для игрока, делающего очередной ход.

**Game.exe**

## “Ломаем” голову

### Коля, Оля и тетя Поля

Когда Коля был молод, как Оля,  
Столько лет было тетушке Поле,  
Сколько Коле теперь вместе с Олей.  
Сколько лет было Коле,  
Когда тетушка Поля  
Была в возрасте Коли?

Ответ, пожалуйста, присылайте в редакцию.

<sup>1</sup> Имеется также вариант игры, в котором выигрывает тот, кто своим ходом заберет все оставшиеся предметы. — **Ред.**

## Конкурс № 49 для учащихся

Приведите примеры использования в наше время цифр римской системы счисления.

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

## Итоги конкурса № 42 для учащихся

Напомним, что задание конкурса было связано с ответами на вопросы, касающиеся алгоритмов решения задач.

Ответы прислали:

- Абалаков Александр и Полушкин Вадим, Кемлянская средняя школа Ичалковского р-на Республики Мордовия, учитель **Силантьев О.П.**;
- Абдрахманов Артур, г. Учалы, Республика Башкортостан, школа № 2, учитель **Галимова З.Н.**;
- Баженов Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;
- Гордиенко Евгений, ученик средней школы села Гилевка Алтайского края, учитель **Збарах В.Ф.**;
- Дебелый Константин, Дубина Светлана и Стаднюк Ирина, пос. Лимбьяха Новоуренгойского р-на Тюменской обл., школа № 1, учитель **Исакова И.С.**;
- Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.Л.**;
- Денисова Татьяна и Попова Татьяна, Москва, школа № 1194, учитель **Кулаженова Н.В.**;
- Дьячкова Татьяна, г. Стерлитамак, Республика Башкортостан, школа № 12, учитель **Дмитриева О.В.**;
- Капустина Ольга и Соловьяненко Валентина, г. Новохоперск Воронежской обл., ПТУ № 45, преподаватель **Липина В.М.**;
- Приходько Марина, средняя школа села Дохновичи Брянской обл., учитель **Семенцова Е.В.**;
- Рашитов Сергей, г. Ленск, Республика Саха (Якутия), школа № 3, учитель **Захарченко Е.А.**;
- Ткаченко Анна, средняя школа села Подсосное Назаровского р-на Красноярского края, учитель **Кривова Н.П.**;
- Чапкевич Михаил, г. Орел, лицей № 4, учитель **Чапкевич И.М.**;
- Шмуляев Андрей, г. Струнино Владимирской обл., школа № 11, учитель **Волков Ю.П.**

**Внимание!  
Конкурс**

Все перечисленные читатели правильно и достаточно полно ответили на заданные вопросы, и выделить кого-либо редакция не сочла возможным. Спасибо всем!

## Итоги конкурса № 43 для учащихся

Напомним, что необходимо было предложить способ передачи текста, изображенного в виде иероглифов, с помощью азбуки Морзе по телеграфу или по рации.

*Правильный ответ.* Китайские и японские радиосты передают тексты не с иероглифов, а с алфавитного письма. Текст сначала произносят, затем изображают последовательность букв слова или словосочетания (каждая буква — звук), а потом уже перекодировывают каждую букву в последовательность точек и тире. До и после передачи такое письмо представлено “нормально”, в виде иероглифов.

Ответы прислали:

— Баженов Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.Л.**;

— Деминцев Борис, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Ильин Геннадий, Парунакян Араик и Черевко Александр, Москва, г. Зеленоград, школа № 1194, учителя **Кулаженова Н.В.** и **Никонова Л.Н.**;

— Мешков Григорий, Москва, лингвистический лицей № 1555, учитель **Холина И.Н.** (Григорий описал в письме принципы построения иероглифов японского языка);

— Трубицын Сергей и Зюзин Сергей, г. Новохоперск Воронежской обл., ПТУ № 45, преподаватель **Липина В.М.**;

— Хаустова Кристина, Вознесенская основная школа Красногвардейского р-на Оренбургской обл., учитель **Гриднев А.Б.**

В присланных ответах предлагается тем или иным способом кодировать каждый иероглиф. Но поскольку, как отметила в своем письме Кристина Хаустова, в китайском языке около 50 000 иероглифов, радисты не смогут запомнить последовательность точек и тире даже для наиболее часто используемых иероглифов. В то же время описанный выше способ передачи текста требует знания кодов только каждой из букв.

Редакция благодарит всех участников этого конкурса, задание которого было непростым.

## Итоги конкурса № 44 для учащихся

Напомним, что задание конкурса было связано с получением числа 2006 из цифр 1, 2, ..., 9 или чисел, образованных этими цифрами, а также с аналогичной задачей получения чисел 1, 2, ..., 20 из цифр 2, 0, 0 и 6.

Ответы прислали:

— Агеева Мария, Винакур Валерий, Исаев Николай, Кудряшов Евгений, Малявкин Георгий, Ногтев Сергей, Приходько Ирина, Смирнов Юрий, Харлапанова Злата и Шелюх Василий, пос. Надвоицы, Республика Карелия, школа № 1, учитель **Богданова Л.М.**;

— Бурцев Анатолий, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Вайзерт Владислав, пос. Лимбяха Новоуренгойского р-на Тюменской обл., школа № 1, учитель **Исакова И.С.**;

— Гайсин Рашид, г. Уфа, Республика Башкортостан, школа № 18, учитель **Искандарова А.Р.**;

— Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.Л.**;

— Волков Никита, Республика Карелия, г. Медвежьегорск, школа № 2, учитель **Лешукова Н.М.**;

— Калякина Дарья, Новобураский р-н Саратовской обл., Ириновская средняя школа, учитель **Брунов А.С.**;

— Клипов Игорь, г. Балашов Саратовской обл., гуманитарно-педагогический лицей-интернат, учитель **Сухорукова Е.В.**;

— Митин Илья, г. Зеленокумск Ставропольского края, школа № 3, учитель **Ржевский Д.В.**;

— Морозова Дина, Республика Татарстан, г. Казань, школа № 22, учитель **Осипова А.А.**;

— Осипов Алексей, пос. Садовый Новосибирской обл., школа № 70, учитель **Юрцева Е.В.**;

— Парунакян Араик и Черевко Александр, Москва, г. Зеленоград, школа № 1194, учителя **Кулаженова Н.В.** и **Никонова Л.Н.**;

— Победимова Светлана и Трунова Елена, г. Новокузнецк, школа № 100, учитель **Папшева В.В.**;

— Самойлов Иван, г. Тобольск, гимназия № 10 им. М.Д. Лицмана, учитель **Махмутова З.К.**;

— Стафеев Дмитрий, г. Новоуральск Свердловской обл., школа № 58, учитель **Стафеева Н.А.**;

— Твердохлебова Елена, г. Старый Оскол Белгородской обл., школа № 24, учитель **Винникова О.Е.**;

— Филонов Евгений и Фролов Артем, средняя школа поселка Тавричанка Альшеевского р-на, Республика Башкортостан, учитель **Корнеева М.В.**

Победителем конкурса признаны Анатолий Бурцев, представивший наибольшее число примеров, и



Елена Твердохлебова, единственная из всех участников конкурса, нашедшая решение применительно к числу 17 ( $17 = 20 - 0! - \lfloor \sqrt{6} \rfloor$ , где символы  $\lfloor \cdot \rfloor$  [соответствуют целой части числа, расположенного между ними]). Поздравляем! Отметим также оригинальные решения Артема Фролова и Евгения Филонова, использовавших функции *cos* и *exp*.

## Итоги конкурса № 45 для учащихся

Напомним, что необходимо было ответить на вопрос, почему жесткий магнитный диск, лазерный диск и другие устройства долговременной памяти называют *внешней памятью*, ведь они находятся *внутри* системного блока компьютера.

*Ответ.* Термин “внешняя память” имеет историческое происхождение. На ЭВМ первых поколений память, находившаяся внутри процессора, размещенного в отдельном шкафу, или плотно примыкавшая к этому шкафу, получила естественное название *внутренней*. А память, сконструированная в виде отдельных устройств для хранения информации (например, накопители на магнитных барабанах и лентах), не менее естественно стали называть *внешней*. Современная трактовка этих терминов для компьютеров типа IBM PC такая:

— *внутренняя память* — электронная (полупроводниковая) память, установленная на системной плате или на платах расширения;

— *внешняя память* — память, реализованная в виде устройств с различными принципами хранения информации и обычно с подвижными носителями. Устройства внешней памяти могут размещаться как в системном блоке компьютера, так и в отдельных корпусах (Еремин Е.А. Популярная лекция об устройстве компьютера. СПб.: БХВ-Петербург, 2003).

Ответы прислали:

— Агеева Марья, Винакур Валерий, Исаев Николай, Кудряшов Евгений, Малявкин Георгий, Ногтев Сергей, Приходько Ирина, Смирнов Юрий, Харлапанова Злата и Шелюх Василий, пос. Надвоицы, Республика Карелия, школа № 1, учитель **Богданова Л.М.**;

— Александров Алексей и Яценко Роман, г. Новочеркасск, Суворовское училище МВД РФ, преподаватель **Воронкова О.Б.**;

— Букаева Любовь и Чапкевич Михаил, г. Орел, лицей № 4, учитель **Чапкевич И.М.**;

— Буранбаев Нур, село Акъяр Хайбуллинского р-на, школа № 1, учитель **Юлдашбаев А.М.**;

— Валеев Руслан и Нигматуллин Руслан, Республика Башкортостан, Альшеевский р-н, село Раевский, школа № 2, учитель **Евграфова Н.Ф.**;

— Валиев Карим, Республика Татарстан, г. Казань, школа № 9, учитель **Орлова Ф.Х.**;

— Винникова Олеся, г. Старый Оскол Белгородской обл., школа № 24, учитель **Винникова О.Е.**;

— Гайсина Галя и Гайсин Рашит, г. Уфа, Республика Башкортостан, школа № 18, учитель **Искандарова А.Р.**;

— Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.А.**;

— Евсик Андрей, средняя школа села Благодатное Петровского р-на Ставропольского края, учитель **Левшин Г.В.**;

— Загафуранова Айгуль, средняя школа деревни Сейтяк Балтачевского р-на, Республика Башкортостан, учитель **Загафуранова А.Ф.**;

— Калякина Дарья и Молодцов Сергей, Новобураский р-н Саратовской обл., Ириновская средняя школа, учитель **Брунов А.С.**;

— Копылова Валерия, пос. Жатай, Республика Саха (Якутия), школа № 1, учитель **Копылова Л.Ю.**;

— Лобода Андрей, г. Старый Оскол Белгородской обл., школа № 24, учитель **Винникова О.Е.**;

— Минченко Александр, средняя школа села Дохновичи Брянской обл., учитель **Семенцова Е.В.**;

— Мусина Ляйсан, г. Учалы, Республика Башкортостан, лицей № 12, учитель **Рязанова Ч.П.**;

— Победимова Светлана, г. Новокузнецк, школа № 100, учитель **Папшева В.В.**;

— Петухов Артем, г. Шарья Костромской обл., школа № 21 (фамилия учителя информатики в письме, к сожалению, не указана);

— Попков Денис, Попков Илья и Садыки Роман, г. Новоуральск, Центр внешкольной работы, педагог **Тумашова Е.В.**;

— Рашитов Сергей, г. Ленск, Республика Саха (Якутия), школа № 3, учитель **Захарченко Е.А.**;

— Сизов Леонид, село Комсомольское, Республика Чувашия, школа № 1, учитель **Родионов П.В.**;

— Трубицын Сергей, г. Новохоперск Воронежской обл., ПТУ № 45, преподаватель **Липина В.М.**;

— Ференец Анна, г. Норильск Красноярского края, школа № 9, учитель **Романюк Т.А.**;

— Черевко Александр, Москва, г. Зеленоград, школа № 1194, учителя **Кулаженова Н.В.** и **Никонорова Л.Н.** В подавляющем большинстве ответов термин *внешняя* в названии памяти связан с особенностями долговременного хранения информации на устройствах соответствующего типа, возможностью смены устройств и т.п. Наиболее близким к правильному ответу, в том числе к его историческому аспекту, является ответ Любови Букаевой, которая признана победителем конкурса и будет награждена дипломом. Поздравляем!

## Итоги конкурса № 46 для учащихся

Напомним, что необходимо было предложить такую форму записи условия, определяющего, есть ли среди заданных цифр трехзначного числа одинаковые, чтобы оно (условие) не было сложным, т.е. чтобы в нем не использовались логические связи **или**, **и**, **не** (**AND**, **OR**, **NOT**).

Ответы прислали:

- Агапкин Александр и Третьяков Алексей, г. Челябинск, школа № 97, учитель **Мазничевская Л.И.**;
- Агеева Марья, Винакур Валерий, Исаев Николай, Кудряшов Евгений, Малявкин Георгий, Ногтев Сергей, Приходько Ирина, Смирнов Юрий, Харлапана Злата и Шелюх Василий, пос. Надвоицы, Республика Карелия, школа № 1, учитель **Богданова Л.М.**;
- Бурцев Анатолий, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;
- Винникова Олеся, г. Старый Оскол Белгородской обл., школа № 24, учитель **Винникова О.Е.**;
- Глижинский Дмитрий, г. Бендеры, Республика Молдова, гимназия № 2, учитель **Глижинская С.А.**;
- Дорофеев Денис, Старозятцинская средняя школа, Удмуртская Республика, Якшур-Бодьинский р-н, учитель **Соболева Л.Ю.**;
- Калязинцев Александр, г. Струнино Владимирской обл., школа № 11, учитель **Волков Ю.П.**;
- Калякина Дарья, Новобураский р-н Саратовской обл., Ириновская средняя школа, учитель **Брунов А.С.**;
- Кононов Владимир, средняя школа поселка Ивот Дятьковского р-на Брянской обл., учитель **Брин Д.С.**;
- Морозова Дина, Республика Татарстан, г. Казань, школа № 22, учитель **Осипова А.А.**;
- Огнева Дарья и Семенов Станислав, Нижегородская обл., г. Саров, лицей № 15, учитель **Самсонова Л.Н.**;
- Пехов Олег, пос. Белый Яр Верхнекетского р-на Томской обл., школа № 1, учитель **Слезко П.С.**;
- Савин Михаил, средняя школа рабочего поселка Пинеровка, Саратовская обл., Балашовский р-н, учитель **Пичугин В.В.**;
- Стародубец Андрей, г. Холмск Сахалинской обл., лицей “Надежда”, учитель **Романькова С.Н.**;

— Чапкевич Михаил, г. Орел, лицей № 4, учитель **Чапкевич И.М.**

Большинство участников конкурса предложили оформить условие так, как показано в следующем фрагменте программы на школьном алгоритмическом языке:

```
если (a - b) * (a - c) * (b - c) = 0
  то
    вывод нс, "Среди заданных цифр есть
    одинаковые"
  иначе
    вывод нс, "Среди заданных цифр нет
    одинаковых"
```

**все**

Учащиеся из школы № 1 поселка Надвоицы работали также оригинальный вариант, верный для ненулевых цифр:

```
если a/b + b/c + c/a = b/a + c/b + a/c
  то
    вывод нс, "Среди заданных цифр есть
    одинаковые"
  иначе
    вывод нс, "Среди заданных цифр нет
    одинаковых"
```

**все**

Еще один оригинальный вариант предложила Олеся Винникова:

```
если  $\sqrt{abc \cdot acb \cdot bac \cdot bca \cdot cab \cdot cba} = abc \cdot bca \cdot cab$ 
  то
    вывод нс, "Среди заданных цифр есть
    одинаковые"
  ...
все
```

где  $abc, acb, \dots$  — числа, образованные цифрами  $a, b$  и  $c$ .

Олеся и все перечисленные выше ребята из Надвоицкой школы признаны победителями конкурса и будут награждены дипломами. Поздравляем!

Заметим, что в ряде ответов предлагалось использовать вложенный условный оператор, в то время как в условии конкурса речь шла о записи условия в приведенном фрагменте программы.

### 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

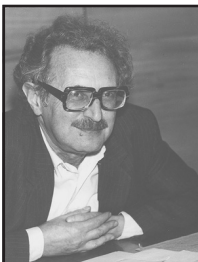
Дорогие коллеги! Мы обязательно поощрим любую активность ваших учеников. Предложите им поучаствовать в конкурсах (которые специально не слишком сложны) — уверены, ребятам будет приятно увидеть свою фамилию в газете, в списке приславших правильные ответы. Может быть, они заметят какие-то погрешности, 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.



## ДЕСЯТЫЕ СОЛОВЕЙЧИКОВСКИЕ ЧТЕНИЯ

29–30 сентября 2006 года. Московский городской дом учителя

### ТЕМА: «ПЕДАГОГИКЕ СОТРУДНИЧЕСТВА — 20 ЛЕТ»

Ровно двадцать лет назад, в октябре 1986 года, в подмосковном поселке Переделкино состоялась встреча педагогов-новаторов. Первыми в отечественной педагогической практике ответили на самый сложный вопрос школьной педагогики: как учить детей без отбора и отсева. Подводя итоги своей двадцатилетней экспериментальной работы (60–80-е годы), педагоги-новаторы независимо друг от друга разными путями пришли к одинаковым выводам, которые описываются емким понятием «педагогика сотрудничества». Отчет о встрече, написанный Симоном Соловейчиком, и стал манифестом педагогики сотрудничества.

Теперь, через двадцать лет, многие из авторов манифеста снова окажутся вместе. На чтения для встречи с учителями приглашены С.Н. Лысенкова, В.Ф. Шаталов, В.А. Караковский, М.П. Щетинин, Е.Н. Ильин, Ш.А. Амонашвили, Л.А. Никитина.

Вместе с педагогами-новаторами мы постараемся понять, что произошло со школой и с российским образованием за двадцать лет со дня публикации манифеста — двадцать лет, в которые уместилась перестройка и все, что последовало за ней.

**Дорогие учителя, приходите на чтения!**

**Вход, как всегда, свободный.**

Открытие чтений 29 сентября в 10 часов утра

Подробная программа чтений — в газете «Первое сентября» и на сайте [www.1september.ru](http://www.1september.ru)

Справки по телефону: (495) 249-31-38. Секретарь Людмила Владимировна Урбанская

Адрес Московского городского дома учителя: Москва, ул. Пушкинская, дом 4, строение 2. Проезд: центр, станция метро «Кузнецкий мост»



## Фестиваль педагогических идей «Открытый урок»

### ВСЕ МАТЕРИАЛЫ БУДУТ ОПУБЛИКОВАНЫ!

Дорогие коллеги! Издательский дом «Первое сентября» приглашает вас принять участие в фестивале педагогических идей «Открытый урок» 2006/07 учебного года.

### САМЫЙ МАССОВЫЙ ОТКРЫТЫЙ ПЕДАГОГИЧЕСКИЙ ФОРУМ!

В 2006/07 учебном году фестиваль «Открытый урок» проводится в четвертый раз. В фестивале 2003/04 учебного года приняли участие более 2600 педагогов, в 2004/05 учебном году — 4700, в прошлом году — более 5000.

Фестиваль стал самым массовым и представительным открытым педагогическим форумом. Материалы всех участников публикуются. Каждый участник получает полный комплект итоговых материалов, включающий:

- персональный диплом;
- сертификат, подтверждающий факт публикации материалов;
- книги - сборники тезисов всех статей;
- компакт-диски (CD-ROM) с полнотекстовыми версиями всех материалов.

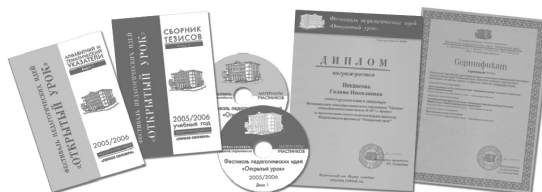
В специальном разделе представлена информация об учебных заведениях.

Со всеми материалами можно ознакомиться на сайте фестиваля

<http://festival.1september.ru>

#### КАК ПРИНЯТЬ УЧАСТИЕ В ФЕСТИВАЛЕ?

Формы заявок на участие в фестивале публикуются во всех газетах нашего Издательского дома. Заявки также можно подавать в режиме on-line на нашем сайте.





## РОБОТЛАНДСКИЙ УНИВЕРСИТЕТ ОТКРЫВАЕТ НАБОР СТУДЕНТОВ на 2006/2007 учебный год

В Университет принимаются индивидуальные и коллективные ученики.

**Коллективный студент** — это группа детей, работающая под руководством одного или нескольких наставников (наставник, как правило, школьный учитель).

**Индивидуальный студент** — это учитель, желающий пройти обучение индивидуально, без группы детей.

В конце годового курса при успешном обучении школьный учитель получает удостоверение от негосударственного образовательного учреждения "Роботландия+" и удостоверение государственного образца от Российской академии повышения квалификации и переподготовки работников образования.

Занятия в Университете платные (цены — на сайте [www.botik.ru/~robot](http://www.botik.ru/~robot)).

Они начинаются 10 октября текущего года и продолжаются в течение двух семестров до мая следующего года. Для подписчиков "Информатика" предусмотрена 10%-ая скидка за обучение.

Для получения более подробной информации можно написать по адресу [kurs@robotland.pereslavl.ru](mailto:kurs@robotland.pereslavl.ru) (администрация Университета) или заглянуть на сайт [www.botik.ru/~robot](http://www.botik.ru/~robot) (здесь можно отправить заявку, заполнив электронную форму).

### Краткое описание курсов

#### 11. Зимние вечера

Курс для младших школьников (3–4-е классы). Знакомство с алгоритмами и исполнителями, электронная почта; Вики-конструирование, детские курсы. Только для коллективных учеников.

12. *Азы информатики-I. Знакомимся с компьютером. Работаем с информацией* (4–5-е классы). Только для коллективных учеников.

14. *Азы информатики-II. Пишем на компьютере* (5–6-е классы). Только для коллективных учеников.

15. *Азы информатики-III. Рисуем на компьютере* (6–7-е классы). Только для коллективных учеников.

16. *Азы информатики-IV. Выходим в Интернет* (7–8-е классы). Новый курс: компьютерные сети, электронная почта, сетевые технологии. Только для коллективных учеников.

#### 21. Тропинка конструктора

Курс для детей 4–7-х классов, изучивших основы информатики. Создание компьютерных проектов, имеющих практическое применение на школьных уроках, на базе роботландских программ-конструкторов. Только для коллективных учеников.

#### 31. Азы программирования-I. Плюстик и Кукарача

Для детей 6–8-х классов, изучивших основы информатики (исполнитель, алгоритм, программа, первичные инструментальные навыки работы с компьютером). Этот курс для детей, которые нашли себя в программировании и желают повысить уровень своего мастерства, составляя головоломные алгоритмы для роботландских исполнителей Плюстик и Кукарача.

#### 32. Азы программирования-II. Корректор

Продолжение курса 31 на базе исполнителя Корректор.

#### 42. Web-конструирование

Для детей 6–8-х классов, изучивших основы HTML-документов. Основы проектирования, web-дизайна и юзабилити.

### ОБРАЗЕЦ ЗАЯВКИ

#### Заявка на обучение в Роботландском университете

Предполагаемый курс (отделение)	42. Web-конструирование
Форма обучения	Коллективный ученик
Руководитель (обучаемый)	Иванов Петр Николаевич
Профессия	Учитель информатики средней школы
Стаж работы в школе	3 года
Директор школы	Петров Семен Сергеевич
Электронный адрес	<a href="mailto:ivanov@sch62.nsk.ru">ivanov@sch62.nsk.ru</a>
Название учреждения и его почтовый адрес (с индексом)	630090, г. Новосибирск, ул. Жемчужная, 6, средняя школа № 62
Почтовый адрес для посылок	630091, г. Новосибирск, ул. Строителей, д. 45, кв. 124, Иванову Петру Николаевичу
Предполагаемое число детей в группе и их возраст (для коллективного ученика)	20 школьников, 10–11-е классы
Первый раз в Роботландском университете	Да
Руководитель (обучаемый) — подписчик газеты "Информатика"	Да
Откуда получена информация об университете	Газета "Информатика".

Заявки принимаются по адресу: [kurs@robotland.pereslavl.ru](mailto:kurs@robotland.pereslavl.ru) до 1 октября.

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

ГАЗЕТЫ  
ИЗДАТЕЛЬСКОГО ДОМА  
**Первое сентября**

гл. ред. — Е.В. Бирюкова,  
индекс подписки — 32025;

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

**Библиотека в школе**  
гл. ред. — О.К. Громова,  
индекс подписки — 33376;

**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. Сёмкина,  
индекс подписки — 32033;

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

**Искусство**  
гл. ред. — М.Н. Сартан,  
индекс подписки — 32584;

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

**Литература**  
отв. сек. — С.Ф. Дмитренко,  
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. Картелишвили  
С.Б. Кишкина  
Н.П. Медведева  
Ю.А. Первин (редактор вкладки  
"Началка")  
**Корректор** **Дизайн и верстка**  
Е.Л. Володина 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: 8-495-249-98-70

**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 17.08.2006.  
Номер подписан 17.08.2006.  
Заказ № 615517  
Цена свободная

**ИНДЕКС ПОДПИСКИ**  
**для индивидуальных подписчиков** **32291**  
**комплекта изданий** **32744**

Тел.: 8-495-249-31-38, 249-33-86. Факс 8-495-249-31-84

Internet: [inf@1september.ru](mailto:inf@1september.ru)  
WWW: <http://www.1september.ru>

**ИЗДАТЕЛЬСКАЯ ПОДПИСКА** Тел.: (499) 249-47-58 E-mail: [podpiska@1september.ru](mailto:podpiska@1september.ru)