

# ИНФОРМАТИК

*До встречи  
в новом году!*

Очередная встреча читателей с редакцией нашей газеты состоится 5 января 2000 г. в Перми. В рамках областной учительской конференции «Рождественские чтения» будет проведен круглый стол «Профессиональная газета учителя информатики», который состоится 5.01.2000 в 16<sup>00</sup> в 522-й аудитории 2-го корпуса ПГУ. Телефон оргкомитета конференции: (3422) 39-65-84 (Русаков Сергей Владимирович). Ждем вас, уважаемые читатели!

## Читайте в номере

### Уроки ..... 2–6

**Я.Н. Зайдельман. “Буки программирования”. Материалы Роботландского университета**

Что делать после того, как Азы пройдены? Конечно же, за Азами должны следовать Буки! Но сначала надо повторить Азы...

### Информатика в лицах ..... 7–13

**Иван Долмачев. Алан Тьюринг**

Продолжение биографии Алана Мэтисона Тьюринга (создателя ценной теоретической модели — машины Тьюринга), которая публикуется со многими отступлениями.

В представленной главе рассказывается главным образом об одном из самых оригинальных изобретений выдающегося математика Георга Кантора (1845—1918) — диагональном процессе.

### Тематический обзор ..... 14–26

Тематический обзор всего напечатанного в газете “Информатика” в 1999 году (№ 1—48) с аннотациями. Материалы собраны по разделам в хронологическом порядке.

### Внеклассная работа по информатике ..... 27–30

**Н.Л. Попова. Игра “Домино”**

Предлагаемое домино можно использовать на внеклассных мероприятиях по информатике. Затрагиваются темы: “Теоретическая информатика”, “DOS”, “Компоненты компьютера”, “Алгоритмизация”.



# “Буки программирования”

## Материалы Роботландского университета

**Я.Н. Зайдельман**

### Тема 1. Повторим Азы

**АЗЫ — начальные сведения.**

(Толковый словарь)

**БУКА — фантастическое страшное существо, которым пугают детей.**

(Толковый словарь)

**И снова, снова к тем азам...**

(Юрий Левитанский)

**Пятачок мужественно почесал за ухом и сказал, что до пятницы он совершенно свободен и с большим удовольствием пойдет с Пухом, в особенности если там Настоящий Бука.**

(А.Милн в пересказе Б.Заходера)

Наш курс называется “Буки программирования”. Как вы, наверное, догадываетесь, это вовсе не те Буки, которыми пугают детей. Наши Буки получили свое название из русской азбуки. Помните, как она начинается? Аз, Буки, Веди... Азами обычно называют начальные сведения, самые простые элементы какой-

то науки, то, с чего начинается обучение. А что делать после того, как Азы пройдены? Ну конечно же, за Азами обязательно должны следовать Буки! Вот ими мы и займемся.

Но, прежде чем углубляться в Буки, надо повторить Азы. Хотя бы для того, чтобы мы все точно знали: каждый ученик на наших курсах владеет всеми необходимыми основными понятиями и называет их одними и теми же словами.

Итак, начнем. Сначала мы немного поговорим о программировании вообще, а потом повторим основные понятия, которые должны быть вам уже знакомы. Если же по дороге встретится что-то новое, не пугайтесь и постарайтесь во всем разобраться. Все понятия, которые мы будем обсуждать при повторении Азов, не раз потом встретятся нам в Буках.

### Производственное и теоретическое программирование

Программирование можно разделить на производственное и теоретическое.

Производственное программирование — это искусство создания реальных программ, которые выполняются на реальных компьютерах. Теоретическое программирование — это искусство построения алгоритмов.

Для производственного программирования необходимо знать технические особенности конкретных компьютеров и правила взаимодействия с операционной системой, надо уметь строить удобный и понятный интерфейс, защищать программу от неправильных действий пользователя и т.д. и т.п.

Теоретическое программирование занимается решением задач. Есть исходные данные, нужно получить результат. При этом нас, как правило, не интересует, как именно будет происходить ввод данных, мы не заботимся о красивом внешнем виде, главное — получить правильный ответ. Тот, кто хоть раз участвовал в олимпиаде по информатике, хорошо представляет, о чем здесь идет речь.

Теоретическое программирование тесно связано с производственным. При разработке реальных производственных программ постоянно возникают задачи, для решения которых надо разрабатывать алгоритмы, то есть обращаться к теоретическому программированию. С другой стороны, именно задачи, возникающие в производственном программировании, дают толчок развитию теории — ведь их надо как-то решать, а для этого приходится придумывать алгоритмы.

В реальной жизни нужны в первую очередь специалисты-практики, люди, владеющие навыками производственного программирования, умеющие разрабатывать настоящие полезные программы. Но плох тот специалист, который не знает теории! Ведь в основе любых систем лежат одни и те же общие теоретические принципы. Поэтому в нашем курсе мы сосредоточимся только на теоретическом программировании и не будем рассматривать производственные вопросы.

## Программирование, язык, реализация

Приступая к изучению программирования, ученики часто спрашивают: “А какой язык мы будем учить?”. И всегда очень удивляются, услышав ответ: “Никакой. Мы вообще не будем учить язык”.

Давайте попробуем разобраться в этом вопросе.

Как мы уже знаем, теоретическое программирование — это разработка алгоритмов. Можно сказать, что программирование — это некоторый способ мыслить.

Какую роль в этом мышлении играет язык? Лингвисты уже очень давно спорят о взаимном влиянии мышления и языка. Мы не будем вмешиваться в их споры, но одно совершенно очевидно: чтобы донести свои мысли до других, их нужно облечь в какую-то форму. В роли этой формы и выступает язык. Не будем судить, в какой мере это справедливо для естественных языков, но в программировании дело обстоит именно так.

Программирование — это умение мыслить, язык — средство, позволяющее записать эти мысли и сделать их понятными другим людям, а иногда и ком-

пьютеру. Если мыслей нет, то и записывать нечего. Поэтому наша цель — изучение именно программирования, а не языка, а язык мы будем использовать как инструмент, как средство, необходимое для достижения этой цели.

Кроме Программирования (П) и Языка Программирования (ЯП), можно выделить еще один уровень: Реализация Языка Программирования (РЯП). Реализация — это компьютерная программа (или набор программ), которая позволяет нам не просто писать программы на каком-то языке, но и исполнять их на компьютере. В реализации обычно уточняются конкретные детали языка, вводятся дополнительные возможности и ограничения.

Очень полезно с самого начала понимать, на каком уровне (П, ЯП или РЯП) мы находимся в каждый конкретный момент при обсуждении того или иного вопроса. Когда речь идет об общих принципах, о методах решения задач (уровень П), неважно, на каком языке записываются иллюстрирующие примеры, главное — разобраться в алгоритмической сути. Когда обсуждаются принципиальные свойства языка (уровень ЯП), несущественны специфические особенности конкретных реализаций.

В курсе “Буки программирования” основное изложение будет идти на уровне П. Иногда мы будем спускаться на уровень ЯП, чтобы разобраться, как лучше воплотить изученные методы в том или ином языке.

## Выбираем язык

Прочитав предыдущие рассуждения, вы, возможно, скажете: “Все это, конечно, хорошо, но ведь программировать без языка нельзя! Что-то все равно надо выбрать!”

Наиболее подходящими для первоначального обучения можно считать языки КуМир и Паскаль. Именно этими языками мы будем чаще всего пользоваться в методических материалах.

**КуМир** значительно лучше проработан методически, его использование позволяет с самого начала формировать правильные понятия и при этом не отвлекаться на технические детали. Недостатков у КуМира почти нет, если не считать стойкого психологического предрешения многих учителей и программистов.

**Паскаль** более традиционен. Для этого языка легче достать (то есть скопировать нелегальный) транслятор, о Паскале есть много книг, принято считать, что этот язык почти идеально подходит для начального обучения. Должен признаться, что с последним утверждением я решительно не согласен. Паскаль — не идеал, а всего лишь меньшее зло по сравнению с Бейсиком и Си. Впрочем, здесь мы эту тему развивать не будем.

В принципе можно использовать и любой другой процедурный язык, но желательно заранее знать те проблемы, с которыми придется столкнуться.

**Бейсик.** Классический Бейсик (с номерами строк, без полного набора алгоритмических конструкций) без крайней необходимости лучше не использовать. Сейчас, когда уходят со сцены маломощные машины типа БК, УКНЦ, Ямахи, классический Бейсик наконец-то стал отмирать, и я бы не стал грустить по этому поводу. Его использование очень и очень нежелательно.

Современный структурный Бейсик (например, QuickBasic) использовать можно. Он унаследовал некоторые родовые недостатки Бейсика (например, необязательность описаний), но в целом это вроде бы нормальный язык.

А вот применять Visual Basic и его производные я бы не рекомендовал. В этой системе слишком много “красивостей”, отвлекающих от собственно программирования и мешающих понять суть дела.

**Си.** Очень хороший язык для профессионального программирования, но вряд ли пригодный для первоначального обучения. Достоинства языка, с точки зрения профессионала-программиста, оборачиваются недостатками для педагога. Предельная лаконичность языка позволяет писать очень красиво, но, чтобы оценить эту красоту, надо понимать суть дела, а лаконичная форма мешает непосвященным разобраться в этой самой сути. Одна из главных проблем Си — указатели. Программировать на Си без использования указателей очень сложно (нельзя, например, изменить в процедуре значение параметра), а изучать с начинающими указатели — дело весьма благодарное.

**Лого.** Замечательный язык, только надо помнить, что Сеймур Пейперт придумал его вовсе не для обучения программированию. Для наших целей этот язык совсем не подходит уже хотя бы потому, что он непроцедурный.

## Основные понятия процедурного программирования

Компьютер — это информационная машина. Процессор и память — две основные части любого компьютера — соответствуют двум главным составляющим информационного процесса: хранению (представлению) информации и ее преобразованию (обработке).

Любая программа описывает некоторый информационный процесс. Следовательно, программисту необходимы средства для описания информации и ее обработки. В классических процедурных языках информация представляется в программе в виде переменных, а ее обработка — в виде команд.

Реальная эффективность применения компьютеров основана на их способности быстро и безошибочно многократно выполнять одни и те же действия с большими объемами однородной информации. Для этого необходимо иметь средства описания однородной информации и повторяющихся действий. В процедурных языках это циклы и массивы.

Большие задачи сложно решать целиком. Человек не в состоянии уследить одновременно за множеством деталей и тонкостей, без которых не обходится программирование. Следовательно, нужны средства, позволяющие разбить задачу на независимые части, с которыми можно будет работать по отдельности. В классических процедурных языках имеются процедуры — средство разбиения алгоритма, то есть процесса, но нет средства для подобного разбиения информации. В языках следующего поколения такая возможность появилась — это объекты.

Таким образом, мы перечислили минимальный набор основных понятий, связанных с процедурным программированием. Более наглядно их можно увидеть на схеме.



Теперь рассмотрим некоторые из этих понятий чуть подробнее.

### Переменная

Использование переменных — основной способ представления информации в процедурном программировании. Любые сложные структуры данных, массивы, указатели, объекты можно рассматривать как разновидности переменных. В таблице перечислены основные свойства переменных.

Свойство	Для человека	Для компьютера
Имя	Обозначение в программе	Место в памяти
Тип	Допустимые значения и операции	Способ интерпретации содержимого памяти
Вид	Информационная роль	Время жизни
Значение	Информационное содержание	Код в памяти

Рассмотрим эти свойства более подробно.

**Имя переменной.** Для человека имя переменной — это ее обозначение в программе. Как правило, в языках программирования разрешается использовать в качестве имен любые комбинации букв и цифр, начинающиеся с буквы. Формально выбор имени не имеет значения, но каждому программисту можно посоветовать помнить слова капитана Врунгеля: “Как вы яхту назовете, так она и поплывет”. Удачное имя переменной, соответствующее ее сути, способно значительно облегчить понимание программы.

Для компьютера имя — это место в памяти. Каждой переменной отводится определенное место, и в машинной программе, которая получается после трансляции текста на исходном языке, имена заменяются на обращения к соответствующему адресу.

Некоторые языки (в частности, КуМир, Паскаль, Си) требуют, чтобы каждая переменная обязательно была описана до использования. Это позволяет транслятору заранее распределить память и более строго контролировать допустимость выполняемых операций.

Другие языки (например, Бейсик или Перл) позволяют использовать переменные без специальных предварительных описаний.

Начинающим программистам важно научиться представлять информацию с помощью переменных, строить структуры данных. Поэтому на начальном этапе очень полезно использовать языки с обязательными описаниями. Еще одно их достоинство — меньшее количество возможных ошибок в программе: если оказалось, что имя записано неверно из-за случайной опечатки, в языке без обязательных описаний ошибка не будет замечена транслятором — он решит, что здесь используется новая переменная.

**Тип переменной** определяет множество значений, которые может принимать переменная, и множество операций, в которых она может участвовать. Например, значениями переменных целого типа (такой тип определен во многих языках программирования) могут быть целые числа из некоторого диапазона (конкретные границы диапазона, как правило, определяются на уровне реализации языка), для них определе-

ны арифметические операции (сложение, вычитание, умножение, деление, нахождение остатка) и операции сравнения.

Для компьютера тип — это способ интерпретации содержимого памяти. В современных компьютерах для представления информации используется двоичное кодирование. При этом для кодирования различной информации используются разные принципы, но полученный в итоге двоичный код сам по себе не содержит никаких признаков, которые позволили бы определить его природу и отличить, например, числа от текста. Указывая в программе тип переменной, мы сообщаем компьютеру, как следует преобразовывать информацию в соответствующем месте памяти.

Понятие типа есть практически во всех процедурных языках. При этом можно выделить два принципиально разных подхода: статические и динамические типы.

В языках со статическими типами тип указывается при описании переменной, он связан с ее именем. Если, например, переменная описана как целая, то в дальнейшем ей нельзя присвоить вещественное или символьное значение. К языкам со статическими типами относятся Паскаль, КуМир, Си, Бейсик.

В языках с динамическими типами тип связывается не с именем, а со значением переменной. В этом случае любой переменной можно присвоить значение любого типа, но текущее значение ограничивает множество допустимых операций. Динамические типы используются в таких языках, как Лого, Перл, JavaScript.

Начинающим лучше использовать языки со статическими типами. Это приучает к дисциплине программирования и помогает лучше понять принципы построения структур данных.

**Вид переменной** — это информационная роль, которую переменная играет в программе. Чаще всего встречаются такие виды, как локальные и глобальные переменные, входные и выходные параметры.

Для компьютера вид — это время жизни переменной, он определяет, в какой момент времени будет выделена память для переменной и когда эту память можно освободить.

Сам термин “вид” нельзя считать устоявшимся. Некоторые авторы вообще не рассматривают такое свойство переменных, ограничиваясь типом. Иногда при рассмотрении языков программирования обращают внимание на класс памяти (это понятие обычно связывают с языком Си) и область видимости. Однако эти понятия относятся прежде всего к тонкостям конкретных языков и их реализаций, почти не затрагивая информационной сущности.

При изучении основ программирования методически очень важно именно понятие вида. Необходимо с самого начала научиться рассматривать каждую, даже самую маленькую, программу как процесс информа-

ционного преобразования, уметь выделять в ней входную, выходную и промежуточную (внутреннюю) информацию.

К сожалению, языки программирования, как правило, не содержат средств для явного описания вида. В маленьких программах, построенных по схеме “ввод исходных данных — обработка — вывод результата” переменные отличаются только по использованию (одни вводятся, другие выводятся), но на их описаниях это никак не сказывается.

Как исключение нельзя не отметить КуМир. Этот язык, очень тщательно проработанный именно с методической точки зрения, и здесь оказался на высоте. Даже в самой простой программе на КуМире можно выделить аргументы и результаты, то есть отделить входную информацию от выходной на уровне описаний. Это позволяет четко разделить информационную постановку задачи от ее алгоритмического решения. В программе на КуМире заголовок алгоритма описывает условие задачи, а тело — решение. Для лучшего понимания информационной роли заголовка можно посоветовать в качестве упражнений составлять заголовки алгоритмов, не программируя собственно решений, в том числе для задач, где придумать решение может быть довольно трудно.

## Цикл

Языки программирования обычно предоставляют возможность использования нескольких видов цикла. Чаще всего встречаются циклы с предусловием, с постусловием и со счетчиком.

В цикле с предусловием (он же цикл ПОКА) некоторое условие проверяется перед каждым исполнением тела цикла. Это конструкции `нц пока` — в КуМире, `while` — в Паскале и Си.

В цикле с постусловием (цикл ДО) условие проверяется после выполнения тела. Тем самым гарантируется, что хотя бы один раз тело будет выполнено. В КуМире этот цикл реализуется конструкцией `нц ... кц_при`, в Паскале — `repeat ... until`, в Си — `do ... while`.

В цикле со счетчиком (цикл ДЛЯ) повторением управляет переменная, принимающая новое значение при каждом исполнении тела цикла. Это `нц для` в КуМире, `for` в Паскале. В Си тоже есть цикл `for`, который часто используется для моделирования цикла со счетчиком, но на самом деле `for` в Си — это просто особая форма цикла с предусловием.

На самом деле все это разнообразие форм кажущееся. Конечно, в практическом программировании удобно иметь разные виды цикла и пользоваться каждый раз наиболее подходящим. Однако фундаментальная суть цикла остается одной и той же, и наиболее четко она выражена в цикле ПОКА. Не случайно именно этот цикл устроен практически одинаково во всех

языках программирования, именно он вошел в основную теорему структурного программирования, утверждающую, что любой алгоритм можно записать, используя три основные структуры: последовательность, ветвление и цикл.

Цикл ПОКА состоит из заголовка-условия и тела, включающего произвольные команды. Конкретный синтаксис в различных языках может отличаться, но общая суть не меняется.

Рассмотрим основные свойства цикла ПОКА.

1. Перед выполнением тела цикла условие всегда истинно.

Это свойство очевидно: ведь если условие не выполняется, происходит завершение цикла. Однако, несмотря на очевидность, часто приходится видеть примерно такие фрагменты:

```
нц пока X>Y
    если X>Y ...
```

Формально никакой ошибки здесь нет, но подобное дублирование условия свидетельствует, что автор программы не до конца разобрался в механизме работы цикла или в логике решаемой задачи.

2. Сразу после окончания цикла условие всегда ложно. Это свойство тоже очевидно, оно следует непосредственно из механизма работы цикла ПОКА.

Осознанное применение этого свойства часто помогает правильно строить циклы в сложных случаях. Бывает, например, что легко понять, при каком условии цикл надо заканчивать, а условие продолжения, которое надо записать в заголовок, не столь очевидно. В этой ситуации можно просто внести в заголовок отрицание условия окончания. Благодаря свойству 2 после окончания цикла условие окончания будет заведомо верно.

Свойства 1 и 2 очень часто используются при доказательстве правильности программ.

3. Тело цикла может не исполниться ни разу.

Это возможно, если условие при первой же проверке оказалось неверным. Важно понимать, что такая ситуация не является ошибочной — это всего лишь один из вариантов нормального исполнения цикла.

4. В середине цикла условие не проверяется.

Это свойство особенно важно для начинающих программистов. У некоторых школьников складывается неверное представление, будто у цикла существует какой-то волшебный сторож, который ловит момент, когда условие нарушается, и тут же прекращает работу цикла. Очень важно с самого начала разобраться в действительной схеме выполнения.

# Алан Тьюринг

Иван Долмачев

Продолжение. Начало в № 40/99

## ГЛАВА III

*Сущность математики лежит в ее свободе.*

Г. Кантор

В предыдущей главе (см.: “Информатика”, № 40/99) мы остановились на том, что Алан Тьюринг взялся за решение одной математической проблемы, поставленной знаменитым Давидом Гильбертом. Чтобы понять, почему эта проблема была такой важной, необходимо достаточно подробно рассказать о самых значительных событиях в истории математики того времени. Вы познакомитесь с увлекательной и удивительной историей того, как “абстрактные” проблемы “чистой” математики (относящиеся даже не только к собственно математике, но и к логике и философии) привели к теоретической конструкции универсального вычислительного устройства и доказательству существования алгоритмически неразрешимых проблем.

Эту историю я начну за несколько десятков лет до рождения Тьюринга — с 70-х годов прошлого столетия, когда профессор университета немецкого города Галле, уроженец Петербурга Георг Фердинанд Людвиг Филипп Кантор заинтересовался определенными математическими задачами, которыми до него не занимался никто.

“Вряд ли можно назвать какую-либо возникшую в последние четверти прошлого столетия математическую дисциплину, которая оказала бы большее влияние на последующий прогресс всей математики и шире — на математическое мышление в целом, чем теория множеств. К идеям теории множеств в разное время подходили с разных сторон многие ученые, но оформление ее в самостоятельную науку, со своим особым предметом и методами исследования, осуществил на протяжении четверти века в работах 1872—1897 гг. Георг Кантор”. Этими словами начинается редакторское предисловие А.Н. Колмо-



Георг Кантор  
(1845—1918)

рова и А.П. Юшкевича к книге [1]. Георг Кантор первым вступил в одну из самых абстрактных областей математики, именно он впервые исследовал с математической (а не метафизической) точки зрения взгляд на бесконечность — понятие, волновавшее человеческий ум на протяжении многих веков. Теория множеств Кантора до сих пор остается фундаментом для классических разделов математики, таких, как алгебра, геометрия и анализ.

Судьба научных идей Кантора оказалась очень непростой. У них были как преданные защитники, такие, как Гильберт, сказавший однажды: “Никто не сможет нас изгнать из рая, созданного для нас Кантором!” [7], — так и непримиримые противники, среди которых был, например, авторитетный математик Леопольд Кронекер (1823—1891), отозвавшийся как-то о работах Кантора с долей презрения: “Это не математика, это теология!”.

Эта глава почти целиком посвящена подробному описанию одной из самых оригинальных идей Кантора — диагональному процессу. Диагональный процесс, изобретенный Кантором в связи с доказательством существования трансцендентных чисел и понятием несчетных множеств, лежит, как мы увидим в дальнейших главах, в основе знаменитого парадокса Рассела, повергнувшего, по словам Гильберта, математический мир в состояние шока, в основе доказательства легендарной теоремы Гёделя о неполноте формальных исчислений и в основе тьюринговского доказательства алгоритмической неразрешимости некоторых проблем (и на самом деле в основе очень многих других фундаментальных логико-математических конструкций).

Глава большей частью написана относительно “математическим” языком и, пожалуй, требует для понимания некоторого напряжения. Но для “обрисовки исторической эволюции идей”, как я писал в предисловии, сделать подобное “математическое отступление” (которое не будет единственным) представляется совершенно необходимым.

### КАНТОРОВСКИЙ ДИАГОНАЛЬНЫЙ ПРОЦЕСС

Пусть имеется квадратная таблица размера  $n \times n$ , каждая строка и каждый столбец которой занумерованы числами от 1 до  $n$ . Известно, что в каждой строке таблицы нарисован один из двух символов — крестик или нолик, причем расстановка крестиков и ноликов в таблице совершенно произвольна. На рис. 1а показана одна из таких возможных таблиц размера  $5 \times 5$ .

	1	2	3	4	5
1	0	X	X	0	0
2	0	X	X	X	0
3	X	0	X	0	0
4	0	X	0	0	X
5	X	X	X	X	X

а

	1	2	3	4	5
	X	0	0	X	0

б

Рис. 1

Рассмотрим следующую задачу. Требуется написать строку из  $n$  символов (каждый из этих символов должен быть крестиком или ноликом) так, чтобы она не совпала целиком ни с одной строкой нашей таблицы. Например, для таблицы на рис. 1а в качестве ответа можно взять строку Х0000 или строку, состоящую из одних ноликов. А, скажем, строка Х0Х00 в качестве ответа не подойдет: она входит в таблицу (это третья строка).

Как можно решить эту задачу в общем случае, т.е. как описать какой-нибудь общий способ нахождения строки, не входящей в заданную квадратную таблицу? Для читателей “Информатики”, наверное, более понятной будет такая “программистская” формулировка этой задачи. Дан массив-таблица  $T$  (двухмерный массив размера  $n \times n$ ), заполненный крестиками и ноликами. Требуется заполнить массив-строку  $D$  (одномерный массив длины  $n$ ) крестиками и ноликами так, чтобы  $D$  не совпадал целиком ни с одной из строк массива-таблицы  $T$ . (Вместо крестиков и ноликов можно, конечно, взять числа 1 и 0.)

Самый простой способ решения — это “тупой перебор”: будем по очереди выписывать (генерировать) всевозможные строки из крестиков и ноликов, сравнивать их с каждой из строк заданной таблицы  $T$  и повторять это до тех пор, пока мы не найдем строку, не входящую в таблицу. Такая строка заведомо в конце концов найдется: ведь всевозможных строк длины  $n$ , составленных из крестиков и ноликов, имеется  $2^n$  штук, а в таблице имеется только  $n$  строк, причем некоторые из строк таблицы могут даже быть одинаковыми. Поэтому существует по крайней мере  $2^n - n$  различных строк, не входящих в таблицу. (Например, в случае таблицы  $5 \times 5$  этих строк будет  $2^5 - 5 = 32 - 5 = 27$ .) Любую из них можно взять в качестве ответа.

Способ решения “тупым перебором”, конечно, не представляет собой ничего особенно примечательного (не говоря уж о том, что он кажется неоправданно трудоемким для того, чтобы найти всего лишь одну какую-нибудь строку, не входящую в таблицу).

Существует другой, гораздо более экономный и остроумный способ решения задачи. Вернемся к рис. 1а и посмотрим на элементы таблицы, расположенные по диагонали (на рис. 1а эти “диагональные элементы” обведены рамками). В таблице на рис. 1а по диагонали стоит ОХХ0Х, т.е. эта последовательность — “диагональ” нашей таблицы. Преобразуем “диагональ”, заменяя в ней каждый крестик на нолик и наоборот. Тогда из “диагонали” ОХХ0Х получится Х00Х0 (см. рис. 1б). Полученная строка не будет целиком совпадать ни с одной строкой таблицы! Легко догадаться, почему. Если бы она совпадала с первой строкой таблицы, то ее первый элемент совпадал бы с первым элементом “диагонали” (ноликом), если бы она совпадала со второй — то ее второй элемент совпадал бы со вторым элементом “диагонали”, и т.д. Но мы изготовили строку Х00Х0 так, что ее элементы как раз не совпадают с соответствующими элементами диагонали! Значит, наша строка не может совпадать ни с первой, ни со второй и вообще ни с какой строкой таблицы.

Наше рассуждение можно оформить более “математически”, если воспользоваться обозначениями индексов (номеров) элементов таблицы и элементов строки. Пусть  $T$  — исходная таблица (двухмерный массив) и  $T[i, j]$  обозначает ее элемент, находящийся в строке номер  $i$  и столбце номер  $j$ . (Например,  $T[3, 2]$  — нолик, а  $T[2, 3]$  — крестик в таблице на рис. 1а.) У “диагональных элементов” номера их строки и столбца одинаковые, т.е. на диагонали стоят элементы  $T[1, 1]$ ,  $T[2, 2]$ ,  $T[3, 3]$ , ..., иначе говоря, “диагональ” образована всевозможными элементами вида  $T[i, i]$ .

Как мы изготовили строку (одномерный массив)  $D$ , не совпадающую ни с одной из строк таблицы  $T$ ? Принцип построения состоял в требовании, чтобы каждый элемент  $D[i]$  (т.е. элемент строки  $D$ , находящийся в позиции номер  $i$ ) не совпадал с соответствующим “диагональным элементом”  $T[i, i]$ . Это можно выразить так:

$$D[i] \text{ не равно } T[i, i] \text{ для всех индексов } i. \quad (*)$$

Отсюда сразу видно, почему строка  $D$  не может совпадать целиком ни с одной из строк таблицы  $T$ . Действительно, если бы это было не так, то  $D$  совпадала бы с одной из строк таблицы, скажем, со строкой номер  $i$ . Тогда мы бы имели

$$D[j] = T[i, j]$$

для всех индексов  $j$ , в том числе и для индекса  $j$ , равного  $i$ . Но если подставить в предыдущее равенство индекс  $j$  равным  $i$ , то получится

$$D[i] = T[i, i],$$

что противоречит требованию (\*). Значит, строка  $D$  не совпадает ни с одной из строк таблицы  $T$ .

Метод построения строки  $D$ , выраженный требованием (\*), чрезвычайно прост. Особого внимания заслуживает то обстоятельство, что для построения

“исключительной” строки  $D$  нам не понадобилось знать все элементы таблицы  $T$ . Нужна была только небольшая ее часть — “диагональ”. (Для решения с помощью “тупого перебора”, описанного ранее, понадобились бы все элементы таблицы  $T$ .)

Процесс получения “исключительной” строки  $D$  по принципу (\*) получил название *канторовского диагонального процесса*, или *диагонального метода*, *диагональной процедуры*.

Рассмотренная нами задача кажется на первый взгляд довольно надуманной и в общем-то бесполезной, хотя ее решение с помощью диагонального процесса и выглядит весьма элегантно. Оказалось, однако, что простая идея диагонального процесса, придуманная Кантором, нашла очень глубокие проявления в математике и в математической логике (об этом будет рассказано ниже).

Заметим, что диагональный процесс “работает” и в случае таблицы из крестиков и ноликов, которая мыслится как *бесконечно протяженная*. Такую таблицу (см. рис. 2) легко себе представить: она неограниченно продолжается вправо и вниз — иначе говоря, каждая ее строка бесконечна, и таких строк в таблице бесконечно много.

	1	2	3	4	5	...
1	0	X	X	0	0	...
2	0	X	X	X	0	...
3	X	0	X	0	0	...
4	0	X	0	0	X	...
5	X	X	X	X	X	...
...	...	...	...	...	...	...
	а					
	1	2	3	4	5	...
	X	0	0	X	0	...
	б					

Рис. 2

Для этой бесконечной таблицы наша первоначальная задача тоже имеет смысл: можно задаться вопросом, как построить строку (бесконечную последовательность) из крестиков и ноликов, чтобы она не совпадала целиком ни с одной из (бесконечных) строк исходной таблицы. Конечно, такая задача уже носит чисто математический характер: в ней участвует абстракция бесконечной последовательности. О каком-нибудь решении методом “тупого перебора” тут не может быть и речи. Тем не менее канторовский диагональный процесс все же дает нам метод построения *бесконечной* последовательности, не совпадающей ни с одной из строк таблицы! У бесконечной таблицы тоже есть “диагональ” (естественно, бесконечная), и мы можем последовательно, как и раньше, применять принцип (\*), заменяя в

“диагонали” крестики на нолики и наоборот, продолжая этот процесс неограниченно. Построенная таким образом последовательность  $D$  не будет совпадать ни с одной из строк таблицы  $T$  ровно по тем же причинам, что и прежде:  $D$  не может совпадать с первой строкой таблицы, потому что ее первый элемент не совпадает с первым элементом “диагонали”; не может совпадать со второй строкой, потому что ее второй элемент не совпадает со вторым элементом диагонали; и так далее для всех строк таблицы.

Наше описание диагонального процесса по существу совпадает с оригинальным изложением Кантора в его знаменитой статье 1891 г. “Об одном элементарном вопросе учения о многообразиях” [2]. (Только Кантор рассматривал не крестики и нолики, как это сделали мы, а “два каких-либо исключаяющих друг друга признака [Charaktere]”.)

Над идеей диагонального процесса полезно поразмышлять каждому любознательному уму. Если вернуться к формулировке задачи (для бесконечной таблицы), то можно подумать, что она вообще лишена смысла — ведь кажется, что для предъявления бесконечной строки  $D$ , не совпадающей ни с одной из строк заданной бесконечной таблицы, необходимо обозреть эту бесконечную таблицу целиком. Вместе с тем диагональный процесс обеспечивает метод пошагового построения бесконечной строки  $D$ , причем для выполнения каждого шага нужно знать только “конечный” кусок бесконечной диагонали.

Теперь я расскажу о том, в связи с решением какой задачи Кантор изобрел свой диагональный процесс.

## СЧЕТНОЕ И НЕСЧЕТНОЕ

Математика постоянно имеет дело с бесконечными совокупностями (множествами). Таковы, например, множество всех натуральных чисел (1, 2, 3, ...), множество всех точек на прямой, множество всех фигур на плоскости. Изображая бесконечные множества на бумаге, мы часто пишем что-нибудь в следующем роде:

$$x_1, x_2, x_3, x_4, \dots \quad (1)$$

Здесь многоточие призвано подчеркнуть, что последовательность “занумерованных иксов” неограниченно продолжается вправо. Запись типа (1) несет в себе и другой важный смысл: элементы рассматриваемого множества могут быть перенумерованы (“пересчитаны”). Это значит, что каждому элементу заданного множества можно сопоставить уникальный номер — натуральное число. При этом каждое натуральное число будет номером какого-то однозначно определенного элемента множества. Такое сопоставление номеров, как говорят, образует *пересчет* элементов данного множества. “Программистская формулировка” может быть такой: некоторое бесконечное множество допускает пересчет, если все его элементы могут быть “записаны” в “бесконечный одномерный массив”  $x[1]$ ,  $x[2]$ ,  $x[3]$ , ... так, что каждый элемент этого множества “записан” в какой-то элемент массива  $x[i]$ , причем только один раз (т.е. в разные ячейки массива “записаны” разные элементы множества). Вместо

“программистских”  $x[1]$ ,  $x[2]$ ,  $x[3]$ , ... мы пишем “математические”  $x_1, x_2, x_3, x_4, \dots$ .

Например, множество натуральных чисел, разумеется, допускает пересчет:

$x_1$	$x_2$	$x_3$	$x_4$	...
1	2	4	5	...

— то есть здесь  $x_1 = 1, x_2 = 2, x_3 = 3$  и т.д. Множество всех четных чисел также допускает пересчет:

$x_1$	$x_2$	$x_3$	$x_4$	...
2	4	6	8	...

(здесь  $x_1 = 2, x_2 = 4, x_3 = 6$  и т.д.). Допускает ли пересчет множество всех целых чисел? Это множество обычно изображается в виде последовательности, “бесконечной и вправо, и влево”:

$$\dots, -3, -2, -1, 0, 1, 2, 3, \dots, \quad (2)$$

— которая, как мы видим, не дает пересчет вида (1), поскольку последовательность (1) “бесконечна только вправо”. “Отрезать” в (2) “бесконечный хвост слева” нельзя, потому что тогда “пропадет” бесконечно много отрицательных чисел. Но легко догадаться, как нужно “переставить” целые числа, чтобы получился пересчет:

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	...
0	1	-1	2	-2	3	-3	...

Нетрудно построить и другие пересчеты множества всех целых чисел. Гораздо менее простой пример — пересчет множества всех рациональных дробей вида  $\frac{m}{n}$ , где  $m$  и  $n$  — натуральные числа. Для получения такого пересчета нужно догадаться, что все рациональные дроби могут быть помещены в таблицу с бесконечным числом строк и бесконечным числом столбцов:

	1	2	3	4	5	
1	$\frac{1}{1}$	$\frac{2}{1}$	$\frac{3}{1}$	$\frac{4}{1}$	$\frac{5}{1}$	→ ...
2	$\frac{1}{2}$	$\frac{2}{2}$	$\frac{3}{2}$	$\frac{4}{2}$	$\frac{5}{2}$	...
3	$\frac{1}{3}$	$\frac{2}{3}$	$\frac{3}{3}$	$\frac{4}{3}$	$\frac{5}{3}$	...
4	$\frac{1}{4}$	$\frac{2}{4}$	$\frac{3}{4}$	$\frac{4}{4}$	$\frac{5}{4}$	...
5	$\frac{1}{5}$	$\frac{2}{5}$	$\frac{3}{5}$	$\frac{4}{5}$	$\frac{5}{5}$	...
...	...	...	...	...	...	...

Рис. 3

В этой таблице в столбце номер  $m$  и строке номер  $n$  находится дробь  $\frac{m}{n}$ . Все элементы таблицы можно последовательно “обойти”, продвигаясь “змейкой” из левого верхнего угла и “разворачиваясь” на первой строке и первом столбце таблицы, как показано стрелками на рис. 3. Тогда по “змейке” мы обойдем все рациональные дроби, получив не что иное, как их пересчет:

$$\frac{1}{1}, \frac{2}{1}, \frac{1}{2}, \frac{3}{2}, \frac{2}{1}, \frac{1}{1}, \frac{2}{3}, \frac{3}{4}, \frac{2}{2}, \frac{1}{2}, \frac{1}{4}, \frac{2}{5}, \frac{3}{4}, \frac{2}{3}, \frac{1}{2}, \dots \quad (3)$$

Здесь мы уже не пишем для наглядности символы  $x_1, x_2, x_3, \dots$  сверху, как это делали раньше. Идея “обхода змейкой” бесконечной таблицы принадлежит Кантору и находит важное применение в основах теории множеств. “Канторовская змейка” изображена на бронзовом обелиске Г. Кантору в г. Нойштадте. Кстати говоря, этот же обелиск украшен знаменитым высказыванием Кантора о математике, вынесенном в эпиграф к этой главе (по-немецки “Das Wesen der Mathematik liegt in ihrer Freiheit”).

Но вернемся к пересчету (3).

Если из последовательности (3) вычеркнуть все сократимые дроби ( $\frac{2}{2}, \frac{2}{4}, \frac{3}{3}$  и т.д.) и заново перенумеровать оставшиеся, то мы получим пересчет множества (положительных) рациональных чисел:

$$\frac{1}{1}, \frac{2}{1}, \frac{1}{2}, \frac{3}{1}, \frac{4}{1}, \frac{3}{2}, \frac{2}{3}, \frac{1}{4}, \frac{1}{5}, \frac{5}{1}, \dots \quad (4)$$

Вычеркнуть сократимые дроби нужно было потому, что каждое рациональное число должно входить в пересчет лишь один раз, а в (3) это не имело места: например, число  $1 = \frac{1}{1} = \frac{2}{2} = \frac{3}{3} = \dots$  входит в (4) бесконечное количество раз. Так произошло потому, что одно и то же рациональное число может быть представлено в виде дроби  $\frac{m}{n}$  различными способами (но только одним способом в виде несократимой дроби).

Множества, допускающие пересчет, получили название “счетных”. Как мы только что видели, счетными являются множество натуральных чисел, множество четных чисел, множество рациональных дробей, множество положительных рациональных чисел. Пересчет может быть устроен сравнительно сложно: например, для получения пересчета множества положительных рациональных чисел мы сначала расположили рациональные дроби в таблицу на рис. 3, обошли ее “змейкой”, получив пересчет (3), а затем вычеркнули “лишние” (сократимые) дроби, получив окончательный пересчет (4). Кантор задался вопросом: а существует ли пересчет, пусть даже и очень сложный, множества всех действительных чисел? (Напомним, что, кроме рациональных чисел, множество которых, как мы видели, допускает пересчет, существуют и иррациональные числа. Например, число  $\sqrt{2}$  — иррационально,

т.е. оно не может быть равным никакой рациональной дроби  $\frac{m}{n}$ .) Иначе этот вопрос можно сформулировать так: является ли множество всех действительных чисел (обозначаемое далее через  $\mathbf{R}$ ) счетным?

Выдающемуся математику Рихарду Дедекинду (1831—1916) вопрос о счетности множества действительных чисел, присланный ему Кантором в письме 1873 г., показался “не представляющим практического значения”. Это было вполне естественно — с такого рода вопросами математика того времени еще не сталкивалась, и до Кантора никому из математиков ничего подобного не приходило в голову. (Когда Дедекинд получил письмо Кантора, самого понятия “счетности” еще не существовало — мы пользуемся здесь современной терминологией; Кантор и Дедекинд в своей переписке пользовались другими терминами.) В ответном письме Дедекинд признается, что ответа на вопрос Кантора он не знает, советуя Кантору не уделять ему слишком много труда, и одновременно приводит доказательство счетности множества всех алгебраических чисел.

Действительное число называется алгебраическим, если оно является корнем какого-нибудь алгебраического уравнения с целыми коэффициентами. Любое рациональное число  $\frac{m}{n}$  — алгебраическое: оно является корнем уравнения

$nx - m = 0$ . Число  $\sqrt{2}$ , не являясь рациональным, также алгебраическое: оно является корнем уравнения второй степени  $x^2 - 2 = 0$ . Вообще все числа, получаемые из целых чисел с помощью арифметических действий, и операции извлечения корня — алгебраические (например,  $\frac{1 + \sqrt{2}}{2}$  или  $\sqrt[6]{23 + \frac{1 - \sqrt[4]{22}}{2\sqrt{7}}}$ ; в то же время существуют алгебраические числа, не представимые указанным способом). Числа, не являющиеся алгебраическими, называются *трансцендентными*. Первым, кто доказал (в 1844 г.), что не все числа алгебраические (т.е. что существуют трансцендентные числа), был французский математик Жозеф Лиувиль (1809—1882). В 1874 г. Шарль Эрмит (1822—1901) доказал трансцендентность неперова числа (числа  $e$  — “основания натурального логарифма”) и признался в опубликованной им статье, что он долго, но безуспешно пытался доказать трансцендентность числа  $\pi$  (“пи”). Лишь в 1882 г. Карл Линдеман (1852—1939), развивая методы Эрмита, доказал, что число  $\pi$  трансцендентно. Эта замечательная теорема Линдемана о трансцендентности числа  $\pi$  положила конец более чем двухтысячелетним поискам решения знаменитой античной задачи о “квадратуре круга”. Из трансцендентности числа  $\pi$  следует, что с помощью лишь циркуля и линейки невозможно построить квадрат, равновеликий (равный по площади) данному кругу, т.е. “квадратура круга” невозможна.

“Позвольте предложить Вам вопрос, имеющий для меня некоторый теоретический интерес, на который я, однако, не могу ответить. Возможно, Вы сумеете это сделать и любезно напишете мне. Речь идет вот о чем.

Возьмем совокупность всех целых положительных индивидов  $n$  и обозначим ее через  $(n)$ . Затем рассмотрим совокупность всех действительных положительных числовых величин и обозначим ее через  $(x)$ . Тогда возникает вопрос: можно ли сопоставить  $(n)$  с  $(x)$  так, чтобы каждому индивиду одной совокупности соответствовал один и только один индивид другой? На первый взгляд кажется, что это невозможно, так как  $(n)$  состоит из дискретных частей, тогда как  $(x)$  образует континуум. Однако это возражение ничего не дает, и как бы я ни был склонен думать, что между  $(n)$  и  $(x)$  невозможно однозначное соответствие, я тем не менее не могу найти основание для этого, хотя оно, возможно, просто и именно оно занимает меня.

Не столь ли соблазнительно было бы заключить, что  $(n)$  нельзя поставить в однозначное соответствие с совокупностью  $(p/q)$  всех рациональных чисел  $(p/q)$ ? И тем не менее нетрудно доказать, что  $(n)$  можно поставить в однозначное соответствие не только с этой совокупностью, но и с более общей совокупностью  $(a_{n_1, n_2, \dots, n_v})$ , где  $n_1, n_2, \dots, n_v$  — любое число неограниченных целых положительных индексов”.

(Г.Кантор — Р.Дедекинду, 29 ноября 1873 года)

“...Мой вопрос я поставил перед Вами потому, что он возник у меня уже несколько лет тому назад и я все время интересовался, имеет ли встреченная мною трудность субъективную природу или же она заключена в самой проблеме. Поскольку Вы говорите мне, что и Вы не в состоянии ответить на него, то я могу предположить, что верна как раз вторая возможность. Впрочем, хотел бы добавить, что я никогда не занимался этим всерьез, поскольку не видел для себя практического интереса, и я вполне разделяю Ваше мнение, когда Вы говорите, что поэтому-то данная проблема не заслуживает того, чтобы уделять ей много внимания. Тем не менее хорошо бы решить ее: если бы, например,

...Мой вопрос я поставил перед Вами потому, что он возник у меня уже несколько лет тому назад и я все время интересовался, имеет ли встреченная мною трудность субъективную природу или же она заключена в самой проблеме. Поскольку Вы говорите мне, что и Вы не в состоянии ответить на него, то я могу предположить, что верна как раз вторая возможность. Впрочем, хотел бы добавить, что я никогда не занимался этим всерьез, поскольку не видел для себя практического интереса, и я вполне разделяю Ваше мнение, когда Вы говорите, что поэтому-то данная проблема не заслуживает того, чтобы уделять ей много внимания. Тем не менее хорошо бы решить ее: если бы, например,

...Мой вопрос я поставил перед Вами потому, что он возник у меня уже несколько лет тому назад и я все время интересовался, имеет ли встреченная мною трудность субъективную природу или же она заключена в самой проблеме. Поскольку Вы говорите мне, что и Вы не в состоянии ответить на него, то я могу предположить, что верна как раз вторая возможность. Впрочем, хотел бы добавить, что я никогда не занимался этим всерьез, поскольку не видел для себя практического интереса, и я вполне разделяю Ваше мнение, когда Вы говорите, что поэтому-то данная проблема не заслуживает того, чтобы уделять ей много внимания. Тем не менее хорошо бы решить ее: если бы, например,

...Мой вопрос я поставил перед Вами потому, что он возник у меня уже несколько лет тому назад и я все время интересовался, имеет ли встреченная мною трудность субъективную природу или же она заключена в самой проблеме. Поскольку Вы говорите мне, что и Вы не в состоянии ответить на него, то я могу предположить, что верна как раз вторая возможность. Впрочем, хотел бы добавить, что я никогда не занимался этим всерьез, поскольку не видел для себя практического интереса, и я вполне разделяю Ваше мнение, когда Вы говорите, что поэтому-то данная проблема не заслуживает того, чтобы уделять ей много внимания. Тем не менее хорошо бы решить ее: если бы, например,

...Мой вопрос я поставил перед Вами потому, что он возник у меня уже несколько лет тому назад и я все время интересовался, имеет ли встреченная мною трудность субъективную природу или же она заключена в самой проблеме. Поскольку Вы говорите мне, что и Вы не в состоянии ответить на него, то я могу предположить, что верна как раз вторая возможность. Впрочем, хотел бы добавить, что я никогда не занимался этим всерьез, поскольку не видел для себя практического интереса, и я вполне разделяю Ваше мнение, когда Вы говорите, что поэтому-то данная проблема не заслуживает того, чтобы уделять ей много внимания. Тем не менее хорошо бы решить ее: если бы, например,

$$(a_{n_1, n_2, \dots, n_v}),$$

где  $n_1, n_2, \dots, n_v$  — любое число неограниченных целых положительных индексов”.

(Г.Кантор — Р.Дедекинду, 29 ноября 1873 года)

...Мой вопрос я поставил перед Вами потому, что он возник у меня уже несколько лет тому назад и я все время интересовался, имеет ли встреченная мною трудность субъективную природу или же она заключена в самой проблеме. Поскольку Вы говорите мне, что и Вы не в состоянии ответить на него, то я могу предположить, что верна как раз вторая возможность. Впрочем, хотел бы добавить, что я никогда не занимался этим всерьез, поскольку не видел для себя практического интереса, и я вполне разделяю Ваше мнение, когда Вы говорите, что поэтому-то данная проблема не заслуживает того, чтобы уделять ей много внимания. Тем не менее хорошо бы решить ее: если бы, например,

ответ был *отрицательным*, то тем самым мы бы получили новое доказательство теоремы Ливилля, в которой утверждается существование трансцендентных чисел...”

(Г.Кантор — Р.Дедекинду,  
2 декабря 1873 года)

“...С обратным курьером я ответил, что ответа на первый вопрос я не знаю, но одновременно сформулировал и полностью доказал, что даже совокупность всех алгебраических чисел может быть поставлена в соответствие указанным образом с совокупностью ( $n$ ) натуральных чисел [...]. Однако высказанное мною мнение, что первый вопрос не заслуживает того, чтобы уделять ему много труда, поскольку он не представляет никакого практического интереса, было решительно опровергнуто данным Кантором доказательством существования трансцендентных чисел”.

(Из личных записей Р.Дедекинда,  
1873 год; цит. по [1])

Но очень скоро Дедекинду пришлось признать, что он поспешил с выводами, и изменить свое мнение, причем в значительной степени благодаря его же собственному доказательству, посланному Кантору. Дело в том, что через несколько дней Кантору удалось доказать невозможность пересчета множества всех действительных чисел, т.е. он доказал, что множество  $\mathbf{R}$  *несчетно*. Отсюда Кантор очень просто вывел доказательство существования трансцендентных чисел. В самом деле, если все действительные числа были бы алгебраическими, то множество  $\mathbf{R}$  допускало бы пересчет (поскольку множество алгебраических чисел счетно, как показал Дедекинду). Но этого быть не может как раз благодаря несчетности множества  $\mathbf{R}$ .

Кантор опубликовал свои результаты в 1874 г. [3]. Доказательство Кантора вызвало восторг у Дедекинда и многих других математиков: оно было самым коротким и простым доказательством существования трансцендентных чисел (и остается таковым и поныне). Доказательство несчетности множества  $\mathbf{R}$ , опубликованное Кантором в упомянутой статье, вопреки широко распространенному мнению, не использовало диаго-



Рихард Дедекинду  
(1831—1916)

нального процесса. “Диагональный процесс” Кантор придумал гораздо позже и изложил его в докладе на научной конференции, но опубликовал его (в наиболее общей форме) лишь, как уже упоминалось, в 1891 году [2]. Суть канторовского доказательства, основанного на диагональном процессе, сводилась к следующему. Пусть дана произвольная последовательность действительных чисел, лежащих в числовом интервале от 0 до 1:

$$x_1, x_2, x_3, x_4, \dots \quad (5)$$

Докажем, что всегда существует число  $x$ , большее 0 и меньше 1, отличное от всех членов последовательности (5). (Если мы это докажем, то это будет значить, что множество действительных чисел, расположенных в интервале от 0 до 1, не допускает никакого пересчета, иначе говоря, отсюда будет следовать несчетность этого множества и тем более несчетность множества  $\mathbf{R}$ .)

Каждое из действительных чисел из интервала от 0 до 1 может быть записано в виде бесконечной десятичной дроби с нулевой целой частью. (Мы будем представлять действительные числа в виде *бесконечных* десятичных дробей. Например, число 0,25 представляется бесконечной десятичной дробью 0,250000.....) Рассмотрим представление чисел  $x_1, x_2, x_3, x_4, \dots$  в виде бесконечных десятичных дробей, расположенных друг под другом. Например, представление этих чисел может быть таким:

$$\begin{aligned} x_1 &= 0,32433\dots \\ x_2 &= 0,11239\dots \\ x_3 &= 0,61520\dots \\ x_4 &= 0,00719\dots \\ x_5 &= 0,49961\dots \\ &\dots \end{aligned}$$

Цифры после запятой у всех чисел  $x_1, x_2, x_3, \dots$  тем самым образуют таблицу из цифр, имеющую бесконечное число строк и бесконечное число столбцов. В нашем случае эта таблица (точнее, ее часть) выглядит так:

	1	2	3	4	5	...
1	3	2	4	3	3	...
2	1	1	2	3	9	...
3	6	1	5	2	0	...
4	0	0	7	1	9	...
5	4	9	9	6	1	...
...	...	...	...	...	...	...

*a*

	1	2	3	4	5	...
1	1	2	1	2	2	...

*б*

Рис. 4

Читатель, уяснивший идею диагонального процесса, уже, наверное, догадался, как завершить доказательство. Рассмотрим, как и раньше в случае с крестиками-ноликами, диагональ этой таблицы и заменим в ней цифры, например, по следующему правилу: если на диагонали стоит единица, то заменяем ее на двойку, а если на диагонали стоит цифра, отличная от единицы, то заменим ее на единицу (см. рис. 4). Мы получим бесконечную “исключительную” последовательность цифр (единиц и двоек), которая отлична от всех строк таблицы на рис. 4а. Тогда число  $x$ , записанное как десятичная дробь с нулевой целой частью и со знаками после запятой, взятыми из полученной с помощью диагонального процесса “исключительной” последовательности из единиц и двоек, будет отлично от всех чисел  $x_1, x_2, x_3, \dots$ ! (В нашем примере  $x = 0,12122\dots$ ) Таким образом, нужное нам число  $x$ , не входящее в последовательность (5), будет построено. Удивительно остроумное решение!

Как приведенный метод построения соотносится с формальными обозначениями  $T$  и  $D$ , введенными ранее? Здесь  $T[i, j]$  —  $j$ -я цифра после запятой в представлении числа  $x_i$  в виде десятичной дроби. “Массив”  $D$  представляет знаки после запятой у числа  $x$ .  $T$  и  $D$  связаны между собой соотношением (\*). Это соотношение обеспечивает условие того, что число  $x$  отлично от каждого из чисел в последовательности (5). Можно также сказать, что в качестве канторовских “двух исключаяющих друг друга признаков” (крестика и нолика — см. рис. 2) здесь взяты признаки для цифр “быть единицей” и “не быть единицей”.

Очень интересное философское обсуждение диагонального процесса в связи с понятием счетного и несчетного можно найти в заметках выдающегося философа Людвиг Витгенштейна [4].

По мнению автора этих строк, наилучшее из существующих доступных для школьников изложений основ канторовской теории множеств (в том числе и диагонального процесса) имеется в учебнике И.Р. Шафаревича [5].

В отличие от первоначального доказательства несчетности множества  $R$ , данного Кантором в статье [3], приведенное выше доказательство с помощью диагонального процесса вполне доступно большинству нематематиков. В силу этого обстоятельства диагональный процесс и его применение к доказательству несчетности множества  $R$  стали объектом бесчисленных псевдонаучных спекуляций, производство которых не

прекращается и до сих пор, — см., например [6] — одну из наиболее “свежих” (и наиболее диких) работ в этом направлении. (Остается только диву даваться, каким образом подобные “исследования” могут финансироваться грантами Российского гуманитарного научного фонда.)

Кантор еще при жизни подвергался жестоким нападкам со стороны научных оппонентов (среди которых были и авторитетнейшие математики того времени). От этих нападков он, религиозный и мнительный человек, сильно страдал. Но Кантор был уверен в своей правоте и убежденно, хотя иногда и с отчаянием, боролся за свои идеи, следуя своему знаменитому девизу — “Сущность математики лежит в ее свободе”. За свою борьбу он в конце концов поплатился душевным здоровьем (последние годы жизни Кантор провел в различных нервных лечебницах).

О том, какую бурю в мире математических идей вызвала теория множеств Кантора, я расскажу в следующей главе.

#### Библиография

1. Г.Кантор. Труды по теории множеств, под ред. А.Н. Колмогорова и А.П. Юшкевича. Серия “Классики науки”. М.: Наука, 1985.
2. G. Cantor. Über eine elementare Frage der Mannigfaltigkeitslehre — Jahr. Dt. Math. Ver. 1890/1891, Bd. 1, S. 75–78. (Рус. пер. [1], с. 170–172.)
3. G. Cantor. Über eine Eigenschaft des Inbegriffes aller reellen algebraischen Zahlen — J. reine und angew. Math, 1874, Bd. 77, S. 258–262. (Рус. пер.: Г.Кантор. Об одном свойстве совокупности всех действительных алгебраических чисел — [1], с. 2–81.)
4. Л.Витгенштейн. Замечания по основаниям математики, ч. I (1937–1938 гг.), приложение 2. В кн.: Л.Витгенштейн. Философские работы. М.: Гнозис, 1994, с. 59 и далее.
5. И.Р. Шафаревич. Избранные главы алгебры. Гл. VI–VII. Математическое образование, 1998, № 3–4, с. 2–81.
6. В.К. Петросян. Общий кризис теоретико-множественной математики и пути его преодоления. Версия 1.0. М.: Янус-К, 1997.
7. D.Hilbert. Über das Unendliche — Math. Ann. 95 (1925), S. 161–190. (Рус. пер.: Д.Гильберт. О бесконечном — в кн.: Д.Гильберт. Основания геометрии. Добавление VIII, с. 338–364.)

Продолжение следует

## ТЕМАТИЧЕСКИЙ ОБЗОР

всего, напечатанного в газете “Информатика” в 1999 году (№ 1–48), с аннотациями.  
Материалы собраны по разделам в хронологическом порядке

### РАЗРАБОТКИ УРОКОВ И ТЕМ; МАТЕРИАЛЫ ДЛЯ УРОКОВ И ФАКУЛЬТАТИВНЫХ ЗАНЯТИЙ

Языки программирования  
Как это делаю я  
Семинар  
Беседы  
Проекты  
Методика  
Внеклассная работа по информатике;  
кресворды, ребусы  
Шуточки

### ЗАДАЧИ И ЗАДАЧКИ

Наблюдатель

### ЭКЗАМЕНЫ, ЗАЧЕТЫ, ТЕСТЫ

Экзамены в вузы

### НАЧАЛЬНАЯ ШКОЛА

Компьютерные вирусы и защита от них  
НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ  
Информационные технологии в школе  
Дистанционное обучение

### ОБЩИЕ ВОПРОСЫ ПРЕПОДАВАНИЯ ИНФОРМАТИКИ

“Круглый стол”

Учебники и учебные пособия

### ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ

ВЕК ИНФОРМАТИКИ

ЮБИЛЕИ И ПАМЯТНЫЕ ДАТЫ

ИНФОРМАТИКА В ЛИЦАХ

ПРОФЕССИОНАЛЬНАЯ ПОДГОТОВКА

Куда пойти со школьного двора

ПРЕДЛАГАЮ КОЛЛЕГАМ

НАМ ПИШУТ

ЗАКЛАДКИ

ОЛИМПИАДЫ, КОНФЕРЕНЦИИ, ТУРНИРЫ, ШКОЛЫ,

КУРСЫ, СЕМИНАРЫ, ВЫСТАВКИ

НЕ ТОЛЬКО ИНФОРМАТИКА

ОФИЦИАЛЬНЫЕ И НЕОФИЦИАЛЬНЫЕ ДОКУМЕНТЫ

МНЕНИЯ

СОВРЕМЕННОЕ АППАРАТНОЕ И ПРОГРАММНОЕ

ОБЕСПЕЧЕНИЕ

ИНФОРМАЦИЯ

### ОБЗОРЫ

4. Зачем нужны сплайны;

5. Использование сплайнов в графике;

Заключение;

Литература.

Используются языки программирования Паскаль, Си и Бейсик (QBasic).

### РАЗРАБОТКИ УРОКОВ И ТЕМ; МАТЕРИАЛЫ ДЛЯ УРОКОВ И ФАКУЛЬТАТИВНЫХ ЗАНЯТИЙ

#### **В.А. Самойлов**

**Набор текста на компьютере слепым  
10-пальцевым методом**

№ 3

Предлагается простой и “экологически чистый” способ резкого улучшения навыков работы на клавиатуре персонального компьютера и пишущей машинки.

#### **В.А. Руденко, Н.В. Токарева**

**Диктант на уроке по теме  
“Графический редактор”**

№ 3

О преподавании информатики в специальной школе для детей, больных детским церебральным параличом, и, в частности, о проведении “диктанта”, помогающего (наряду с “изложением”) решить ряд возникающих перед учителем проблем.

#### **Ю.А. Соколинский**

**Сплайны — инструмент интерполяции**

№ 4

Рассказывается о сравнительно новом методе интерполяции, основанном на применении сплайн-функций, или просто сплайнов. Сплайн (*spline* — рейка, планка) — это длинная и тонкая линейка, изгибаемая которую можно провести плавную кривую через заданные точки.

Представлены разделы (сначала рассматривается классический метод интерполяции, использующий многочлены Лагранжа, сообщается, какие трудности связаны с этим методом):

Введение;

1. Интерполяционные многочлены Лагранжа;

2. Что такое сплайн и откуда он взялся;

3. Построение сплайна;

#### **Л.З. Шауцукова**

**Решение логических задач средствами алгебры  
логики**

№ 5

Существует много подходов к решению логических задач, причем наибольшее распространение получили три способа:

— средствами алгебры логики;

— табличный;

— с помощью рассуждений.

Рассказывается о первом из них (по книге: “Информатика: Учебник для 7—11-х классов общеобразовательных учебных заведений”. В 2 кн. Нальчик: Эль-Фа, 1997).

#### **Н.А. Юнерман, Н.А. Бархатова, Л.Б. Белкина,**

**В.И. Кадочникова, И.П. Токарева,**

**И.А. Шелегова**

**ЛогоМиры в Екатеринбурге: год работы**

№ 7

Обучающая программная среда ЛогоМиры разработана на базе языка Лого, который создан С.Пейпертом для “комплексного развития образного, логического и алгоритмического компонентов мышления”.

О методике использования среды ЛогоМиры для занятий с учащимися разных возрастных категорий (младших школьников и учащихся среднего звена) и для разных форм учебной работы.

#### **Ю.Л. Воронцова, О.Ю. Салова**

**PAINTBRUSH: изучаем, играя**

№ 10, 12

Авторы не ставили своей задачей просто описать приемы работы в системе Paintbrush. Главная цель данной

публикации — помочь “в легкой, игровой форме освоить азы графических технологий”.

**А.Г. Леонов, О.В. Четвергова**

### История компьютеров

№ 11

(Начало в № 35, 41, 48/98.)

Повествование о компьютерах, начиная с первых счетных устройств.

Ранее были представлены следующие разделы.

Введение. Ранняя история. От Леонардо да Винчи до наших дней (начало XX века). Терминология.

1. Первое поколение компьютеров. “Марк-1”. ENIAC. IAS: формат данных; архитектура системы.

2. Второе поколение компьютеров. Большие системы. IBM 7094. Формат данных. Архитектура системы. Ввод-вывод. (Машины второго поколения производились примерно с 1955-го по 1964 год.)

3. Третье поколение. Мини-компьютеры. Микропрограммирование. PDP-11. Параллельные вычисления. Суперкомпьютеры. Cray-1. Операционные системы. Отечественная вычислительная техника. БЭСМ-6. Эльбрус. (Обычно считают, что машины третьего поколения появились в 1965 году.)

В этом номере опубликован раздел

4. Четвертое поколение. Большие и сверхбольшие интегральные схемы. Микропроцессоры. Микрокомпьютеры. Микропроцессоры Motorola и Intel.

**Е.В. Андреева, И.Н. Фалина**

### Системы счисления и компьютерная арифметика

№ 14, 15, 16, 17, 18, 20

Разделы:

**Часть I.** Представление информации (базовый курс)

Тема 1. Системы счисления как разновидность информационных систем.

Тема 2. История систем счисления.

Тема 3. Позиционные системы счисления.

Тема 4. Двоичное кодирование информации.

Тема 5. Двоичная арифметика.

Тема 6. Перевод чисел из двоичной системы счисления в десятичную и обратно.

Тема 7. Информация и компьютер.

Тема 8. Компьютерное представление целых чисел.

Тема 9. Компьютерное представление вещественных чисел.

**Часть II.** Системы счисления

Глава 1. Позиционные системы счисления.

Глава 2. Арифметические операции в позиционных системах счисления.

Глава 3. Перевод чисел из одной позиционной системы счисления в другую.

Глава 4. Смешанные и нетрадиционные системы счисления.

Глава 5. Задачи для программирования.

**Часть III.** Компьютерная арифметика

Глава 6. Целочисленная компьютерная арифметика.

Глава 7. Вещественные числа и компьютер.

Глава 8. “Длинная” арифметика.

**А.А. Семенов, А.Г. Юдина**

### Первое знакомство с объектно-ориентированным программированием

№ 14, 15, 16

Статья адресована преподавателям информатики, которые хотели бы познакомить своих учеников с объектно-ориентированным программированием на наглядных и не очень сложных примерах. (Предлагается материал для шести занятий.)

Используется система Турбо Паскаль.

**О.В. Ефимова**

### Excel. Рабочая тетрадь. Тетрадь 3

№ 32

Материалы, посвященные одному из самых популярных сегодня программных продуктов — пакету для работы с электронными таблицами Excel for Windows фирмы Microsoft, были представлены в № 43/95 (И.И. Сильванович. “Microsoft Excel for Windows (5.0). Курс лекций”). В этой работе вниманию читателей предлагаются рабочие тетради по теме “Электронные таблицы Excel” (для “сопровождения” уроков информатики или самостоятельного изучения электронных таблиц), содержащие задания и необходимый для их выполнения теоретический материал.

Задания выполняются непосредственно за компьютером. В конце каждой темы приведены вопросы для самоконтроля, имеются контрольные листы с заданиями для самостоятельной работы, по которым выставляются итоговые оценки по теме. Каждое контрольное задание имеет два варианта.

**В первой тетради** (№ 6, 7/98) рассматривались следующие темы:

Тема 1. Окно программы Excel;

Тема 2. Ввод данных;

Тема 3. Выделение элементов электронной таблицы. Столбцы и строки;

Тема 4. Маркер заполнения. Автозаполнение;

Тема 5. Ввод формулы.

**Вторая тетрадь** (№ 34/98) содержала разделы:

Тема 6. Абсолютные и смешанные ссылки;

Тема 7. Работа с диаграммами;

Тема 8. Работа с листами рабочей книги.

**Структура третьей тетради** — такая же, как и двух первых. Представлены два раздела:

Тема 9. Организация работы со списками;

Тема 10. Обработка данных средствами Microsoft Excel.

**Л.Л. Акуленко-Босова**

### Элементы математической логики в курсе школьной информатики

№ 35, 37, 38, 39

**В.М. Нечаев**

### Электронные таблицы и базы данных

№ 36, 41, 43, 46, 47

(Серия работ, основанных на материалах занятий одноименного “дистанционного” курса, проведенного автором.)

Весьма популярными сегодня программными продуктами являются пакет для работы с электронными таблицами Excel 97 и система управления базами данных Access 97 фирмы Microsoft.

Предлагаемый курс ориентирован на школьных учителей информатики и построен так, что даже тот, кто не знаком не только с этими программами, но и вообще с какими бы то ни было электронными таблицами и системами управления базами данных, при желании сумеет с его помощью в короткий срок освоить указанные программные продукты.

**Занятие 1.** Смета на покупку товаров

Разделы: Столбец, строка, ячейка; Числа и текст; Формат числа; Выделение ячеек; Формулы в ячейках; Встроенные функции; Автосуммирование; Предварительный просмотр; Проверочные вопросы; Задания для самостоятельной работы.

**Занятие 2.** Книжная торговля

Ребятам предлагается помочь торговцу компьютерной литературой, который получает товар со склада по оптовым ценам и продает его в своем киоске по розничным, вести дела с помощью электронных таблиц.

**Занятие 3.** Метеорологические наблюдения

Ученикам дается задание обработать на компьютере данные метеорологических наблюдений, которые они сами же и вели, и кто знает, может быть, какая-то важная зако-

номерность, ускользавшая до сих пор от внимания взрослых метеорологов, откроется кому-нибудь из ребят.

**Занятие 4.** Табель успеваемости по учебному предмету  
Учет получаемых учащимися отметок можно вести с помощью компьютера; пусть ребята сами попробуют создать "автоматизированную страницу классного журнала", при этом, возможно, кто-то из них, изучив с помощью учителя механизм формирования итоговых баллов, захочет еще и немного повысить свою успеваемость...

**Занятие 5.** Модель радиоактивного распада атомных ядер  
На первый взгляд далекая от информатики тема, однако, изучая ее, можно познакомиться с массой полезных вещей, в том числе прямо связанных с информатикой, — это и случайные числа, и макросы, и построение диаграмм с помощью компьютера, и статистический анализ...

**А.Н. Котенок**

#### Дидактические материалы по информатике

№ 38

Комплект заданий для самостоятельных работ по основным темам программирования (запись выражений, линейная структура, ветвления, циклы, одномерные и двумерные массивы, символьные выражения).

Работы (всего их 20) можно использовать в качестве обучающих или контролирующих, а также при "итоговом" повторении материала.

**Я.Н. Зайдельман**

#### "Буки программирования". Материалы Роботландского университета

№ 48

Что делать после того, как Азы пройдены? Конечно же, за Азами должны следовать Буки! Но сначала надо повторить Азы...

Разделы:

Тема 1. Повторим Азы

Вступление

Производственное и теоретическое программирование

Программирование, язык, реализация

Выбираем язык

Основные понятия процедурного программирования

Вступление

Переменная

Цикл

#### ЯЗЫКИ ПРОГРАММИРОВАНИЯ

**И.А. Бабушкина, Н.А. Бушмелева, С.М. Окулов,  
С.Ю. Черных**

#### Практикум по Турбо Паскалю.

#### Конспекты занятий по информатике

Выпуск 1. № 22

Выпуск 2. № 26

Выпуск 3. № 30

Предисловие.

1. Начала Паскаля:

1.1. Первые занятия.

1.2. Циклы с условиями.

1.3. Простые типы данных.

1.4. Контрольные работы.

2. Основы Паскаля:

2.1. Процедуры и функции.

2.2. Файловый тип данных.

2.3. Регулярные типы данных.

2.4. Двухмерные массивы.

2.5. Строковый тип данных.

2.6. Множественный тип данных.

2.7. Комбинированный тип данных (записи).

2.8. Контрольные работы.

3. Методы сортировки и поиска данных:

3.1. Алгоритмы сортировки информации.

3.2. Алгоритмы поиска информации.

**А.И. Маргулев**

#### Программирование на языке Visual Basic 5

Выпуск 1. № 23

Выпуск 2. № 28

Выпуск 3. № 31

Visual Basic 5 — язык программирования высокого уровня для создания прикладных программ, работающих в системе Windows (95, 98 или NT), с "профессиональным" интерфейсом, причем он не требует углубления в системные и инструментальные тонкости, как это необходимо, например, при использовании языка C++.

Visual Basic 5 имеет много достоинств, а для начинающих программировать он представляет особый интерес...

Разделы:

Язык без лишних слов (введение);

Глава 1. Объектный подход;

Глава 2. Основы проектирования;

Глава 3. Работа с базами данных;

Глава 4. Компоненты ActiveX;

Глава 5. Visual Basic 5 и Internet.

#### КАК ЭТО ДЕЛАЮ Я

Рубрика "Как это делаю я" появилась в самые первые месяцы существования газеты "Информатика". На ее страницах преподаватели информатики могут делиться опытом решения различных технических и методических проблем, а также вопросов организации занятий.

**В.М. Нечаев**

#### Сжатый диск в Windows 95

№ 10

Очередная статья из длинной серии работ, посвященных модернизации компьютерного класса. Первая публикация (в № 48/97) была посвящена наращиванию оперативной памяти компьютера. Во второй статье (№ 2/98) обсуждались вопросы установки приводов CD-ROM, в третьей (№ 10/98) рассказывалось о том, как была смонтирована аппаратная часть локальной сети, в четвертой (№ 14, 18, 20/98) — об установке операционной системы Windows 95, в пятой (№ 33/98) — о том, как "настроить" сеть, в шестой (№ 40/98) — о проблемах доступа к ее ресурсам, в седьмой (№ 47/98) — о защите компьютера от перенастройки. Теперь, как ясно из названия публикации, речь идет о процедуре уплотнения винчестера в среде Windows 95.

**В.М. Нечаев**

#### Модернизация компьютерного класса

Выпуск 1. № 25

Выпуск 2. № 33

Представлены следующие статьи:

Наращивание оперативной памяти;

Установка привода CD-ROM на компьютер;

Монтаж локальной сети;

Тиражирование дистрибутива операционной системы;

Установка Windows 95;

Сжатый диск в Windows 95;

Настройка локальной сети;

Доступ к разделяемым ресурсам сети;

Ограничение прав пользователя.

**В.М. Нечаев****Работа в локальной сети Windows 95. Статья первая — “Профили пользователя”****№ 38**

Статья открывает новую серию, посвященную использованию локальной сети школьного кабинета в учебном процессе, причем содержание этой первой статьи имеет отношение не только к сетевому классу, но и к автономной работе отдельных, не соединенных друг с другом компьютеров (если сети нет, то соответствующие действия придется совершать за каждой ученической машиной).

Работа адресована учителю, которому надоело постоянные перенастройки рабочей среды на каждой машине в компьютерном классе, производимые любознательными учениками, и который хочет, чтобы для ученика, раз в неделю приходящего в кабинет, не надо было задавать заново привычные ему режимы работы в начале каждого занятия.

**СЕМИНАР****В.А. Петров, С.Н. Поздняков****Алгоритмы над целыми числами****№ 35**

В этой статье алгоритмы операций с целыми числами, которые должен хорошо знать каждый ученик уже в начальной школе (поразрядного сложения и вычитания, умножения “столбиком”, деления “уголком” и некоторые другие), используются для объяснения нескольких важных идей компьютерной математики.

Разделы:

Небольшое введение;

Делимость;

Алгоритм Евклида с вычитанием;

Деление с остатком;

Алгоритм Евклида с делением;

Арифметика остатков;

Задачи (условия 14 задач).

**И.В. Романовский****Наборы из нулей и единиц****№ 37**

Рассказывается о двоичной системе счисления, причем наборы нулей и единиц часто рассматриваются с необычных, а иногда и неожиданных позиций, например, говорится о том, что набор нулей и единиц можно считать точкой многомерного евклидова пространства. Даются упражнения, представлены условия пятнадцати задач (попробуйте, скажем, придумать вопрос, на который невозможно “ответить неправду”).

По материалам второго занятия в Заочной школе современного программирования, которая работает в Санкт-Петербурге. В заключение представлен “обзор некоторых решений задач” первого занятия. (Задачи подготовлены П.Г. Черкасовой.)

**А.А. Дуванов****Транслятор?.. Это очень просто!****№ 41, 42, 44, 45**

Легко ли создать транслятор, т.е. программу, “которая преобразует коды одного языка программирования или формата данных в другой”?

Автор статьи, руководитель Роботландского сетевого университета, утверждает, что решить подобную задачу способен даже школьник (особенно если ему помогут исполнители Кукарача и Корректор).

В последней части публикации рассматривается задача, предлагавшаяся на турнире юных программистов специально для того, чтобы ее никто не решил (тем не менее некоторые дети с ней справились).

**И.Л. Мионов****Секретная наука****№ 44**

Криптография (от греческих слов *kryptos* — тайный, скрытый и *grapho* — пишу) — “способ тайного письма, понятного лишь посвященным, тайнопись”. Ну а вообще это одна из самых интригующих областей современной науки, тем более что конца спирали “алгоритм шифрования — способ взлома” не видно и каждый следующий виток становится еще круче и увлекательнее. Кстати, машина, с полным правом претендующая на звание первого компьютера в мире, была построена (в годы второй мировой войны) для того, чтобы взломать немецкий шифр Enigma.

Разделы:

Секретная наука.

DES, NSA, etc.

[NSA, National Security Agency — Управление национальной безопасности (США); DES, Data Encryption Standard — алгоритм шифрования, разработанный под контролем NSA и предложенный в качестве стандарта в 1976 году].

Существует ли надежный шифр?

Как передать информацию без кодирования.

Как честно сыграть в орлянку.

Как сравнить записки, не раскрывая их содержимого.

И это все?

**Задачи** (14 задач), подготовленные И.Л. Мионовым и П.Г. Черкасовой.

(Криптографии был посвящен также большой материал в № 2/97.)

**БЕСЕДЫ****А.Г. Леонов****Современные форматы графических файлов. Вечера с Вовой****№ 17, 18, 36**

Сегодня самым большим увлечением для многих мальчишек является компьютер и все, что с ним связано. Статья базируется на содержании “вечерних бесед” ее автора с Вовой — одним из таких современных мальчишек. В основном речь идет о форматах BMP, GIF, PCX, TIF, PNG, JPG.

**ПРОЕКТЫ****Ю.А. Соколинский****Компьютер играет в реверси****№ 36**

Дети обычно любят компьютерные игры. А вот смогут ли школьники сами составить программу сравнительно простой “интеллектуальной” игры, подобной в определенном смысле шашкам и шахматам?

“Проект” для учащихся классов с углубленным изучением информатики (которым предоставляется возможность познакомиться с подходом, часто используемым при разработке больших и сложных программных комплексов).

Разделы:

Введение;

1. Правила игры в реверси;

2. Этапы разработки программы;

3. Этап 1 — программа для игры в реверси для двух партнеров (Сценарий, Организация данных, Графические и вспомогательные процедуры и функции, Процедуры нахождения и выполнения хода, Основной алгоритм);

4. Этап 2 — программа игры в реверси для человека и компьютера (Функция оценки эффективности хода, Алгоритм определения ранга, Сценарий и основной алгоритм программы 2);

Заключение.

**Ю.А. Соколинский.****Игра “Жизнь”****№ 47**

Автор убежден, что составить программу популярной игры “Жизнь”, которую придумал в конце 1960-х годов американский математик Конуэй, вполне по силам достаточно подготовленным учащимся. Требуется главным образом умение работать с циклами, массивами строк и символами. Потом, пользуясь созданной программой, можно понаблюдать за “возможностями” этой игры (предлагается девять соответствующих задач).

**МЕТОДИКА****И.Н. Фалина****Выравнивающе-развивающая методика преподавания информатики****№ 33**

Эта методика родилась в стенах школы-интерната имени А.Н. Колмогорова, на базе которой в 1988 году было создано самостоятельное подразделение МГУ имени М.В. Ломоносова — Специализированный учебно-научный центр Московского государственного университета — СУНЦ МГУ. Методика используется в СУНЦ МГУ уже на протяжении пяти лет, при этом (возможно, благодаря ей) учащиеся данного учебного заведения, как правило, выигрывают командное первенство Москвы по программированию, занимают призовые места на олимпиадах по информатике, включая российскую.

Статья адресована всем учителям информатики, а особенно тем, у кого в классе есть новые ученики (или даже один новый ученик).

**ВНЕКЛАССНАЯ РАБОТА ПО ИНФОРМАТИКЕ;  
КРОССВОРДЫ, РЕБУСЫ****Информатика после уроков****№ 8**

**Вопросы для проведения школьных конкурсов “Что? Где? Когда?”, “Брейн-ринг”, викторин**  
Материал подготовлен Д.М. Златопольским.

**В.Г. Федоринов****Кроссворды****№ 7, 8, 35****Ребусы по информатике****№ 10, 11, 12, 17, 20****Кто больше?****№ 10, 12****Кроссворд “Трапедия”****№ 12****На досуге****№ 37**

Предлагаются ребусы, кроссворды и простая игра (№ 10, 12, 37), которую можно использовать, например, на викторинах (выбирается слово — разумеется, на тему, связанную с информатикой, например, директория, архивирование, сумматор, и далее из его букв надо составить как можно больше других слов).

**Н.Л. Попова****Учебное “Домино”****№ 48**

Да, домино, но только не совсем обычное, а связанное с информатикой. Затрагиваются темы: “Теоретическая информатика”, “DOS”, “Компоненты компьютера”, “Алгоритмизация”.

**ШУТОЧКИ****Д.М. Файликов****Тривиальное решение****№ 37**

История, рассказанная ученику 7-го класса Пете Файликову его бабушкой, бывшим программистом Файликовым-старшим.

**ЗАДАЧИ И ЗАДАЧКИ****Д.М. Златопольский****Новая старая задача****№ 1**

“Рассказывают, что в Ханое стоит храм и рядом с ним — три башни (стержни). На первый стержень надеты 64 диска различного диаметра: самый большой диск — внизу, а самый маленький — сверху, больший диск не лежит на меньшем. Монахи этого храма должны были переместить все диски с первого стержня на третий, соблюдая следующие правила:

- 1) можно перемещать лишь по одному диску;
- 2) больший диск нельзя класть на меньший;
- 3) снятый диск нельзя отложить — его необходимо сразу же надеть на другой стержень”.

Эту задачу (которая получила название “Ханойские башни”) и легенду придумал математик Э.Люка в 1883 году.

Когда дисков много (64), процесс оказывается чрезвычайно долгим. Поэтому обычно решение ищут для меньшего количества дисков.

Рассматривается решение одной из разновидностей данной задачи. Представлены соответствующие программы на четырех языках программирования: школьном алгоритмическом, Паскале, Бейсике (используется QuickBasic), Си. (Программа на языке Си подготовлена редакцией.)

**С.Л. Островский, Ю.А. Соколинский****Задачи по теме “Компьютерная графика”  
Преобразования на плоскости****№ 2**

(Выпуски 1, 2, 3 — в № 30, 31, 47/98.)

Можно ли изобразить пространственный объект на плоскости, следуя только строгим, однозначным правилам (выполняя некоторый алгоритм), без всякой апелляции к художественной интуиции? Этот вопрос интересовал еще итальянских художников эпохи Возрождения. Тогда же сложились основы “учения о перспективе”, определяющего методы решения данной задачи. Общий подход здесь заключается в следующем. Выбирается плоскость, в которой будет находиться изображение объекта. Обычно ее называют “плоскостью проектирования”, или “картинной плоскостью”. Через каждую точку изображаемого объекта проводится прямая — линия проектирования. Точка пересечения этой линии и картинной плоскости и есть образ соответствующей точки пространства.

В выпуске рассматриваются два варианта проектирования: *параллельное* и *перспективное* (которые отличаются способом выбора линии проектирования), разбираются соответствующие задания. Используются языки программирования Паскаль, Си и Бейсик.

(Ранее в газете “Информатика” уже публиковался целый ряд работ, посвященных компьютерной графике, — см., например, № 35, 36/96; 24, 29/97; 9, 10, 11, 14/98.)

**Д.М. Златопольский****Кривые Гильберта и Серпинского,  
или Снова рекурсия****№ 8, 9**

В теории функций есть “любопытное” понятие — *всюду плотные кривые*. “Кривая на плоскости называется

всюду плотной в некоторой области, если она проходит через сколь угодно малую окрестность каждой точки этой области". Упрощенно можно считать, что всюду плотные кривые целиком заполняют указанную область. Математики Гильберт и Серпинский дали примеры всюду плотных кривых (примеры эти различаются, но строятся соответствующие кривые по одной и той же схеме).

В первой части статьи (№ 8) рассматривается рекурсивный алгоритм построения кривой Гильберта. Во второй части (№ 9) обсуждается алгоритм построения кривой Серпинского.

Используются четыре языка программирования: школьный алгоритмический, Паскаль, Бейсик (QuickBasic), Си.

**С.М. Окулов**

**Ресторан**

№ 10

"Классическая задача на метод динамического программирования". Используется язык Паскаль.

**С.М. Окулов**

**Разноцветные области**

№ 10

Описываются правила игры для двух игроков на прямоугольном клеточном поле. Рассматривается программа (на языке Паскаль), которая определяет минимально возможное количество ходов, за которые игра может завершиться.

**С.М. Окулов**

**Черно-белые деревья**

№ 16

Разбирается задача, связанная с преобразованием черно-белого изображения в строку десятичных чисел (такие преобразования употребляются в компьютерной графике). Используется язык программирования Паскаль.

**Д.М. Златопольский**

**Динамическое программирование**

№ 17, 18

Метод динамического программирования возник и сформировался в 1950—1953 гг. благодаря работам американского математика Ричарда Беллмана и его сотрудников. Является особым методом поиска оптимальных решений, специально приспособленным к так называемым "многошаговым", или "многоэтапным", операциям.

В статье кратко рассматриваются основные принципы динамического программирования и разбираются три задачи. Используются различные языки программирования: школьный алгоритмический, Паскаль (для первой и третьей задач), Бейсик, Си.

(В газете "Информатика" уже публиковались материалы, посвященные динамическому программированию, — см., например, работу Е.М. Кузницкого в № 15/96, которая была повторно представлена в № 32/96; материал С.М. Окулова в № 10/99.)

**Ю.А. Соколинский**

**Задачи по программированию**

№ 18, 19

Предлагаются задачи для работы в классе на темы: "Ветвления", "Массивы", "Символьные строки". Представлены решения на языках Паскаль, Бейсик (QBasic), Си.

(Предполагается, что составление программы для большинства задач займет у учащегося 20—30 минут.)

**Три пары монет. Задача олимпиады института Вейцмана**

№ 19

Логическая задача, связанная с выявлением (путем взвешивания) фальшивых монет.

**Д.М. Златопольский**

**Кривая дракона**

№ 20

Красивые линии с повторяющимся рисунком можно получать не только с помощью рекурсивных алгоритмов...

Используются четыре языка программирования: школьный алгоритмический, Паскаль, Бейсик (QuickBasic), Си.

(Программы, рисующие на экране различные фигуры, орнаменты и даже изображения растений, используя, как правило, рекурсию, были представлены в № 23/95 и 32/96; 19, 30/98.)

**В.М. Казиев**

**Информатика в этюдах**

№ 35, 37

**Д.М. Златопольский**

**Компьютерный фокус**

№ 35

Хотите на некоторое время стать фокусником? Если да, то эта статья для вас. Надо только иметь в своем распоряжении компьютер.

В основе рассматриваемого фокуса лежит игра, состоящая в отгадывании задуманного числа из некоторого интервала.

Используются четыре языка программирования: школьный алгоритмический, Паскаль, Бейсик (QuickBasic), Си. (Программа на языке Си подготовлена редакцией газеты "Информатика".)

**Л.Л. Акуленко-Босова**

**Задачи по системам счисления**

№ 41

Сумеете ли вы записать год своего рождения с помощью римских цифр? Каким наименьшим количеством гирь можно взвесить груз от 1 до 63 кг с точностью до 1 кг, помещая гири только на одну чашку весов? Когда  $3 \times 3 = 13$ , а  $22 + 44 = 110$ ?

Подборка задач по теме "Системы счисления", ориентированная на учеников 5—7-х классов. Предлагается 49 заданий.

(Ранее системам счисления были посвящены спецвыпуски — № 42/95; 7/97, а также материал в № 8/97.)

**А.Л. Брудно**

**Большая половина**

№ 46

Дан массив  $A$  из  $n$  числовых элементов, больше половины которых равны (между собой). Найти это значение за  $O(n)$  действий (т.е. за число действий, ограниченное первой степенью  $n$ ). Условие задачи и ее решение.

**Д.М. Златопольский**

**Задачи о взвешиваниях**

№ 47

Несколько задач, связанных со взвешиванием монет, то есть на тему, которая обычно вызывает большой интерес у учащихся.

НАБЛЮДАТЕЛЬ

**Кривая дракона: послесловие**

№ 20

Послесловие к опубликованной в этом же номере статье Д.М. Златопольского.

Дается описание более простого алгоритма построения кривой дракона. Сообщается о других алгоритмах, а также о том, где можно найти соответствующую информацию.

**ЭКЗАМЕНЫ, ЗАЧЕТЫ, ТЕСТЫ****ЭКЗАМЕНЫ В ВУЗЫ****В.И. Ракитин****Логические задачи на вступительных экзаменах по информатике****№ 17, 18, 19**

Успешно сдать вступительный экзамен по информатике в вуз без посещения подготовительных курсов весьма сложно. Данная публикация помогает получить представление о том, что может ждать абитуриентов. Представлены разделы: "Основы булевой алгебры", "Разбор задач" ["Таблицы истинности для заданных логических функций", "Задачи с множествами. Основы теории множеств", "Построение логического выражения по условиям, заданным в виде текста", "Построение логической функции по таблице истинности", "Преобразование логических функций (выражений)", "Текстовые логические задачи", "Построение логической функции по заданной логической схеме"].

**Итоги конкурса авторов тестовых заданий по информатике****№ 32**

В этом году по инициативе газеты "Первое сентября" впервые был проведен Всероссийский конкурс авторов тестовых заданий. Подробная информация об условиях конкурса публиковалась как в газете "Первое сентября" (5.09.98), так и в приложении "Информатика" (№ 41/98).

Сообщается о результатах конкурса, и описывается методика, с помощью которой производилась оценка. В заключение приведены примеры удачных и неудачных заданий по информатике.

**НАЧАЛЬНАЯ ШКОЛА****Информатика для малышей  
Пермская версия курса информатики  
для начальной школы****Выпуски 1, 2. № 7, 9**

Предполагается, что курс "Информатика для малышей" будет состоять из следующих модулей.

1. Информация.
2. Словари. Каталоги. Организация текста.
3. Организация информации (таблицы, диаграммы, карты и пр.).
4. Введение в логику.
5. Введение в системологию.
6. Принятие решений. Учет противоречивых требований.

Модуль 2 относительно автономен — в том смысле, что он не опирается непосредственно на материал модуля 1 ("Информация"). Это позволяет изучать его независимо, причем такое автономное изучение возможно как в рамках курса "Информатика для малышей", так и позднее, в средней школе.

**М.А. Плаксин****Словари. Каталоги. Организация текста****Выпуск 1****Книга для учителя**

"Введение", "Примерное поурочное планирование" (уроки 1—6).

**Тетрадь для ученика**

Задания и упражнения, соответствующие первым шести урокам.

**Выпуск 2****Книга для учителя**

Представлены уроки 7—10, посвященные поиску информации.

**Тетрадь для ученика**

В этом выпуске даются задания и упражнения, связанные с поиском информации.

(Сообщается, что желающие приобрести отдельные издания "Книги для учителя" и "Тетради для ученика" могут обратиться в Пермский региональный институт педагогических информационных технологий, — указаны телефон и адрес электронной почты.)

**КОМПЬЮТЕРНЫЕ ВИРУСЫ И ЗАЩИТА ОТ НИХ****Л.О. Сергеев****"Опасный" Интернет: мифы и реалии****№ 37**

Обзор реальных опасностей, которые могут подстергать "обычных" пользователей сети Интернет.

Разделы:

Введение;

Опасность № 1 — вложенные файлы;

Опасность № 2 — троянские программы;

Опасность № 3 — HTML-вирусы;

Опасность № 4 (гипотетическая) — Java-вирусы.

**НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ****А.Б. Ливчак****Что такое связь?****№ 40**

Наверное, можно достаточно быстро объяснить, что такое связь вообще. Однако для того чтобы рассказать о связи в СУБД Access (понятии, определение которого, по сути дела, нигде не приводится), требуется значительно больше времени.

Автор на конкретном примере показывает, что изучение модных программных систем тоже может способствовать знакомству с теоретическими основами дисциплины, которая называется Computer Science.

Разделы:

Введение;

Для чего нужна СУБД;

Запросы и операции над отношениями;

Связь как полуфабрикат соединения;

Связи и ограничения целостности;

Формальное определение связи;

Заключение.

**— ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ШКОЛЕ —****Л.С. Великович****Финансовый кризис и информатизация  
образовательного процесса****№ 3**

На первый взгляд понятия, указанные в заголовке данной публикации, кажутся взаимоисключающими. Однако говорить тут, как ни странно, есть о чем.

Краткий анализ одной частной проблемы "выживания" российского образования.

**А.И. Сенокосов****Виртуальная школа****№ 18, 19**

Постоянный автор газеты "Информатика" — об опыте организации "Виртуальной школы" (в самой обычной средней школе № 104 г. Екатеринбург), основанной на сетевых технологиях и свободе творчества детей, появление которой вызвало резкое повышение интереса детей к информатике.

## ДИСТАНЦИОННОЕ ОБУЧЕНИЕ

**А.А. Дуванов****Интернет для начинающих****№ 2, 5, 7, 9, 11, 12,  
спецвыпуск № 21**

(Продолжение. Начало в № 36, 40, 44, 46/98)

Книга об Интернете, а точнее, учебник. Первоначально задумывался как учебник для Роботландского сетевого университета, однако уже после подготовки первых параграфов автору стало понятно, что область применения книги гораздо шире (хотя основные ее читатели — школьники старшего и среднего возраста и учителя). Изложение ведется на двух “пользовательских” уровнях. Первый блок каждого раздела (представленный в виде диалогов в семье Куков) предназначен для школьников, второй — для учителей. “Заинтересованный” и смысленный школьник может использовать эту книгу как самоучитель, читая сначала диалоги Куков, а потом текст для учителя.

В газете “Информатика” публикуется фрагмент книги. Ранее были представлены “Вступительное слово”, раздел “От автора”, содержание первой главы (которая называется “Электронная почта”), а также параграфы:

1.1. Паутина в доме у Куков. (Введение, основные понятия);

1.2. Письма сам я на почту ношу. (Как устроена электронная почта);

1.3. Я Вам пишу. (Подготовка и отправление писем);

1.4. Жду ответа, как соловей лета. (Прием писем и ответы на них).

Теперь опубликованы следующие разделы:

1.5. Мой адрес не дом и не улица. (Работа с адресной книгой);

1.6. Мы писали, мы писали, наши пальчики устали. (Как правильно писать письма);

1.7. Письмо с фотографией. (Посылка и прием вложений) (“вложением” может быть любой файл, а значит, и текст, и картинки, и музыка, и видео, и другие программы);

1.8. Улыбки из Роботландии. (Доступ к ftp-архивам);

1.9. А где мне взять такую песню? (Поиск информации);

1.10. Что новенького? (Сетевые конференции).

**А.А. Дуванов****Роботландский сетевой университет-2000  
осуществляет предварительный набор  
студентов на 1999/2000 учебный год****№ 15**

Руководитель Роботландского сетевого университета представляет предварительные планы на 1999/2000 учебный год.

**А.А. Дуванов****Сетевая школа Роботландии. Заметки  
администратора****№ 35**

Руководитель Роботландского сетевого университета сообщает о начале четвертого учебного года в этом дистанционном учебном заведении, делится опытом организации работы и приглашает на занятия.

ОБЩИЕ ВОПРОСЫ ПРЕПОДАВАНИЯ  
ИНФОРМАТИКИ

## “КРУГЛЫЙ СТОЛ”

**А.Ю. Уваров****Чему учить на уроках информатики****№ 1**

Автор далек от того, чтобы призывать к “революционным” переменам. “Вопрос в том, на какие изменения, на

какое будущее ориентировать преподавателей информатики, куда направлять переподготовку учителей”.

**Ю.А. Шафрин****Несколько слов об информатике, “Computer  
Science”, ИТ и школьных учебниках****№ 7**

Продолжение дискуссии “О судьбах школьной информатики” (см. № 16, 21, 33/98; 1/99), которая, конечно, не потеряла актуальности.

**Ю.А. Шафрин****“Ребята, давайте жить дружно!”****№ 19**

“...Когда вы выбираете тот или иной учебник, вы должны, на мой взгляд, исходить в первую очередь из целей учебника, сопоставляя их с теми целями, которые перед курсом информатики ставите вы”.

Продолжение дискуссии “О судьбах школьной информатики” (см. также № 16, 21, 33, 44/98; 1, 7/99).

**А.И. Сенокосов****Программно-методический комплект. Прошлое и  
будущее****№ 20**

Продолжение дискуссии “О судьбах школьной информатики” (см. также № 16, 21, 33, 44/98; 1, 7, 19/99).

## УЧЕБНИКИ И УЧЕБНЫЕ ПОСОБИЯ

**А.Г. Кушниренко, Г.В. Лебедев****12 лекций о том, для чего нужен школьный курс  
информатики и как его преподавать****№ 1, 3, 5, 6, 8, 10, 11, 12, 14**

Об основных понятиях, идеях и целях школьного курса информатики “по учебнику” А.Г. Кушниренко, Г.В. Лебедева, Р.А. Свореня “Основы информатики и вычислительной техники” (М.: Просвещение, 1990, 1991, 1993, 1996). Дается также ряд практических советов и предлагаются соответствующие методические приемы.

Авторы надеются, что материал окажется полезным как для учителей и методистов, использующих указанный учебник, так и для тех, кто желает сравнить разные подходы к преподаванию школьного курса информатики или разработать свой собственный курс.

Редакция газеты “Информатика” выражает признательность издательству “Дрофа” за содействие в подготовке данной публикации.

**В.Р. Лещинер****Информатика для гуманитариев****№ 2**

Рецензия на учебное пособие для гуманитарных факультетов педагогических вузов С.А. Бешенкова, А.Г. Гейна и С.Г. Григорьева “Информатика и информационные технологии” (Екатеринбург, 1995).

**А.Г. Гейн****“Информатика” и “Земля Информатика”****А.И. Сенокосов****“Никакая часть данной книги не может быть...”****№ 3**

Две рецензии, открывающие дискуссию о новом учебнике “Информатика 6—7”, который является первой книгой из комплекта учебников для 6—11-х классов под редакцией профессора Н.В. Макаровой, подготовленных издательством “Питер”, г. Санкт-Петербург.

(Статья Н.В. Макаровой “Учебно-методический комплект в поддержку исследовательской концепции базового курса школьной информатики” была помещена в № 46/98.)

### Новый учебник “Информатика 7—9”

№ 6

О новом учебнике, подготовленном авторским коллективом (А.Г. Кушниренко, Г.В. Лебедев, Я.Н. Зайдельман) под руководством А.Г. Кушниренко.

№ 9, 14, 16

О новом комплекте учебников по информатике для 6—11-х классов под редакцией профессора Н.В. Макаровой, подготовленных издательством “Питер”, г. Санкт-Петербург.

Продолжение дискуссии, открытой в № 3/99, о новом учебнике “Информатика 6—7”.

### А.Ю. Уваров

#### Постижение теории через решение задач

№ 15

Рецензия на учебное пособие для общеобразовательной школы А.Г. Юдиной “Практикум по информатике в среде LogoWriter” (фрагмент которого публикуется в этом же номере газеты).

### А.Г. Юдина

#### Практикум по информатике в среде LogoWriter. Избранные главы

№ 15, 16

Фрагмент одноименного учебного пособия (с предисловием автора), признанного победителем конкурса школьных учебников по направлению “Задачник по информатике (практическое руководство) по основным разделам программы с теоретическим введением”. (Глава 7. Построение графиков функций; Глава 8. Моделирование движения.)

### А.И. Сенокосов

#### Информатика в “Виртуальной школе”

Пробный учебник для 7-го класса общеобразовательной школы

Выпуск 1. № 24

Выпуск 2. № 27

Выпуск 3. № 34

Автор предлагает познакомиться с весьма необычным учебником. От существующих учебников информатики его отличает следующее:

— предъявляются достаточно высокие требования к компьютерному классу;

— “направленность в будущее” (следует надеяться, что учебник не устареет в течение ближайших трех — пяти лет);

— широкий спектр затронутых проблем и средств их решения;

— успешная “апробация” на самых обычных учениках самой обычной школы.

Вдобавок предлагаемый курс создает предпосылки к тому, чтобы школьная информатика стала активно работать на другие дисциплины, изучаемые в школе.

По мнению автора, “это чуть ли не единственный учебник информатики, который можно применять так, как применяются учебники по любому другому предмету”.

Разделы:

#### Слово к учителю.

**Глава 1. Введение в информатику** [Предисловие; Информатика; Компьютер; Практикум № 1. Первый раз в компьютерном классе; Кодирование символьной информации; Кодирование графической информации. Часть 1; Видео-

карта; Кодирование графической информации. Часть 2; Кодирование графической информации. Часть 3. (Для дополнительного чтения); Телекоммуникации].

**Глава 2. Компьютерная грамотность** [Файловая система; Практикум № 2. Прогулка по файловому дереву; Электронные документы; Практикум № 3. Моя первая страничка в “Виртуальной школе”; HTML — язык описания электронных документов; Первое знакомство с “отмеченными объектами”; Работа в многооконной многозадачной системе; Практикум № 4; Компьютерная обработка графической информации. Графические редакторы; Практикум № 5. Стандартные инструменты CorelDraw!; Практикум № 6. Схема пути до дома. Часть 1; Практикум № 7. Работа с текстом; Компьютерная обработка графической информации. Знакомство с растровым редактором; Понятие о буфере обмена; Использование тега <Table> для оформления HTML-странички; Обработка фотоизображений в Adobe Photoshop].

Слияние двух изображений с помощью Adobe Photoshop. (Для дополнительного чтения);

Основные форматы графических файлов в Интернете; Практикум № 8. Создание картинка с прозрачным фоном;

Практикум № 9. Создание кадров для мультипликации; Практикум № 10. Механизм управления в файле GIF-формата;

Текстовый редактор;

Практикум № 11. Набираем любимое стихотворение; Стандартная панель инструментов и некоторые дополнительные возможности редактора Microsoft Word;

Публикация документов, подготовленных в Microsoft Word, в “Виртуальной школе”;

Заключение.

### А.Г. Кушниренко, Г.В. Лебедев, Я.Н. Зайдельман

#### Информатика 7—9.

#### Избранные главы нового учебника

Выпуск 1. № 39

Выпуск 2. № 42

Выпуск 3. № 45

Иногда говорят, что информация — это сведения, являющиеся объектом представления, хранения, переработки и передачи. Однако такое определение нельзя считать полным. А вообще легко ли объяснить, что такое информация, что такое компьютер и как компьютер обрабатывает информацию?

Разделы:

Глава 1. Информация и компьютеры;

Глава 2. Исполнители и алгоритмы

(Исполнители; Алгоритмы управления исполнителями; Вспомогательные алгоритмы; Цикл *n раз*; Цикл *пока*; Команды ветвления и контроля; Анализ и тестирование алгоритмов);

Глава 3. Алгоритмы и величины

(Исполнитель Чертежник и работа с ним; Алгоритмы с аргументами; Арифметические выражения и правила их записи).

### А.Г. Гейн, Н.А. Юнерман

#### Информатика 10—11

Выпуск 1. № 40

Выпуск 2. № 43

Выпуск 3. № 46

Учебник “для тех, кто испытывает склонность к изучению естественных наук”.

Публикуются обращения авторов к учителям информатики и следующие разделы:

Предисловие;

**Глава 1. Информационные технологии и компьютер** (Раз компьютер, два компьютер; Где взять информацию; Как получить информацию; Лабораторная работа № 1.

Поиск информации в Интернете; Что такое гипертекст; Лабораторная работа № 2. Изготовление HTML-страницы: первые шаги; Лабораторная работа № 3. Создание гипертекста; Лабораторная работа № 4. Графика на HTML-странице; Что еще можно делать в Интернете; Этика Интернета. Опасности Интернета; Комментарии для учителя).

### Глава 2. Компьютерное моделирование

(О задачах и моделях; Как устроены модели; Рождение модели; Системный подход и информационные модели; Динамические системы и “черные ящики”; Лабораторная работа № 5. Расшифровка “черного ящика”; Модель неограниченного роста; Выбираем средство информационных технологий; Лабораторная работа № 6. Неограниченный рост; Модель ограниченного роста; Лабораторная работа № 7. Ограниченный рост; Самостоятельная жизнь информационной модели; Границы адекватности модели; Лабораторная работа № 8. Поиск границ адекватности модели; Из пушки по...; Лабораторная работа № 9. Метод половинного деления; Как измерить количество информации; Комментарии для учителя).

Кстати, вы разделяете мнение, что курс информатики остается (по своему положению) одним из самых удивительных курсов в общеобразовательной школе?

## ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ

**А.Л. Брудно**

**Что такое алгоритм**

№ 41

Заметка, которая должна представлять интерес для читателей хотя бы уже потому, что мнение ее автора, известного специалиста, доктора физико-математических наук, профессора, организатора московских олимпиад по программированию, автора книг и учебных пособий, не всегда совпадает с мнением авторов школьных учебников информатики.

(Ранее обсуждению понятия *алгоритм* были посвящены статьи в № 35/97 — А.Г. Гейн. “Алгоритм” и № 33/98 — В.А. Носов. “О понятии алгоритма”).

## БЕК ИНФОРМАТИКИ

**Р.И. Подловченко**

**Очерки истории информатики в России**

№ 19

О книге с таким названием, вышедшей в конце 1998 г. в Новосибирске, редакторы-составители которой — академик РАЕН Д.А. Поспелов и доктор технических наук Я.И. Фет.

В этом же номере опубликована вступительная статья из указанной книги (написанная Д.А. Поспеловым).

**Д.А. Поспелов**

**Становление информатики в России**

№ 19

Вступительная статья к книге “Очерки истории информатики в России” (рецензия на которую, подготовленная Р.И. Подловченко, помещена в этом номере).

Согласно определению, данному в Большой советской энциклопедии (3-е изд. М.: Советская энциклопедия, 1972. Т. 10), информатика — это “дисциплина, изучающая структуру и общие свойства научной информации, а также закономерности ее создания, преобразования, передачи и использования в различных сферах человеческой деятельности”. Однако с начала 80-х годов в нашей стране (и не только в ней) информатикой стало называться совсем иное...

**Очерки истории информатики в России. Избранные статьи**

№ 29

Представлены следующие работы.

*Б.А. Трахтенброт.* Алексей Андреевич Ляпунов.

*В.А. Успенский.* Андрей Николаевич Колмогоров — великий ученый России.

*Л.В. Канторович.* Автобиография.

*А.И. Полетаев.* “Военная кибернетика”, или Фрагмент истории отечественной “лженауки” (об Игоре Андреевиче Полетаеве — ученом и популяризаторе кибернетики).

*В.М. Турский.* Андрей Петрович Ершов.

(Вступительная статья Д.А. Поспелова к книге “Очерки истории информатики в России” была опубликована в № 19/99.)

**А.Н. Колмогоров, В.А. Залгаллер**

**Леонид Витальевич Канторович (к 70-летию со дня рождения)**

№ 34

Статья о выдающемся ученом, лауреате Нобелевской премии Л.В. Канторовиче (из книги “Очерки истории информатики в России”; вступительная статья к книге была представлена в № 19/99, а материалы из нее публиковались в № 29/99).

**А.Н. Колмогоров**

**Автоматы и жизнь**

№ 44

Могут ли машины воспроизводить себе подобных? Могут ли машины испытывать эмоции? Могут ли машины сами ставить перед собой задачи, не поставленные перед ними их конструкторами? Вряд ли можно считать разумным нежелание разобраться в этих интересных вопросах.

Популярное изложение доклада, подготовленного выдающимся российским математиком для семинара научных работников и аспирантов механико-математического факультета МГУ им. М.В. Ломоносова.

(Из книги “Очерки истории информатики в России”).

**Е.В. Маркова**

**Эхо ГУЛАГа в Научном совете по кибернетике**

№ 45

О “кибернетиках-шестидесятниках”, бывших узниках ГУЛАГа (А.И. Берге, В.В. Парине, В.В. Налимове, А.А. Баландине, Е.В. Марковой — авторе этой публикации), которые принимали участие в реабилитации, становлении и развитии кибернетики у нас в стране, а также о других пострадавших в то время людях.

(Из книги “Очерки истории информатики в России”).

## ЮБИЛЕИ И ПАМЯТНЫЕ ДАТЫ

**50 лет методу Монте-Карло**

№ 15

В 1949 году в американском журнале **Journal of American Statistical Association** появилась статья под названием “Метод Монте-Карло”.

Небольшая заметка об указанном методе, появление и развитие которого связано с именами таких ученых, как Дж. фон Нейман, С.Улам, Н.Метрополис, а также Г.Кан и Э.Ферми.

**А.И. Сенокосов**

**К XX годовщине переворота**

№ 16

“Три кита, на спинах которых персональный компьютер стремительно выплыл на пик мирового прогресса и стал его символом, по очереди празднуют свое двадцатилетие”. Имеются в виду текстовый редактор WordStar (разработанный в 1978 году), первая в мире электронная таблица VisiCalc (появившаяся в 1979 году), а также

система управления базами данных dBase II (увидевшая свет в 1980 году). Отмечается, что особую роль в успехе информационных технологий сыграла программа VisiCalc (“Визикалк”, “Визуальный калькулятор”), и кратко описывается история ее создания.

**А.Г. Гейн**

**Пушкин и информатика**

**№ 20**

На первый взгляд данная заметка имеет странное название. Никакой информатики Пушкин, конечно, не знал, да и самой информатики в то время еще не существовало. И все же “гениальный Пушкин, видимо, ощущал, какая это мощная вещь — информация. И в его жизни именно информация сыграла роковую роль”.

**Выдающийся проект знаменитого англичанина**

**№ 34**

165 лет назад Чарльз Бэббидж (1792—1871) определил состав и назначение функциональных средств компьютера в своем неосуществленном проекте Аналитической машины.

**Пьер Симон Лаплас**

**№ 35**

**Не все так однозначно...**

**№ 36**

20 лет назад (в 1979 г.) появился журнал **Fuzzy Sets and Systems**, посвященный теории нечетких множеств.

**Первый персональный компьютер**

**№ 37**

В 1969 году, 30 лет назад, появилась электронная вычислительная машина “МИР-2”, в которой использовались дисплей со световым пером и память на магнитных картах. (А первая персональная ЭВМ “МИР” была создана у нас в стране в 1965 году, на десять лет раньше, чем в США.)

**IBM и IBM PC**

**№ 38**

**Pentium для ПРОфессионалов**

**№ 39**

**Маркони и Браун. А как же Попов?**

**№ 40**

**Долгожитель**

**№ 41**

45 лет назад (10 ноября 1954 года) был опубликован первый отчет, связанный с созданием языка Фортран (версия Фортран I).

**Норберт Винер и его “наука об управлении”**

**№ 42**

**Опережая IBM**

**№ 43**

35 лет назад (в 1964 году) было запущено в серийное производство семейство отечественных полупроводниковых ЭВМ “Урал”.

**Универсальный язык для бизнеса**

**№ 44**

**Рожденная в XVIII веке**

**№ 45**

**PKZIP и PKUNZIP**

**№ 46**

**Системотехника как научное направление**

**№ 47**

## ИНФОРМАТИКА В ЛИЦАХ

**Иван Долмачев**

**Алан Тьюринг**

**№ 40, 48**

С машиной Тьюринга (см., например, № 41, 42/97) сегодня в той или иной степени знаком, наверное, каждый математик, “компьютерный” инженер, программист и даже школьник, изучающий информатику. Но вот о создателе этой ценной теоретической модели Алане Мэти-соне Тьюринге у нас почти ничего не знают. Здесь ему повезло гораздо меньше, чем другим классикам в области компьютерных технологий.

Биография Тьюринга со многими отступлениями (в данной части повествования такими отступлениями являются рассказы об одном из величайших математиков Давиде Гильберте, который по экстравагантности поведения нисколько не уступал Тьюрингу, и об одном из самых оригинальных изобретений выдающегося математика Георга Кантора — диагональном процессе).

## ПРОФЕССИОНАЛЬНАЯ ПОДГОТОВКА

### КУДА ПОЙТИ СО ШКОЛЬНОГО ДВОРА

**С.М. Окулов**

**Чем руководствуются мои выпускники при выборе вуза?**

**№ 2**

Автор этой заметки (который преподает информатику и в вузе, и в физико-математическом лицее) является одним из ведущих авторов рубрик “Олимпиады” и “Языки программирования”. Его воспитанники — Антон Лапунов, Виктор Матюхин, Виталий Беров — победители международных олимпиад по информатике 1993-го, 1994-го, 1995-го, 1996 годов (сейчас они студенты МГУ).

(Рубрика “Куда пойти со школьного двора” была открыта в № 48/98 статьей А.И. Сенокосова “Профессия, или Проблема высшего образования для ваших лучших учеников”.)

## ПРЕДЛАГАЮ КОЛЛЕГАМ

**В.И. Карпенко**

**Использование мыши в программах на QBasic**

**№ 9**

О том, как “заставить” работать мышью в программах, написанных на языке QBasic.

**В.Е. Скородумов**

**Ответ молодому педагогу**

**№ 20**

Методист Международного детского компьютерного центра ИПС РАН отвечает на интересный вопрос (опубликованный в группе новостей relcom.education), который касается темы “Электронные таблицы Excel”. При этом отмечается, что данный вопрос “скорее не из области информатики, а из области общей педагогики”.

**И.И. Данилина**

**Познакомимся с экспертной системой**

**№ 37**

Экспертная система — это программа, которая решает задачи или просто помогает принимать решения в определенной области, употребляя информацию, “заложенную” в нее специалистами (экспертами). Существуют как узкоспециализированные экспертные системы, так

и “оболочки”, пользуясь которыми можно, не будучи программистом, создавать свои экспертные системы.

Автор статьи рекомендует ученикам поработать с “настоящей” экспертной системой, посмотреть, как она устроена, и создать что-то подобное самим.

Рассматривается задача, иллюстрирующая основные принципы разработки экспертных систем.

Предлагаются задания для самостоятельной работы.

### НАМ ПИШУТ

№ 9

Письмо учителя информатики Виктора Васильевича Доронина из города Новодвинска Архангельской области (в основном о трудностях, с которыми сталкиваются учителя информатики и их “подопечные” вдали от центра).

### ЗАКЛАДКИ

#### Узелки на память

№ 15

Новая рубрика с материалами о том, что полезного и интересного можно найти на различных WWW-серверах.

### ОЛИМПИАДЫ, КОНФЕРЕНЦИИ, ТУРНИРЫ, ШКОЛЫ, КУРСЫ, СЕМИНАРЫ, ВЫСТАВКИ

#### В.В. Прохоров

**Авторское решение, а более — пространное обсуждение задачи с III Всесоюзной олимпиады школьников по информатике**

№ 5

“... не правы те, кто полагает, что приближенность решения вычислительных задач — свойство, органически присущее компьютерной обработке информации”. Разбирается задача “Сумма”, для которой можно получить ответ вообще без какой бы то ни было вычислительной погрешности.

#### “Школа-99”

№ 10

#### Е.В. Андреева

**Решение задачи X международной олимпиады по информатике “Камелот”**

№ 11

**Решение задачи X международной олимпиады по информатике “Контакт”**

№ 12

“Разбор” задач олимпиады, которая проходила с 3 по 14 сентября 1998 года в Португалии. (Решения двух задач первого тура.) Используется язык Паскаль.

(Сообщение об итогах этой олимпиады опубликовано в № 38/98, а условия задач опубликованы в № 40, 43/98.)

#### Олимпиады по информатике

№ 13

Как обычно, тринадцатый номер газеты является “олимпиадным”. Представлены следующие материалы.

**В.М. Кирюхин. Всероссийская олимпиада школьников по информатике**

Рекомендации методической комиссии по проведению областных, территориальных и республиканских туров в 1999 году;

Задачи, предлагаемые методической комиссией для проведения областных, территориальных и республиканских туров (“Телефонный номер”, “Разбиение на такты”, “Стопами Дирихле”, “Умножение дроби”, “Арифметическая прогрессия”, “Хамелеон”, “Максимальный результат”); Комментарии к задачам для членов жюри.

#### Задачи X международной олимпиады по информатике

Приводятся условия и решения трех задач олимпиады (которая проходила с 3 по 14 сентября 1998 года в Португалии), а именно: “Лампы для праздника”, “Полигон”, “Картинки”. Решения подготовлены Е.В. Андреевой. Используется язык Паскаль.

(Сообщение об итогах этой олимпиады опубликовано в № 38/98, а условия задач представлены в № 40, 43/98; задачи “Камелот” и “Контакт” разбирались в № 11, 12/99.)

**В.М. Казиев. О целях, задачах и методике проведения олимпиад по информатике (на примере олимпиад КБГУ)**

Об опыте проведения олимпиад по информатике в Кабардино-Балкарском государственном университете. Разделы:

Цели и задачи олимпиады;

Методика подбора задач для проведения олимпиад;

Методика оценки заданий олимпиады;

Олимпиада завершена. Что дальше?

Примеры заданий.

Отмечается, что олимпиады (или турниры) по информационным технологиям, программированию (как и методики их проведения) имеют наряду с олимпиадами по информатике полное право на самостоятельное существование.

**Восьмая олимпиада по математике и криптографии 1998 года**

Условия и решения задач олимпиады для школьников 9—11-х классов, которая состоялась 20 декабря 1998 года.

**С.М. Окулов. Задачи 11-й Кировской областной олимпиады школьников по информатике**

Разбираются задачи: “Ребус”, “Делители”, “Троллейбусы”, “Посылки”. Употребляется язык Паскаль.

#### С.М. Окулов

**Задача “Мотоциклисты”**

№ 14

**Задача “Прямоугольники”**

№ 15

Разбираются две задачи 11-й Кировской областной олимпиады по информатике. Используется язык Паскаль. (Решения других задач этой олимпиады рассматривались в № 13/99.)

**“Комтек-99”: картинки с выставки**

№ 18

**История вычислительной техники**

№ 20

Короткий фоторепортаж о выставке в московской гимназии № 1530. Выставка была организована ребятами под руководством их учителя и постоянного автора газеты “Информатика” Д.М. Златопольского.

**Образование XXI века на ВВЦ**

№ 21

**Выставка “Информационные технологии в образовании” в ВВЦ**

№ 23

О выставке “Информационные технологии в образовании” во Всероссийском выставочном центре.

**Детектив вокруг ноутбука. Новости Российского фонда компьютерных программ**

№ 31

#### В.М. Кирюхин

**XI Всероссийская олимпиада школьников по информатике**

№ 33

Сообщается об итогах олимпиады, проходившей в Санкт-Петербурге со 2 по 8 апреля 1999 года. В частности, отмечается, что очень хорошо выступила группа

школьников, для которых эта олимпиада не первая, и в то же время около половины участников не решили полностью ни одной задачи.

На основании итогов олимпиады и результатов предварительных сборов формируется сборная команда России по информатике для участия в международной олимпиаде в Турции (2—8 октября 1999 г.).

### Результаты XI олимпиады школьников по информатике

№ 33

Таблица результатов (участники, занявшие первые 59 мест).

### Задачи XI Всероссийской олимпиады школьников по информатике

№ 33

Условия задач, которые предлагались на олимпиаде: “Круг из прямоугольников” (раздел: компьютерная геометрия), “Новый алхимик” (поиск на графах), “Песнь” (динамическое программирование), “Хлопоты многодетного купца” (сортировка и последовательности), “Микрофония” (моделирование), “Стихоплет” (математическая лингвистика).

*И.Л. Миронов, И.В. Романовский*

### Задача о круге

№ 34

В № 33/99 рассказывалось об итогах XI Всероссийской олимпиады школьников по информатике, проходившей в Санкт-Петербурге (2—8 апреля 1999 г.). Рассматривались условия всех шести задач олимпиады. В этом номере разбирается первая из них. Используется язык программирования Паскаль.

### III международная конференция “Перспективы систем информатики”

№ 36

Летом (с 6 по 9 июля) 1999 года в новосибирском Академгородке проходила конференция, посвященная памяти пионера отечественного программирования академика Андрея Петровича Ершова. Своими впечатлениями об этом мероприятии делятся его организаторы и участники (Л.В. Городняя. “Школьная информатика”; Г.Д. Шапалова. О работе секции “Школьная информатика”; И.В. Потосин. Перспективы систем информатики).

### Дистанционная обучающая олимпиада по географии (ДООГ)

№ 37

*Д.Ю. Усенков*

### ИТО-99: новости с выставки

№ 46

Краткий рассказ о новинках, которые были представлены на традиционной выставке аппаратных и программных средств, проходившей параллельно с конференцией “Информационные технологии в образовании” — ИТО-99.

*М.Е. Белиловская*

### Информационные технологии в образовании...

№ 47

Впечатления о IX международной конференции-выставке “Информационные технологии в образовании” — ИТО-99.

## НЕ ТОЛЬКО ИНФОРМАТИКА

*М.Н. Коптюг*

### Международное образование, ресурсы компьютерной сети и золотой занавес

№ 39

Чего иностранцы совсем не понимают, так это как российским учителям удается сохранить чувство юмора в нынешней ситуации. И многие хотят помочь...

## ОФИЦИАЛЬНЫЕ И НЕОФИЦИАЛЬНЫЕ ДОКУМЕНТЫ

### Материалы сборника “Оценка качества подготовки выпускников основной школы по информатике”

№ 38, 39, 41, 42, 43, 44

Материалы сборника, подготовленного А.А. Кузнецовым, Л.Е. Самовольновой и Н.Д. Угриновичем. Сборник включает:

- обязательный минимум содержания образования по информатике (варианты “А” и “Б”);
- примерную программу курса информатики;
- требования к уровню подготовки выпускников основной школы;
- образцы итоговых заданий по оценке качества подготовки выпускников основной школы по информатике (6 вариантов).

## МНЕНИЯ

### Тесты по информатике. Послесловие

*А.Г. Шмелев*

№ 44

Замечания к тестам по информатике, вошедшим в сборник “Оценка качества подготовки выпускников основной школы по информатике”.

*П.И. Совертков*

### Психологические особенности восприятия кодовой таблицы ASCII

№ 45

## СОВРЕМЕННОЕ АППАРАТНОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

*А.И. Сенокосов*

### Что там inside? Заметки о новом процессоре Pentium III

№ 17

## ИНФОРМАЦИЯ

О III Соловейчиковских чтениях, которые проводились 1 и 2 октября 1999 года в павильоне № 71 Всероссийского выставочного центра (ВВЦ)

№ 39

Конференция учителей информатики в Челябинском государственном педагогическом университете  
Региональные представители “Информатики”

№ 41

*С.Л. Островский*

### Лучший образовательный сервер!

№ 44

По мнению посетителей и участников выставки “Информационные технологии в образовании”, сервер Объединения педагогических изданий “Первое сентября” [www.1september.ru](http://www.1september.ru) признан лучшим в номинации “Образовательные серверы”.

### “Информатика” в Нижнем Тагиле

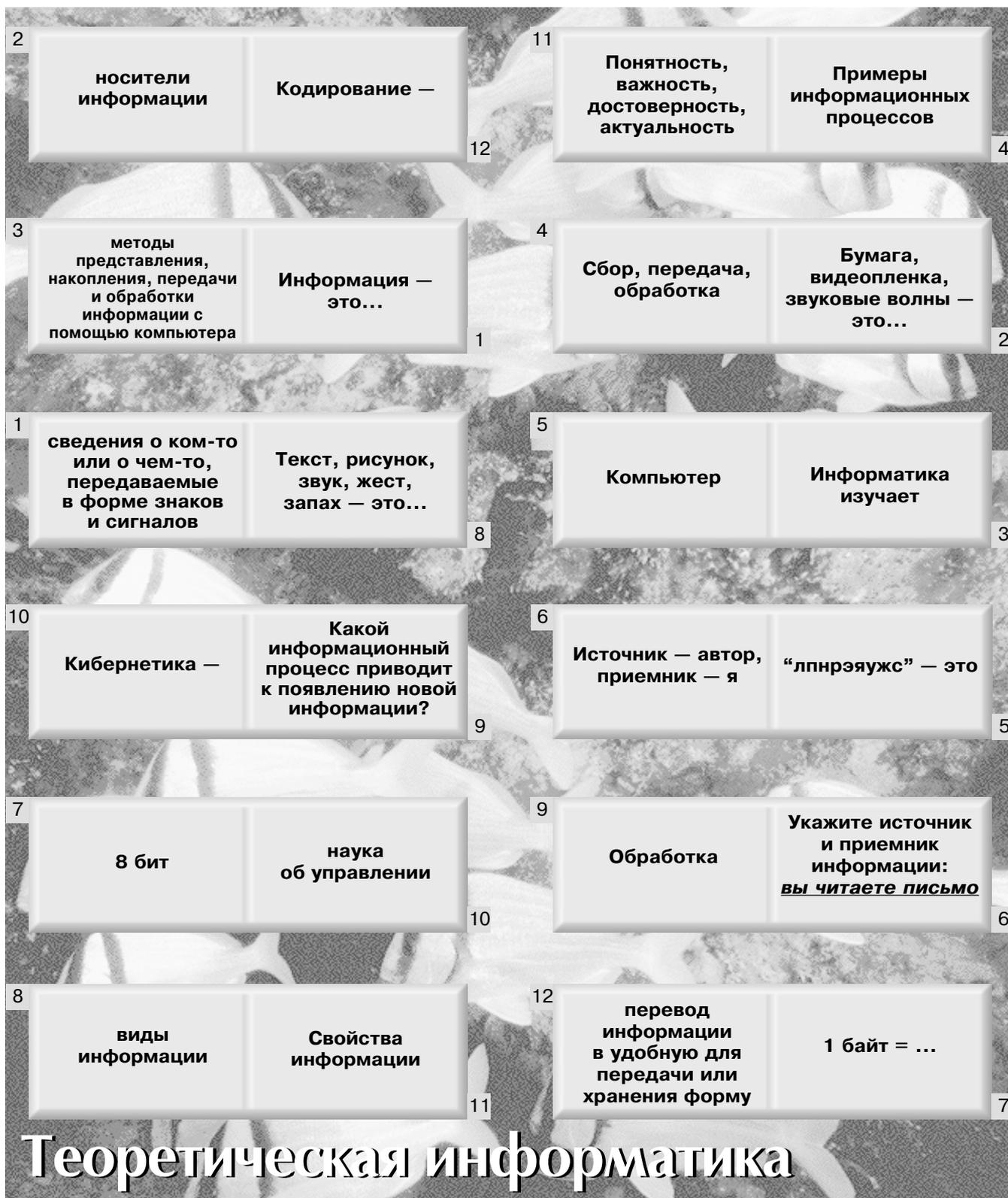
№ 44

О встрече с учителями информатики Нижнего Тагила (29 октября 1999 года), организованной региональным представителем газеты “Информатика” Е.Д. Шишкиным.

# Учебное “домино”

Н.Л. Попова

Данное домино можно использовать на различных внеклассных мероприятиях по информатике — конкурсах, викторинах, КВН. Если вы вырежете изображения костяшек домино и наклеите их на твердую основу, они прослужат не один год.



11

Группа файлов на одном носителе	Каталог — это...
---------------------------------	------------------

12

10

Имя и расширение	Каталог — это...
------------------	------------------

11

12

место на диске, в котором хранятся сведения о файлах и подкаталогах	MS-DOS — это...
---	-----------------

1

1

дисковая операционная система, разработанная фирмой Microsoft	Программы управления внешними устройствами
---	--

2

2

Драйверы	Приглашение DOS имеет вид
----------	---------------------------

3

9

Операционная система	Из чего состоит имя файла
----------------------	---------------------------

10

3

C:\> D:\> A:\>	BIOS — это...
----------------------	---------------

4

7

COMMAND.COM	Расширение исполняемого файла
-------------	-------------------------------

8

8

COM BAT EXE	Система программ, необходимая для организации работы всех задач и распределения ресурсов компьютера
-------------------	---

9

4

Базовая система ввода/вывода	Устройство хранения BIOS
------------------------------	--------------------------

5

5

ПЗУ (ROM)	Файл — это...
-----------	---------------

6

6

Поименованная область памяти	Какая программа является интерпретатором команд DOS
------------------------------	---

7

DOS

12

ПЗУ	Аппаратное обеспечение компьютера
-----	-----------------------------------

1

5

SVGA	Компьютер "рисует" на бумаге с помощью...
------	---

6

7

Жесткий диск	Компьютер "пишет" на бумаге с помощью...
--------------	--

8

11

ОЗУ	ROM
-----	-----

12

1

HARDWARE	Оптическое устройство ввода графической информации
----------	--

2

2

Сканер	Похож на телевизор
--------	--------------------

3

4

Основные цвета цветовой палитры монитора	Один из типов мониторов
--	-------------------------

5

8

Принтер	"Мозг" компьютера
---------	-------------------

9

3

Монитор	Красный, синий, зеленый
---------	-------------------------

4

9

Процессор	Емкость и быстродействие — основные характеристики
-----------	--

10

6

Плоттер	Винчестер
---------	-----------

7

10

Память	RAM
--------	-----

11

# Компоненты компьютера

12	<b>Ветвление</b>	Поименованная область памяти, предназначенная для временного хранения данных	1	3	<b>Точка ветвления алгоритма</b>	<b>Алгоритм</b>	4
1	<b>Переменная</b>	Способы задания значения переменной	2	9	<b>Линейный алгоритм</b>	Для повторения одних и тех же действий используется...	10
4	<b>Точное предписание, приводящее к выполнению поставленной цели</b>	Кто-то или что-то, способные точно исполнять команды	5	2	1. Присвоение 2. Ввод с клавиатуры	<b>Символ</b> 	3
8	Команда не может быть выполнена из-за логической ошибки	Алгоритм, в котором все действия выполняются последовательно	9	11	Задача разбивается на подзадачи, решение которых ведет к решению всей задачи	В зависимости от заданного условия выполняются те или иные действия	12
10	<b>Конструкция "Цикл"</b>	Метод последовательной детализации	11	6	Набор понятных исполнителю команд	"Не понимаю" — сообщение о том, что...	7
7	Задана команда, не входящая в СКИ	"Не могу" — сообщение о том, что...	8	5	<b>Исполнитель</b>	<b>СКИ — это...</b>	6

Издательства «ИНФРА-М», «ВЕСЬ МИР», представляют:  
**О К С Ф О Р Д С К А Я  
 И Л Л Ю С Т Р И Р О В А Н Н А Я  
 Э Н Ц И К Л О П Е Д И Я**

- **ВСЕМИРНАЯ ИСТОРИЯ**  
С древнейших времен и до 1800 года
- **ВСЕМИРНАЯ ИСТОРИЯ**  
С 1800 года и до сегодняшнего дня
- **МИР ПРИРОДЫ**
- **ФИЗИЧЕСКИЙ МИР**
- **ИСКУССТВО**
- **ИЗОБРЕТЕНИЯ**
- **И ТЕХНОЛОГИИ**
- **НАРОДЫ И КУЛЬТУРЫ**
- **ВСЕЛЕННАЯ**
- **СПРАВОЧНЫЙ**

#### Оптовые продажи:

ООО «Издательский Дом ИНФРА-М»: Россия,  
 127214, Москва, Дмитровское шоссе, 107.  
 Тел.: (095) 485-75-98, 485-58-90, 485-71-77,  
 485-76-18;

факс: (095) 485-53-18

[www.infra-m.ru](http://www.infra-m.ru)

E-mail: [books@infra-m.ru](mailto:books@infra-m.ru)

#### Розничные продажи и подписка:

Дом деловой книги: Москва,  
 ул. Марксистская, 9 (м. «Пролетарская»).

Тел.: (095) 270-52-17, 270-52-18.

Магазин «Книги»: Рязанский проспект, 97  
 (м. «Рязанский проспект»).

Тел.: (095) 371-25-11.

Центральные книжные магазины.

**ВЕСЬ  
 МИР**



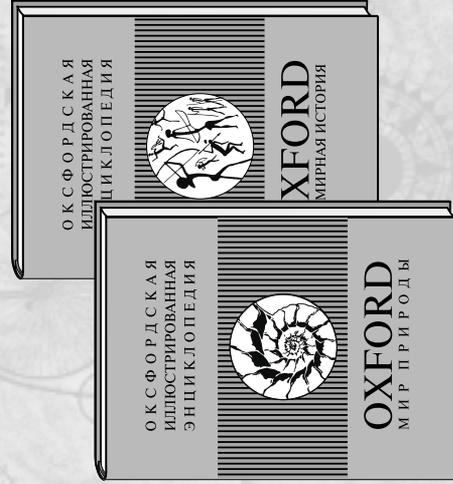
ИНФРА-М

Во всем мире книги с маркой OXFORD символизируют стандарт качественного знания. Создавая Оксфордскую иллюстрированную энциклопедию, авторы и издательство Оксфордского университета стремились к тому, чтобы статьи на самые сложные темы были понятны каждому. Читателями этой книги могут быть люди практически всех возрастов и профессий. Ее можно назвать по-настоящему популярной. Несомненно, это издание займет достойное место в любой библиотеке.

#### Том «ВСЕМИРНАЯ ИСТОРИЯ»

(С древнейших времен и до 1980 года)

и «МИР ПРИРОДЫ» уже в продаже!



Том «ИЗОБРЕТЕНИЯ  
 И ТЕХНОЛОГИИ»

В продаже с 1 января 2000 г.



# Читайте в № 1/2000:

- ❁ В рубрике “Официальные документы” публикуется комментарий главного специалиста МО РФ Л.Е. Самовольновой к комплекту нормативных документов, регламентирующих преподавание курса информатики (эти документы были опубликованы в осенних номерах нашей газеты).
- ❁ В рубрике “Уроки для учителя” мы публикуем материалы очередного занятия, посвященного изучению электронных таблиц. Как всегда, В.М. Нечаев предлагает вашему вниманию интересную задачу, которую можно решить средствами Excel.
- ❁ В рубрике “Семинар” читайте статьи И.В. Романовского о языке программирования Паскаль и Е.М. Кузнецкого о динамическом программировании. (Последняя статья уже публиковалась в “Информатике”. Мы повторяем ее в память о талантливом учителе, которого не стало 31 декабря 1996 г.)
- ❁ В рубрике “Мнения” публикуются интервью, которые наш корреспондент взял у участников осенней выставки “ИТО-99”.
- ❁ В новой рубрике “На стенд в кабинете информатики” читайте заметку о компьютерах IBM 360, сыгравших значительную роль в истории вычислительной техники.

**И это не все!**

**Читайте и выписывайте нашу газету.  
В новом году она станет еще интереснее!**

Гл. редактор  
С.Л. Островский  
Зам. гл. редактора  
Е.Б. Докшицкая  
Редакция:  
И.Н. Фалина,  
Н.Л. Беленькая,  
Н.П. Медведева  
Дизайн и компьютерная  
верстка:  
Н.И. Пронская  
Корректоры:  
Е.Л. Володина,  
С.М. Подберезина

©ИНФОРМАТИКА 1999  
выходит четыре раза в месяц  
При перепечатке ссылка  
на ИНФОРМАТИКУ обязательна,  
рукописи не возвращаются

121165, Киевская, 24  
тел. 249 4896  
Отдел рекламы  
тел. 249 9870

Учредитель: ООО “Чистые пруды”  
Регистрационный номер 012868

Отпечатано в типографии ОАО ПО “Пресса-1”.  
125865, ГСП, Москва, ул. “Правды”, 24.  
Тираж 5500 экз.  
Заказ №

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

Тел. (095)249 3138, 249 3386. Факс (095)249 3184

Internet: [inf@1september.ru](mailto:inf@1september.ru)  
Fidonet: 2:5020/69.32  
WWW: <http://www.1september.ru>

ОБЪЕДИНЕНИЕ  
ПЕДАГОГИЧЕСКИХ  
ИЗДАНИЙ  
«ПЕРВОЕ СЕНТЯБРЬ»

Первое сентября (А.С. Соловейчик), индекс подписки — 32024; Английский язык (Е.В. Громушкина), индекс подписки — 32025; Биология (Н.Г. Иванова), индекс подписки — 32026; Воскресная школа (монах Киприан (Яценко), индекс подписки — 32742; География (О.Н. Коротова), индекс подписки — 32027; Здоровье детей (А.У. Лекманов), индекс подписки — 32033; Информатика (С.Л. Островский), индекс подписки — 32291; Искусство (Н.Х. Исмаилова), индекс подписки — 32584; История (А.Ю. Головатенко), индекс подписки — 32028; Литература (Г.Г. Красухин), индекс подписки — 32029; Математика (И.Л. Соловейчик), индекс подписки — 32030; Начальная школа (М.В. Соловейчик), индекс подписки — 32031; Немецкий язык (М.Д. Бузоева), индекс подписки — 32292; Русский язык (Л.А. Гончар), индекс подписки — 32383; Спорт в школе (Н.В. Школьников), индекс подписки — 32384; Управление школой (Н.А. Широкова), индекс подписки — 32652; Физика (Н.Д. Козлова), индекс подписки — 32032; Химия (О.Г. Блохина), индекс подписки — 32034; Школьный психолог (М.Н. Сартан), индекс подписки — 32898.