

ИНФОРМАТИК



Читайте в номере

Олимпиады 2-10

XI Всероссийская олимпиада школьников по информатике проходила весной этого года в Санкт-Петербурге. Более 150 участников состязались в решении шести задач (по три в каждом туре), отобранных научным комитетом. Задачи были посвящены различным разделам информатики: компьютерная геометрия, динамическое программирование, математическая лингвистика и др. В номере представлены статья председателя жюри олимпиады В.М. Кирюхина, условия всех задач, а также таблица результатов, которые... озадачивают: из них видно, что около половины участников не решили полностью ни одной задачи! Почему? Читайте материалы этого и следующего номеров, в которых будут опубликованы решения задач олимпиады.

Методика 11-15

Каждый год первого сентября нам приходится решать одну и ту же задачу: в классе новенький. А если класс новый и в него набрали ребят из разных школ, то работать учителю информатики в нем еще труднее. Но деваться-то некуда: такой уж у нас предмет. Один из подходов, который можно использовать в "сборных" классах и классах с большим количеством новых детей, предлагает И.Н. Фалина, старший преподаватель СУНЦ МГУ. Описываемая методика основана на многолетнем практическом опыте автора и потому особенно интересна. У вас в классе тоже есть новые дети? Обязательно прочитайте эту статью!

Тематический выпуск

Модернизация компьютерного класса

О том, как "настроить" локальную сеть, о проблемах доступа к ее ресурсам, а также о том, как защитить компьютеры от перенастройки.

XI Всероссийская олимпиада школьников по информатике

В.М. Кирюхин,

кандидат технических наук, доцент,
Москва

XI Всероссийская олимпиада школьников по информатике проходила в этом году в северной столице нашей страны, Санкт-Петербурге, в период со 2 по 8 апреля. Третий год подряд Аничков дворец собрал лучших школьников в области информатики, большинство из которых — победители республиканских, областных и региональных олимпиад. Интерес к такого рода соревнованиям не ослабевает, а, наоборот, постоянно возрастает, несмотря на то, что компьютерная техника и информационная технология стали для всех привычным делом. Стремительное развитие этого направления науки и техники требует прилива свежих сил, и в этой связи важно не только освоение уже имеющихся достижений в области новых информационных технологий, но и создание собственного научного потенциала, способного обеспечить ведущую роль России в мире в следующем тысячелетии.

Несмотря на то, что в проведении олимпиады были задействованы все компьютерные резервы организаторов, принять всех желающих оказалось невозможным. Тем не менее большинство регионов России прислали своих представителей. В нынешней олимпиаде участвовали 134 школьника из 57 регионов страны, а также Москвы и Санкт-Петербурга. Впервые в олимпиаде приняли участие школьники из Байконура.

Организаторы олимпиады — Министерство общего и профессионального образования РФ, Комитет по образованию администрации Санкт-Петербурга и Санкт-петербургский городской Дворец творчества юных — сделали все возможное, чтобы смотр молодых дарований России в области информатики прошел на высоком уровне. Этому в немалой степени способствовали и спонсоры олимпиады, среди которых следует выделить старейшего спонсора олимпиад и сборной команды России по информатике — российскую компанию “ЭЛКО Технологии”.

Традиционно Всероссийская олимпиада проходит в два тура, и последняя не стала исключением. На каждом туре для решения задач участникам было отведено 5 часов. В распоряжение каждого был выделен IBM PC-совместимый компьютер в следующей конфигурации: процессор не ниже Intel 486, операционная система MS-DOS. В процессе решения задач допускалось



Аничков дворец. 1741—1818.

Архитекторы — М.Г. Земцов, Г.Д. Дмитриев, В.В. Растрелли, И.Е. Старов, К.И. Росси и др. В 1937 г. переоборудован под Дворец пионеров (архитекторы — А.И. Гегелло, Д.Л. Кричевский)

использование одной из систем программирования: Turbo Pascal v.7.0, Borland C++ v.3.1 и Quick Basic. В отведенное время требовалось решить три задачи: формализовать каждую из них, разработать по возможности лучший алгоритм решения, а также написать и отладить соответствующие программы на одном из допустимых языков программирования. Результатом решения каждой задачи являлась работоспособная программа в исполняемом виде.

Отбором и подготовкой задач для олимпиады занималось авторитетное жюри, в состав которого входили известные ученые и специалисты школьной информатики. Среди них д.ф.-м.н., профессор И.В. Романовский (Санкт-Петербург), к.т.н., доцент С.М. Окулов (г. Киров), к.ф.-м.н., доцент В.В. Прохоров (г. Екатеринбург), к.ф.-м.н., доцент Е.В. Андреева (Москва), к.ф.-м.н., доцент С.Г. Волченков (г. Ярославль), к.ф.-м.н., доцент В.Д. Лелюх, к.п.н. А.В. Алексеев (г. Красноярск), И.Л. Миронов (Санкт-Петербург), А.В. Лапунов (Москва), А.П. Овсянников и Т.В. Овсянникова (г. Троицк Московской обл.), И.А. Серегин (г. Нальчик). Возглавлял работу жюри к.т.н. доцент В.М. Кирюхин.

При выборе олимпиадных задач жюри исходило из следующих соображений. Во-первых, поскольку по статусу данная олимпиада является олимпиадой по информатике, а не по программированию, то задачи должны предоставлять школьникам возможность проявить свои способности на всех этапах решения с использованием компьютера, начиная с формализации и кончая тестированием и отладкой программы.

Во-вторых, задачи должны отражать практическую направленность информатики и привлечь внимание участников к совершенно неожиданным проявлениям информатики в реальной жизни. Более того, задачи должны стимулировать дальнейшее изучение информатики и показывать все ее многообразие.

В-третьих, сложность задач должна быть разумной. С одной стороны, при их решении не должны использоваться специальные знания, а с другой стороны, они должны предоставлять возможность большинству школьников продемонстрировать свои знания и умения в полной мере.

На рассмотрение жюри научным комитетом было представлено 27 задач. После долгих обсуждений было выбрано по три задачи на каждый тур. При выборе задач учитывался также тот факт, что в нынешнем году отмечается юбилей А.С. Пушкина. Для каждой задачи был подобран эпиграф из его произведений.

По своей тематике предложенные на олимпиаде задачи отличались достаточным разнообразием и оригинальностью. Там были представлены многие интересные разделы информатики: компьютерная геометрия (задача № 1 “Круг из прямоугольников”), поиск на графах (задача № 2 “Новый алхимик”), динамическое программирование (задача № 3 “Песнь”), сортировка и последовательности (задача № 4 “Хлопоты многодетного купца”), моделирование (задача № 5 “Микрофония”), математическая лингвистика (задача № 6 “Стихоплет”). Особой оригинальностью отличалась задача “Песнь”. Неумение до конца разобраться с текстом задачи является ахиллесовой пятой многих школьников. Каждый раз мы повторяем: читайте условие задачи внимательно, каждое слово и каждая фраза в нем имеют вполне определенный смысл. И тем не менее снова и снова именно об это спотыкаются многие ребята. Жюри еще раз хотело подчеркнуть важность проблемы формализации, и это в значительной степени определяло “олимпиадность” задачи “Песнь”. Как впоследствии оказалось, не многие участники взялись за решение этой задачи, но те, что до конца разобрались с формулировкой задачи, практически полностью ее решили.

Проверка решений участников олимпиады осуществлялась по тестам с помощью автоматизированной системы тестирования. Максимальное количество баллов, которое мог набрать каждый участник по результатам олимпиады, составляло 200, по 100 баллов за каждый тур. Распределение баллов по задачам представлено в таблице.

Название задачи	Количество баллов
Круг из прямоугольников	33
Новый алхимик	33
Песнь	34
Хлопоты многодетного купца	30
Микрофония	35
Стихоплет	35

По итогам олимпиады абсолютным чемпионом в третий раз стал Мартыянов Владимир, ученик 11-го класса из Нижнего Новгорода (155 баллов). Только один балл проиграл чемпиону занявший второе место Пастухов Роман, ученик 9-го класса из Оренбурга, что является для Романа несомненным успехом. Третье место занял Баутин Михаил (136 баллов), ученик 10-го класса из Нижнего Новгорода. Совсем немного проиграл третьему призёру еще один юный участник олимпиады — Митричев Петр, ученик 8-го класса из Москвы (134 балла). На пятом месте Бабенко Максим, ученик 11-го класса из Саратова (127 баллов). Замкнул шестерку лучших Жежерун Андрей, ученик 10-го класса из Самары (117 баллов). В общей сложности победителям олимпиады было вручено 7 дипломов первой степени, 22 диплома второй степени и 29 дипломов третьей степени. Итоговые результаты победителей олимпиады представлены на стр. 4, 5.

Результаты решения задач на этой олимпиаде вызвали неоднозначную оценку членов жюри и научного комитета. С одной стороны, отлично выступила группа школьников, для которых эта олимпиада не первая. С другой стороны, около 50% участников в общей сложности полностью не решили даже одной задачи из шести, то есть набрали в итоге менее 32 баллов. Наряду с этим нельзя не заметить, что в группе лидеров в этом году больше учащихся 8—10-х классов, чем школьников выпускных классов. Из семи победителей олимпиады, награжденных дипломами первой степени, только два одиннадцатиклассника. Самой приятной неожиданностью для организаторов и жюри олимпиады было участие в соревнованиях ученика 5-го класса из Сарова Нижегородской области Тарадая Юрия. Хоть он и не набрал много баллов, но сам факт его участия говорит о многом.

Торжественно прошло закрытие олимпиады. Помимо почетных дипломов, всем победителям было вручено большое количество памятных подарков, любезно предоставленных спонсорами олимпиады.

По итогам олимпиады и по результатам предварительных сборов отобраны кандидаты в сборную команду России по информатике, которая примет участие в международной олимпиаде. В этом году международная олимпиада состоится в Турции в период со 2 по 8 октября 1999 г. Хотелось бы надеяться на высокий уровень подготовки всех кандидатов на поездку в Турцию. Особенно хотелось бы пожелать успехов Мартыянову Владимиру, для которого эта олимпиада будет третьей, а на двух предыдущих он становился абсолютным победителем.

Результаты XI олимпиады школьников по информатике

Место	Ф.И.О.	Класс	Город	Учебное заведение	Сумма баллов	Диплом
1	Мартьянов Владимир	11	Нижний Новгород	Лицей № 40	155	1
2	Пастухов Роман	9	Оренбург	Гимназия № 1	154	1
3	Баутин Михаил	10	Нижний Новгород	Лицей № 40	136	1
4	Митричев Петр	8	Москва	Школа № 827	134	1
5	Бабенко Максим	11	Саратов	Физ.-техн. лицей № 1	127	1
6	Жежерун Андрей	10	Самара	Самарский муниципальный университет Ноянковой	117	1
7	Круглов Алексей	10	Нижний Новгород	Лицей № 40	116	1
8	Никокошев Илья	11	Москва	СУНЦ МГУ	94	2
9	Зиновьев Никита	11	Санкт-Петербург	Физ.-мат. лицей № 231	86	2
10	Паньгин Андрей	11	Сосновый Бор, Ленинградская обл.	Лицей № 8	82	2
11	Анохин Павел	11	Орел	Лицей № 40	81	2
12	Пестов Андрей	10	Киров	Физ.-мат. лицей	80	2
13	Крылов Рюрик	11	Троицк, Челябинская обл.	МОУ № 16	80	2
14	Прокушкин Иван	11	Киров	Физ.-мат. лицей	79	2
15	Поярков Алексей	10	Рыбинск, Ярославская обл.	Многопрофильный лицей № 2	76	2
16	Курочкин Юрий	10	Санкт-Петербург	Аничков лицей	70	2
17	Юсупов Тимур	10	Нижний Новгород	Лицей № 40	70	2
18	Четвертаков Евгений	11	Новосибирск	СИМП ВКИ НГУ	70	2
19	Хвостиков Иван	9	Орел	Гимназия № 19	63	2
20	Куршев Евгений	10	Переславль-Залесский, Ярославская обл.	Гимназия	63	2
21	Жмогинов Андрей	11	Нижний Новгород	Лицей № 40	63	2
22	Рожков Михаил	11	Владивосток, Приморский край	Средняя школа № 75	63	2
23	Волков Сергей	9	Мурманск	Мурманский политехнический лицей	62	2
24	Квят Сергей	9	Нальчик, Кабардино-Балкария	ММОУ "Лицей № 2"	62	2
25	Альхов Геннадий	11	Санкт-Петербург	Физ.-мат. лицей № 239	61	2
26	Красин Иван	9	Калуга	Школа № 48	59	2
27	Жиляев Владислав	10	Орел	Гимназия № 19	59	2
28	Хопрячков Владимир	9	Нальчик, Кабардино-Балкария	ММОУ "Лицей № 2"	58	2
29	Лобанов Павел	11	Брянск	Средняя школа № 17	57	2

Место	Ф.И.О.	Класс	Город	Учебное заведение	Сумма баллов	Диплом
30	Зверев Илья	11	Псков	Псковский технический лицей	55	3
31	Сытник Сергей	11	Казань	Физ.-мат. лицей № 131	55	3
32	Леденев Сергей	10	Орел	Гимназия № 19	54	3
33	Данилов Александр	11	Саров, Нижегородская обл.	Лицей № 3	54	3
34	Савич Павел	11	Москва	Лицей № 1502	54	3
35	Москаев Кирилл	10	Москва	СУНЦ МГУ	53	3
36	Воронин Дмитрий	11	Оренбург	Средняя школа № 18	51	3
37	Штучкин Александр	10	Саратов	Физ.-техн. лицей	50	3
38	Наумов Александр	10	Ульяновск	Гимназия № 33	49	3
39	Сабинин Иван	10	Новосибирск	Муниципальная гимназия № 3	49	3
40	Строганов Андрей	11	Дубна, Московская обл.	Лицей "Дубна"	49	3
41	Филоненко Филипп	11	Пермь	Физ.-мат. школа № 146	49	3
42	Дремов Владимир	10	Волгодонск, Ростовская обл.	Школа № 24	48	3
43	Кордыс Александр	11	Саранск, Республика Мордовия	Школа № 34	48	3
44	Кошман Денис	11	Петрозаводск, Карелия	Университетский лицей	48	3
45	Федюнин Виталий	11	Томск	МОУ "Академия"	47	3
46	Козлов Андрей	11	Нововоронеж, Воронежская обл.	Средняя школа № 5	46	3
47	Егупов Константин	10	Тула	Средняя школа № 14	44	3
48	Фурманов Кирилл	10	Фрязино, Московская обл.	Средняя школа № 1	44	3
49	Кириллин Александр	9	Санкт-Петербург	Физ.-мат. школа № 261	43	3
50	Алферов Роман	10	Челябинск	Физ.-мат. лицей № 31	41	3
51	Асемов Алексей	10	Саранск, Республика Мордовия	Гимназия № 12	40	в/к
52	Шагламджян Карен	10	Санкт-Петербург	Физ.-техн. школа им. Иоффе	39	3
53	Горбунов Максим	11	Железногорск, Красноярский край	Школа № 91	39	3
54	Романов Иван	10	Саратов	Лицей колледжа прикладных наук	38	3
55	Антонов Михаил	11	Якутск, Республика Саха (Якутия)	Республиканский колледж	38	3
56	Бородин Тимофей	11	Кострома	Лицей № 34	38	3
57	Золотухин Юрий	11	Златоуст, Челябинская обл.	Школа № 15	37	3
58	Гумеров Максим	11	Уора, Башкортостан	Лицей при УГАТУ	36	3
59	Свердин Александр	10	Москва	Средняя школа № 609	35	3

Задачи XI Всероссийской олимпиады школьников по информатике

Санкт-Петербург, 2–8 апреля 1999 г.

Задача 1. Круг из прямоугольников

имя входного файла: input.txt
 имя выходного файла: output.txt
 ограничение времени на каждом тесте: 10 с без учета времени записи результата на диск
 баллы: 33

*Но и вдали, в краю чужом,
 Я буду мыслию всегдашней
 Бродить Тригорского кругом...*

А.С. Пушкин

Плоскость разбили на одинаковые прямоугольники размера $M \times N$ со сторонами, параллельными осям координат, и вершинами, расположенными в точках $(M-i, N-j)$, где i и j принимают всевозможные целые значения. Пусть на этой плоскости задана точка $P(x, y)$ с целочисленными координатами. Назовем расстоянием от точки P до некоторого прямоугольника наименьшее из расстояний от P до точек этого прямоугольника, включая его границу. В частности, расстояние от точки до прямоугольника, в котором она содержится, равно 0.

Требуется написать программу, перечисляющую прямоугольники, удаленные от P на расстояние, не превосходящее L . Прямоугольники должны быть перечислены в порядке неубывания этого расстояния.

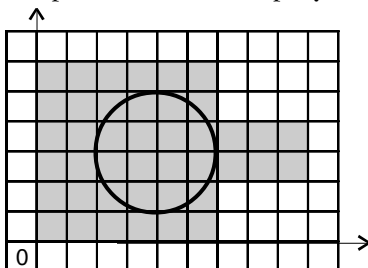
Формат входных данных

Во входном файле содержатся целые числа M, N, L, x и y ($0 < M \leq 10, 0 < N \leq 10, 0 \leq L \leq 300, -30000 < x, y \leq 30000$), разделенные пробелами и/или переводами строк.

Формат выходных данных

Выведите в выходной файл координаты левых нижних углов искомых прямоугольников в описанном выше порядке. Прямоугольники, равноудаленные от P , могут выводиться в произвольном порядке.

Ниже приведен пример входного и выходного файлов для случая, представленного на рисунке.



Пример входного файла

```
3 2 2
4 3
```

Пример выходного файла

```
3 2
3 0
0 2
3 4
0 0
0 4
6 2
```

Задача 2. Новый алхимик

имя входного файла: input.txt
 имя выходного файла: output.txt
 ограничение времени на каждом тесте: 10 с
 баллы: 33

*Счастливым день! могу сегодня я
 В шестой сундук (сундук еще не полный)
 Горсть золота накопленного всыпать.
 Немного, кажется, но понемногу
 Сокровища растут.*

А.С. Пушкин

Алхимик Петя изобрел философский камень, с использованием которого можно проводить некоторое множество алхимических реакций по превращению одних веществ в другие. Масса каждого вещества, вступающего в реакцию, и масса каждого образующегося в результате реакции вещества составляет ровно один грамм. Естественно, закон сохранения массы при этом может нарушаться.

Изначально у Пети имеется один грамм свинца. С помощью философского камня Петя может превратить свой свинец в другие вещества, на которые он потом также сможет воздействовать философским камнем. Выполняя одну за другой алхимические реакции, Петя стремится получить как можно больше золота.

Требуется написать программу, определяющую по заданному описанию алхимических реакций, выполняемых философским камнем, наибольшее количество золота, которое может получить Петя.

Формат входных данных

В первой строке записано целое число K ($1 \leq K \leq 6$) — количество различных веществ, участвующих и образующихся в алхимических реакциях.

Вторая строка содержит список этих веществ, разделенных пробелом (в списке обязательно есть свинец и золото). Названия веществ не длиннее 10 букв.

В третьей строке записано целое число $L (1 \leq L \leq 100)$ — количество типов реакций, выполняемых философским камнем.

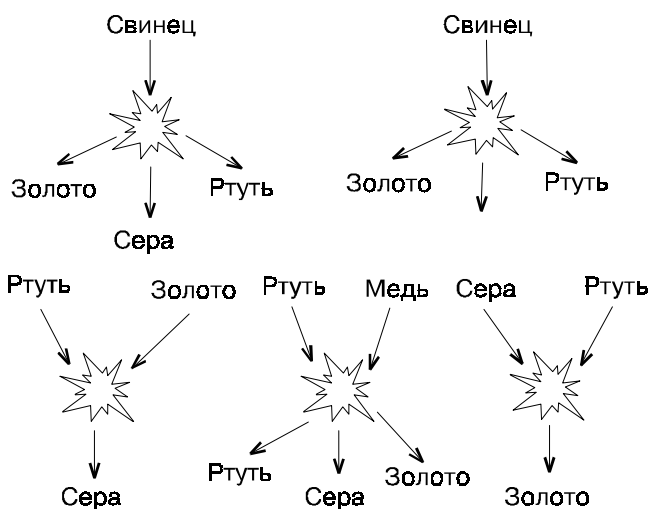
Далее в файле идут L описаний этих реакций. Каждое описание реакции состоит из двух строк:

- в первой строке — названия веществ, вступающих в реакцию,
- во второй строке — названия веществ, получающихся в результате реакции.

Формат выходных данных

Ваша программа должна вывести в выходной файл либо одно целое число — искомое количество граммов золота, либо сообщение QUANTUM SATIS, если Петя может получить любое количество золота.

Ниже приведены примеры входного и выходного файлов для представленных на рисунке алхимических реакций, выполняемых философским камнем.



Задача 3. Песнь

имя входного файла:	input.txt
имя выходного файла:	output.txt
ограничение времени на каждом тесте:	10 с
баллы:	34 балла

*...Звуки мертвив,
Музыку я разъял, как труп. Поверил
Я алгеброй гармонию...*

А.С. Пушкин

В традиционной музыке используются музыкальные звуки из некоторого набора, именуемого *звукорядом*. Звуки звукоряда принято группировать в *октавы*, в каждой из которых 12 звуков. Порядковый номер звука в пределах одной октавы назовем *нотой*. Таким образом, каждый звук можно задать парой чисел —

номером октавы и номером ноты. Номер октавы K — произвольное целое число, номер ноты N принимает значение из интервала $[0, 12]$. Звуки можно обозначать этими двумя числами, записанными рядом без пробелов (второе число всегда двузначное). Эту запись назовем *кодом* Q . Например, $Q = -108$ для (-1) -й октавы, восьмой ноты. Значение Z , определяемое по формуле $Z = K \cdot 12 + N$, назовем *абсолютным номером* Z звука в звукоряде (для приведенного выше примера $Z = -4$).

Набор всех звуков, ноты которых принадлежат заданному подмножеству P номеров нот, назовем *гармонией* G . Это означает, что любой звук с абсолютным номером $Z = K \cdot 12 + n$, где n — номер ноты из P — принадлежит этой гармонии при любом значении K . Отсюда следует, что гармония однозначно определяется указанием P . Две гармонии назовем *эквивалентными*, если при прибавлении некоторого одного и того же целого числа ко всем абсолютным номерам звуков первой гармонии получаются все элементы второй гармонии. Ограничимся рассмотрением только таких наборов гармоний, в которые наряду с каждой из гармоний входят и все эквивалентные ей. Для описания набора такого вида достаточно указать из каждой совокупности эквивалентных по одной гармонии G_i или соответствующему ей подмножеству P_i *Базой* B набора гармоний назовем совокупность всех таких P_i .

Всякую совокупность одновременно звучащих звуков (не менее двух) будем называть *аккордом* A . Для некоторого заданного набора гармоний назовем *гармонией аккорда* A такую гармонию G из него, что все звуки аккорда A принадлежат G .

Будем говорить, что некоторый звук *в тему* для некоторого аккорда, если этот звук принадлежит хотя бы одной гармонии из множества всех гармоний этого аккорда.

Последовательное звучание произвольных звуков назовем *мелодией*. Каждому звуку мелодии может быть сопоставлен аккорд в порядке исполнения мелодии. Будем считать мелодию *благозвучной* для этой последовательности аккордов, если каждый ее звук оказывается в тему для соответствующего ему аккорда. *Кучерявостью* мелодии назовем сумму модулей разностей абсолютных номеров Z последовательно исполняемых звуков данной мелодии.

Задание

Пусть даны: база B набора гармоний, последовательность аккордов A и начальный звук Q . Требуется написать программу, находящую наименее кучерявую из всех благозвучных мелодий, начинающихся с этого звука.

Формат входных данных

В первой строке записано целое число M — количество заданных элементов в базе набора гармоний ($1 \leq M \leq 200$). Далее следуют M строк, каждая из которых содержит описание одного из элементов в базе набора гармоний

в виде последовательности составляющих ее номеров нот, записанных через пробел. В следующей строке находится целое число L — количество заданных аккордов ($1 \leq L \leq 200$). Каждая из последующих L строк содержит описание одного из аккордов в виде последовательности кодов составляющих его звуков, записанных через пробел. Описания аккордов следуют в порядке их исполнения. Последняя строка входного файла содержит код начального звука Q . Все значения кодов звуков записываются тремя цифрами.

Формат выходных данных

В первую строку выходного файла следует вывести минимально возможную кучерявость среди всех мелодий, удовлетворяющих описанным выше требованиям. Оставшиеся строки должны содержать L целых чисел — коды Q звуков, составляющих соответствующую мелодию. Если вариантов мелодий несколько, нужно вывести любую из них. Для заданных во входном файле данных всегда будет существовать хотя бы одна благозвучная мелодия.

Пример

входного файла

```
3
1 5 8 11
1 5 8 12
1 6 8
5
101 106
010 112
-101 004
201 202
110 111
102
```

Пример

выходного файла

```
4
102 104 104 106 106
```

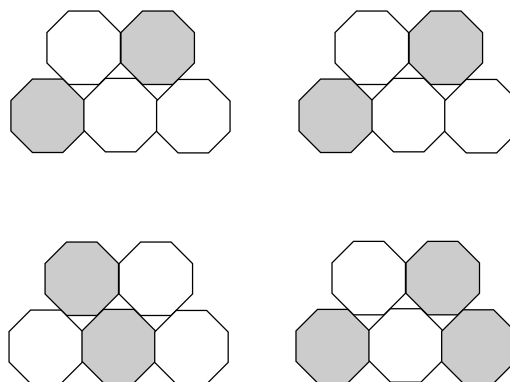
Задача 4. Хлопоты многолетнего купца

имя входного файла:	input.txt
имя выходного файла:	output.txt
ограничение времени на каждом тесте:	10 с
баллы:	30

Среди заботливых купцов...
А.С. Пушкин

Было у купца N дочерей-красавиц. Путешествуя по заморским странам, захотел он привезти каждой дочери в подарок по набору драгоценных камней. Что бы никого из дочерей не обидеть, купец решил, что все наборы должны состоять из равного количества камней, одинаково стоить и любые два набора должны отличаться хотя бы одним камнем (см. рисунок).

Заморский ювелир представил купцу список различных видов камней с указанием цены каждого вида, пообещав продать любое количество камней любого вида.



Одинаковые наборы

Разные наборы

Требуется написать программу, определяющую наборы для всех дочерей согласно указанным выше требованиям. При наличии нескольких решений программа должна выдавать одно любое.

Формат входных данных

Первая строка входного файла содержит целое число N — количество дочерей ($1 \leq N \leq 100$).

В следующей строке файла задано целое число K — количество имеющихся у ювелира видов камней ($1 \leq K \leq 100$).

Далее следуют K строк. Каждая строка содержит название вида камня и его цену, разделенные пробелом. Название вида камня — слово, длина которого не превышает 10 букв, цена камня — целое число, не превышающее 32 767.

Формат выходных данных

Выходной файл должен содержать описание N наборов, по одному описанию в каждой строке. Описание набора состоит из следующих пар: вид камня и количество камней данного вида. Все слова и числа внутри строки должны быть разделены пробелами.

Если решение найти не удастся, выходной файл должен содержать единственное слово NO.

Пример

входного файла

```
2
3
янтарь 1
изумруд 2
алмаз 3
```

Пример

выходного файла

```
алмаз 1 янтарь 2
изумруд 2 янтарь 1
```

Задача 5. Микрофония

имя входного файла:	input.txt
имя выходного файла:	output.txt
ограничение времени на каждом тесте:	10 с
баллы:	35



ИНФОРМАТИК **A**

еженедельное приложение
к газете «ПЕРВОЕ СЕНТЯБРЯ»

**М
О
Д
Е
Р
Н
И
З
А
Ц
И
Я**



В.М. НЕЧАЕВ

**КОМПЬЮТЕРНОГО
КЛАССА**

Настройка локальной сети

Итак, Windows 95 я установил, и описание этого процесса, таящего в себе несколько “подводных камней”, дал в прошлый раз (см. № 25/99). Ходом установки тогда руководил Мастер — так называют особую программу, которая непосредственно во время работы вела со мной оживленный диалог по поводу разного типа устройств, составляющих конфигурацию компьютера, а также и относительно того, что включать в набор программных компонентов системы, а что пока не надо. Имея уже опыт общения с Мастером, я придерживался, как всегда, определенной линии поведения, а именно, старался ограничиться только необходимым, особенно когда это касалось устройств. Моя цель была простой — получить работающую систему, пусть хотя бы и не в полном составе. Все, что не было затребовано сразу, могло быть добавлено потом. А то, бывает, наказыываешь так много всего, что в итоге ничего и не получишь, поскольку вероятность сбоя в работе Мастера при этом заметно увеличивается. Мастер, как говорится, тоже человек, и не следует перегружать его заданиями, связанными с распознаванием и настройкой всевозможной компьютерной периферии.

Особые трудности могли возникнуть из-за сетевой платы. Я даже на время установки вообще изъясил ее из компьютера — открыл корпус и осторожно вынул из разъема, от греха подальше, чтобы Мастеру даже и дела никакого не было до сети. Разумеется, приходилось отказываться и от всех программных компонентов, связанных с сетью. В таком облегченном варианте все прошло очень гладко, и вот мой компьютер уже стал работать под управлением новой операционной системы. Теперь пожалуйста, хочешь — измений ее, хочешь — дополний, уже без особого риска заказывая то, что не употреблялось в первый раз. Как раз дополнением я сейчас и намерен заняться, и прежде всего надо бы вернуть сетевую плату на место, чтобы уж закрыть после этого крышку системного блока, покончив таким образом с аппаратной частью.

Одно только меня интересует, пока компьютер еще работает в первоначальном составе, сразу после той описанной минимальной установки. Я хочу посмотреть, что же в данный-то момент имеется на жестком диске, от чего мне потом вести отсчет его заполнения новыми файлами. Кроме того, я, может быть, внесу какие-нибудь безобидные изменения в режимы работы системы, хотя бы для пробы. Система Windows 95 очень хорошо к этому приспособлена.

Как мне удобно, так и сделаю

А зачем изменять, вроде же все нормально действует? Нет, на самом деле некоторые настройки мне, быть может, захочется поменять, ну и просто любопытно испытать возможности управления системой. Открываю двойным щелчком сначала **Мой компьютер**, а затем диск С и вижу две папки, **Program Files** и **Windows**, да еще три файла в “корневой директории”, если говорить по-старому. В действительности файлов не три, а в несколько раз больше, но не каждый заказан для просмотра. Чтобы увидеть их все, я

щелкаю¹ по меню **Вид**, а внутри него — по нижней строке **Параметры . . .** и далее по закладке **Просмотр**.

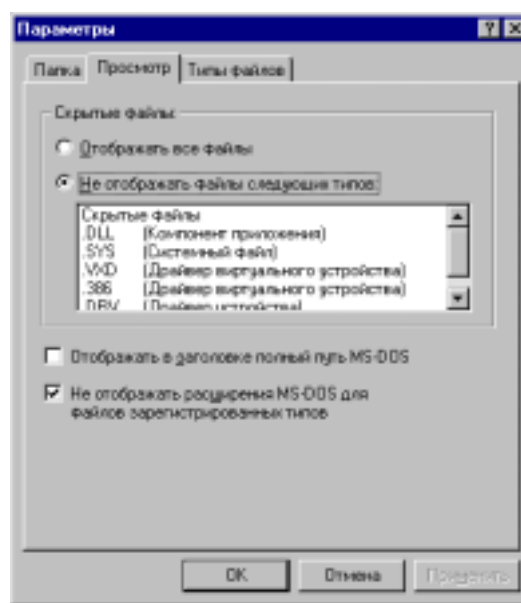


Рис. 7.1. А если я хочу наоборот?

Вот такие параметры просмотра папок имеют силу в настоящий момент, а я возьму, да и поставлю вместо этого черную точку напротив пункта **Отображать все файлы**. И еще, заодно уж, чтобы второй раз не заходить сюда, уберу галочку из строки **Не отображать расширения MS-DOS** — пусть отображает! Кроме того, желательно уместить все внутри небольшого окошка, безо всяких там линеек прокрутки; поэтому придется еще раз зайти в меню **Вид** и выбрать вместо **Крупных значков** пункт **Список**, я лично всегда именно его выбираю, если объектов в окне много.

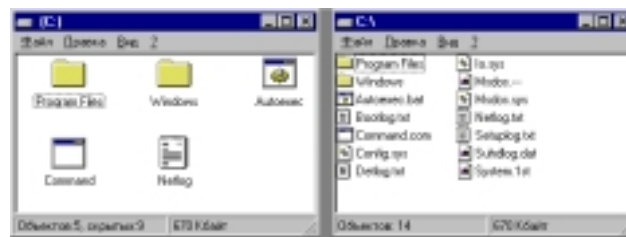




Рис. 7.2. Одно и то же окно, но в разных режимах просмотра

Впрочем, это я так, хочу лишь посмотреть, что есть на диске. В действительности же начальная настройка просмотра как раз была очень хороша (за исключением только **Крупных значков**), и я потом вернусь именно к ней. Потому

¹ Тут уже один раз, а не два. Вообще правило такое: по экранным кнопкам или пунктам разных меню — щелчок однократный, по дискам же, папкам и файлам, чтобы их открыть, — двойной.

что файлы некоторых типов и не должны отображаться (под “некоторыми” понимаются, как это видно из первого рисунка, чисто служебные файлы, которые своим присутствием в окне лишь затрудняют выбор действительно нужных значков). Конечного пользователя надо ведь ориентировать не на обслуживание машины, а на прикладные задачи, решаемые с ее помощью². Да и галочку для расширений тоже нужно вернуть на место, и вот почему. Когда я через некоторое время буду устанавливать Microsoft Office и другие приложения, большинство из них при установке регистрируются операционной системой, вследствие чего, например, расширение имени файла DOC для документов редактора Word будет ею пониматься, но не будет мне показываться. Это и само по себе лучше, поскольку название “Моя статья.DOC” режет глаз, но самое главное, если понадобится переименовать файл, например, в “Мое выступление”, а я поленюсь переходить на латинский шрифт или просто забуду указать для нового имени расширение DOC, то этот файл перестанет распознаваться системой как соответствующий документ, он как бы потеряет связь с текстовым редактором, в котором был создан. Я по значку этого переименованного (с пустым, получается, расширением) файла буду щелкать мышью, чтобы открыть документ для работы, а Windows будет спрашивать: “Да, но с помощью какого же приложения его обычно открывают?” И ничего не поделаешь, придется разяснять, с помощью какого. Нет, начальный режим просмотра был именно таким, каким надо.

А вот для переключателя режима работы клавиатуры, на индикаторе которого сейчас написано **Ru**, действительно нужны изменения. Щелкаю по этому значку мышью, только использую не левую ее клавишу, как ранее, при работе с DOS, а правую (очень полезную для Windows 95), и в вызванном таким способом контекстном меню, то есть в списке операций, которые возможно произвести с данным объектом, из двух предложенных пунктов выберу **Свойства**³. Появится карточка, где можно поставить черную точку напротив сочетания  — вместо левый . Так будет гораздо удобнее (для руки) переходить с режима ввода русских букв на режим ввода латинских букв, и наоборот.

Теперь посмотрю значки на **Рабочем столе**. Не маловато ли их, всего лишь четыре штуки? Нет, не маловато, ведь на машине нет еще прикладных программ, одна только система. Более того, считаю, что один из значков даже лишний. Самый нижний, **Установка Microsoft Network** (сокращенно MSN), можно, не жалея, выбросить в корзину. Все равно данной службой компьютерной связи с прославленной фирмой мне пользоваться не придется. И не из-за того даже, что для этого требуется модем, которого в классе пока нет. Просто расплачиваться за услуги MSN надо с помощью каких-то кредитных карточек, а как — я не знаю. Да вообще и нет у меня такой уж сильной потребности обращаться за технической поддержкой прямо в Microsoft.

Между прочим, я выбрасываю даже не саму процедуру связи с MSN, а всего-навсего ее ярлык. То есть маленький файл размером 318 байт, имеющий расширение имени LNK (*link* — связь), вся задача которого — указывать, где на диске находится соответствующая программа с именем MSNEXCH.EXE (в папке C:\WINDOWS\SYSTEM), чтобы долго ее не искать, а прямо с рабочего стола запускать на выполнение. Получается, что и этот файл с установщиком мне не нужен, а потому надо не полениться и удалить его тоже. Тогда на диске освободится уже 17 килобайт⁴. Тут следует отметить, что из всего записанного на винчестер с самого начала, даже в моем минимальном варианте установки, какая-то часть (чуть ли не 10%) хранится зря, то есть никогда не потребуется⁵. Однако сейчас я не буду думать о таких мелочах, в конце концов пока свободное место на диске есть.

Что бы такое еще настроить, может, поиграть со свойствами экрана? Довольно увлекательно, по крайней мере на первых порах, размещать на рабочем столе разные фоновые рисунки, узоры, заставки, менять цветовые схемы, добавлять к указателю мыши длинный шлейф. Все это забавно, хотя и быстро надоедает; но ведь есть еще и звуковые эффекты, захочется их тоже попробовать и т.д. Нет уж, не буду даже и начинать, дело на безделье не меняют. Гораздо важнее другое — сколько места на диске осталось? Винчестер-то у меня маленький, его емкость всего лишь 340 мегабайт, из которых, между прочим, 40 и еще 45 уже израсходованы на два дополнительных логических диска: E (для дистрибутива системы) и D (для файлов, создаваемых учащимися в течение учебного года). Основной диск C имел, таким образом, с самого начала емкость 255 Мб, а сейчас, естественно, какая-то его часть занята системой. Так вот какая?

Рисунок 2 не может, конечно, ввести меня в заблуждение: 670 Кб относятся к суммарному объему лишь тех 12 файлов, имена которых указаны в окне; две же папки, содержащие основную часть системы, здесь в счет не идут. Чтобы узнать размер такой папки, надо посмотреть ее свойства (правая клавиша, последний пункт). Возьмем, к примеру, **Windows** — на диске эта папка, содержащая около тысячи файлов, занимает почти сорок мегабайт.

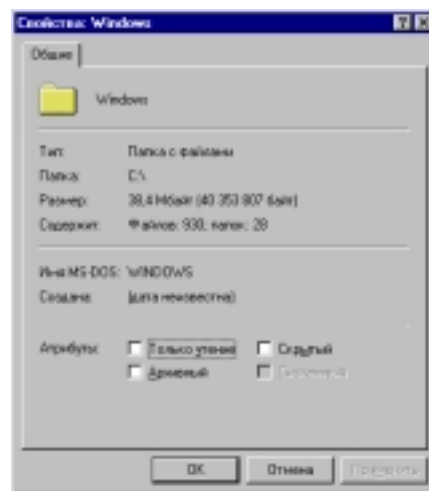


Рис. 7.3. Общий взгляд на папку Windows

² Недаром при работе с Windows 95 принято употреблять новые термины: не директория, а папка; не программа, а приложение; не текст или рисунок, а документ.

³ Тут уже левой, а не правой. Также правило — и в этом случае, и вообще: правая клавиша предоставляет целый спектр возможностей, а левой осуществляется какое-то конкретное действие из этого спектра. И еще: многоточие всегда означает, что это только начало диалога, — тебя еще будут спрашивать о подробностях, что да как.

⁴ Все эти сведения о ярлыке и его программе-хозяйке я получил, щелкнув по нему правой клавишей мышки и выбрав пункт **Свойства**.

⁵ Например, много лишних шрифтов, правда, я еще не знаю точно, какие можно спокойно удалять.

Вторая папка — **Program Files** — гораздо меньше (до двух мегабайт не дотягивает), но в дальнейшем, по мере установки разных приложений, она, конечно, разрастется. Впрочем, то же самое можно сказать и о первой, о **Windows**, рост которой будет происходить главным образом за счет пополнения одной из вложенных в нее папок, а именно **System**. Сейчас она имеет размер 20 мегабайт (360 файлов), а может увеличить свой объем чуть ли не на порядок. Причем очень часто пользователь компьютера по неведению совершенно теряет контроль над данным процессом. Тут надо придерживаться по мере возможности одного важного правила инсталляции-деинсталляции программных продуктов, но о нем я расскажу в какой-нибудь из следующих статей цикла. Сейчас это не тема нашего разговора, просто, чтобы потом сразу иметь точку отсчета, я должен отметить для себя, сколько мегабайт и какими папками занято. И в дальнейшем постоянно следить за расходом дискового пространства. Все-таки, что бы ни говорилось, система Windows 95 рассчитана на работу с современными — гигабайтными — винчестерами, где уже не приходится экономить на каждой мелочи. А у меня другого пути нет, кроме как поддерживать строгий порядок при использовании имеющихся ресурсов.

Ну, это все правильно, однако пора сетевую плату на место возвращать и, закрыв системный блок, включать питание. По идее, новое устройство должно быть автоматически опознано операционной системой. Более того, поскольку и BIOS моей машины, и сама плата поддерживают технологию *Plug-and-Play*, мне, возможно, и настраивать ничего не потребуется: ни указывать адреса ввода/вывода информации (I/O Address), ни подбирать свободный, не занятый другими устройствами номер запроса на прерывание (IRQ). Что ж, посмотрим.

Так и оказалось!

Да, сетевая плата была правильно опознана. Как только я включил питание и началась загрузка системы, на экране появилось на пару секунд небольшое сообщение, которое я тем не менее успел прочитать. Сначала располагался заголовок **Обнаружено новое устройство** — речь шла о “Plug-and-Play Ethernet Card”, — а затем указывалось: Сейчас идет установка программного обеспечения для него. Какое именно обеспечение выбрать — было предложено мне с помощью “диалоговой” карточки уже привычного вида:

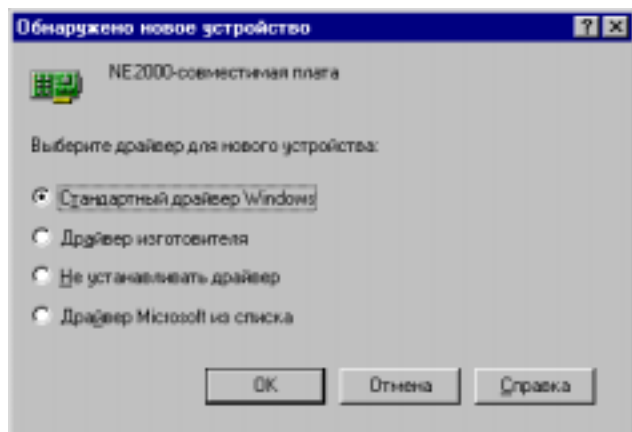


Рис. 7.4. Плата у меня самая обычная

В последнем пункте подразумевается, что я точно знаю фирму-изготовитель и конкретную модель сетевой платы, и в этом случае, если только данная модель известна Windows 95, для нее будет найден на дистрибутиве подходящий драйвер. На случай совсем уж уникальной платы есть второй пункт — **Драйвер изготовителя**, чтобы система могла взять нужную программу с прилагаемой к устройству дискеты. Дискета-то у меня и правда имеется, но вряд ли есть необходимость ее использовать. Думаю, что ничего специального здесь не требуется, и полагаюсь на стандартный драйвер. В ответ на ОК система выдает следующий текст: Для идентификации компьютера в сети необходимо задать имена компьютера и рабочей группы — и даже сопровождает его несколько резким звуковым восклицанием. А я разве возражаю против идентификации? Наоборот, согласен, и поэтому получаю на экране карточку, озаглавленную кратко **Сеть**.



Рис. 7.5. У каждого компьютера должно быть свое имя

Что это за имя такое — **Default** (по умолчанию)? Лучше уж в соответствии с нашей отечественной традицией дать более простое название — компьютер номер такой-то. Поначалу я было хотел употребить какие-нибудь красивые, например, “звездные” имена — Вега, Сириус, Альтаир, однако не решился, представив себе путаницу, которая может произойти. Так что в классе у меня двенадцать рабочих мест с именами **01**, **02** и так далее до **12**⁶.

Причем все они составляют одну рабочую группу **206**, по номеру аудитории, а в графу **Описание компьютера** можно и совсем ничего не вводить. Хотя если есть желание, то почему бы не вписать что-либо, вроде приглашающего **Имеется принтер** или сурового **486DX2/16/340**. Описание будет появляться на экранах у других участников рабочей группы, когда они станут обращаться к сети, рядом с изображением моей машины, помогая им ориентироваться.

⁶ Именно так, с нулями впереди, чтобы при указании в виде списка 11-й и 12-й не вклинивались между 1-м и 2-м.

Все это, конечно, дело вкуса, все подобные названия и описания; кроме того, при желании их в любой момент можно изменить. Но карточка **Сеть** имеет, помимо закладки **Компьютер**, где она сейчас раскрыта, еще две другие. Та, что слева, то есть **Конфигурация**, содержит более важную информацию. Я к этой закладке перейду, чтобы просмотреть список программных компонентов сети, которые предлагается установить.

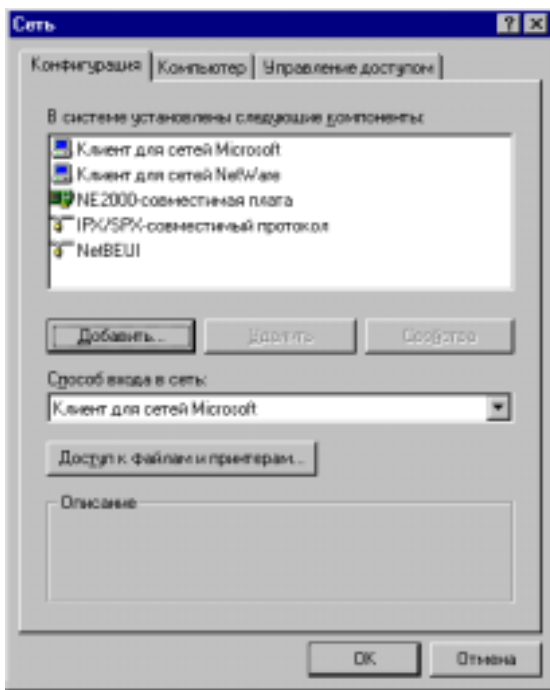


Рис. 7.6. Вот так все и будет, если я не вмешаюсь

Сейчас здесь пять строчек: два клиента, два протокола и одна плата. Спрашивается, зачем мне того и другого по два, — обязательно ли так нужно, или это просто сделано с запасом? Очевидно, с запасом, на тот случай, когда в сети есть не только Windows-машины, но еще и серверы другой операционной системы NetWare. Может быть, когда-то так в моем классе и будет, но пока что это явное излишество. Значит, с одним клиентом я могу совершенно свободно расстаться.

Далее, что касается протокола, то для клиента сети Microsoft вполне достаточно одного-единственного NetBEUI. Второй же, IPX/SPX, тоже может быть использован, однако обязательным является лишь для клиента NetWare. Это сразу становится ясным, когда хочешь отказаться от услуг данного протокола. А именно, стоит только выделить четвертую строку, как две экранные кнопки **Удалить** и **Свойства**, до сих пор бледные, приобретают яркость, то есть оказываются доступными, и если нажать первую из них, то будет удален не только отмеченный протокол, но и связанный с ним неразрывно клиент NetWare. Таким образом, в списке останется уже не пять строк, а всего три.

Будет ли сеть правильно функционировать с таким набором? Нет, не будет, надо обязательно добавить еще один элемент — службу доступа к файлам и принтерам компьютера. Непонятно даже, почему она не включена в начальный список, ведь без нее мою машину не увидят в сети другие пользователи. А потому я щелкаю по кнопке **Дос-**

туп к файлам и принтерам и в появившемся после этого дополнительном окошке ставлю одну галочку (верхнюю) в любом случае, а вторую (нижнюю) — если именно к данному компьютеру подключен единственный в моем классе принтер.

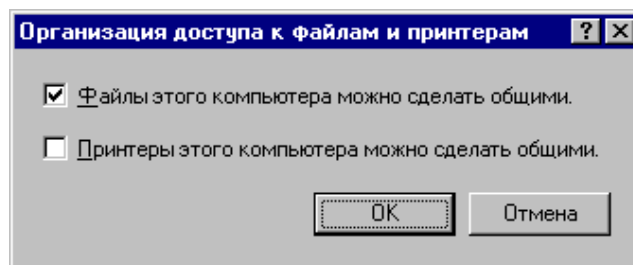


Рис. 7.7. Для того и вхожу в сеть, чтобы обмениваться файлами

Причем галочка еще не означает, что любому желающему уже разрешено свободно читать мои файлы или печатать на моем принтере. Нет, пока речь идет только о потенциальной возможности доступа к моим ресурсам. А право доступа (кому что позволить, а кому — нет) я буду в дальнейшем определять сам. Сейчас я лишь заявляю о своем желании это сделать. В результате к списку программных компонентов сети добавляется еще одна строка, и теперь здесь фигурирует по одному представителю каждого вида программного обеспечения: один клиент, одна плата, один протокол и одна служба.

В закладке **Конфигурация** нерассмотренным остался последний настраиваемый параметр — способ входа в сеть. Предлагается входить в сеть в качестве клиента сетей Microsoft, однако имеется и альтернатива: как обычно в подобных случаях, используя черный треугольничек-стрелку⁷ слева, можно просмотреть перечень возможностей и выбрать подходящую. Советую на первых порах ограничиться **Обычным входом в Windows**, и вот почему. Сеть мне нужна прежде всего для того, чтобы можно было разделять общие ресурсы класса: большой винчестер учительской машины, ее же принтер, привод CD-ROM, а в дальнейшем модем, сканер и т.д. Такое деление не требует очень уж серьезного администрирования, поскольку мои ученики обладают в общем-то равными правами, тем более что все рабочие места располагаются в одной комнате, где я слежу за порядком и всегда могу вмешаться, если возникнет какой-то спор.

Другое дело, когда пользователи сети имеют разный статус, скажем, когда наряду с учащимися пользователями являются также директор, завуч, бухгалтерия, отдел кадров. При входе в сеть с их компьютеров, пожалуй, стоит регистрироваться под разными именами, чтобы режим работы, например, начальства, отличался от режима работы подчиненных. А вообще администрирование сети — большая отдельная проблема, к разговору о которой, надеюсь, мы еще вернемся, причем мне представляется преждевременным браться за ее решение сразу при установке сети. Надо поработать сначала, не осложняя себе жизнь дополнительными хлопотами, а когда

⁷ Я называю такие экранные кнопки-треугольнички — «Огласите весь список, пожалуйста!».



освоишься, то никто не помешает перейти к чему-нибудь более изощренному. Да еще неизвестно, будет ли в этом нужда.

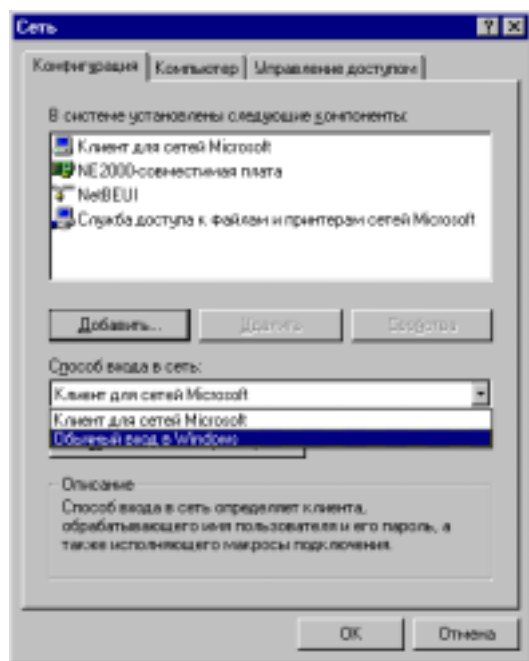


Рис. 7.8. Ради упрощения загрузки системы

Впрочем, регистрация при входе в сеть — это только первый шаг к администрированию, сама по себе она ни к чему не обязывает. Так что можно выбрать любой из вариантов. В более сложном случае, при включении компьютера во время загрузки системы, у клиента сети будет запрашиваться имя пользователя, а если он захочет его защитить, то еще и пароль. При обычном же способе входа в сеть никаких расспросов не последует, а доступ к ресурсам будет одинаковым для всех. Мне как раз это сейчас и подойдет.

В третью же закладку **Управление доступом** можно заглянуть, да только менять там ничего не придется — как предложено по умолчанию **На уровне ресурсов**, так и следует оставить. Другой вариант — **На уровне пользователей** подошел бы, если бы в сети имелся сервер Windows NT, то есть специально выделенный компьютер с установленной на нем более мощной операционной системой, позволяющей хранить списки пользователей с присвоенными им правами доступа. Но это дело будущего, когда еще я заведу такой сервер, пока же надо принять все как есть.

Таким образом, рассмотрев все закладки и подобрав то, что надо, я покидаю карточку **Сеть**, чтобы продолжить процесс настройки. Происходит копирование недостающего программного обеспечения с дистрибутива системы, который, напомним, у меня всегда под рукой, на логическом диске E, после чего, спустя минуту, следует предложение перезагрузиться. Я, естественно, соглашаюсь и жду появления рабочего стола на экране монитора. Однако сталкиваюсь с одной небольшой формальностью. Тут мне не совсем ясно, ведь я же заказал обычный вход в Windows, и тем не менее система почему-то запрашивает у меня имя

и пароль. Видимо, в первый раз положено это сделать, хочешь или не хочешь. В качестве имени пользователя я, недолго думая, ввожу имя компьютера⁸, то есть его номер, тот же самый, который вводился на карточке **Сеть**, а вместо пароля просто нажимаю клавишу **Enter**. Получается, что пароль у меня пустой, но все равно его полагается еще и подтвердить. Опять нажимаю клавишу **Enter**, ну теперь все, установка сети закончена, и на рабочем столе появляется новый значок **Сетевое окружение**, прямо под **Моим компьютером**.

Одно слово — техника!

Но не всегда все проходит так гладко, иногда бывает, что система ошибается, например, с выбором прерывания для сетевой платы, и тогда надо заниматься этим самому, в ручном режиме. Уже во время перезагрузки я могу увидеть тревожное сообщение о том, что сетевая плата работает с ошибками, а с какими именно, надо будет разбираться. Прежде всего следует перейти к значку **Система** (через **Пуск**, **Настройку** и **Панель управления**) и конкретно в закладку **Устройства**. Стоит по ней щелкнуть мышью, как она раскрывается, причем прямо указывая восклицательным знаком на строчку, где встретилась неполадка.

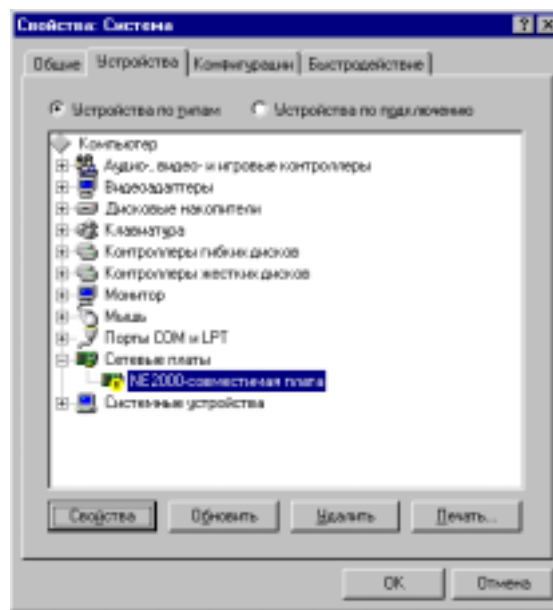


Рис. 7.9. Что-то с сетью не в порядке

Что за неполадка? — а посмотри свойства выделенного элемента. Тут одно из двух, либо само устройство опознано неверно, либо оно с чем-нибудь ранее установленным конфликтует:

⁸ А вообще надо различать имя компьютера — оно одно, и имена пользователей, которых может быть сколько угодно разных. С одной и той же машины в сеть могут входить разные пользователи, и наоборот, один пользователь может входить в сеть под своим именем с разных рабочих мест.

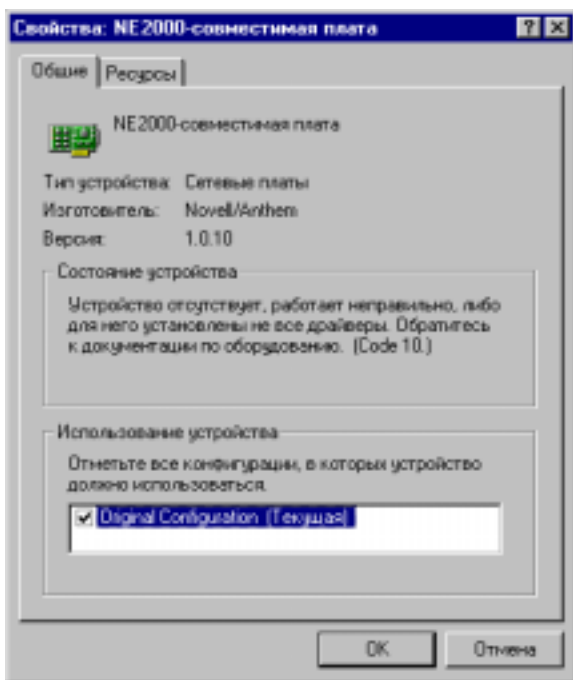


Рис. 7.10. Не то устройство

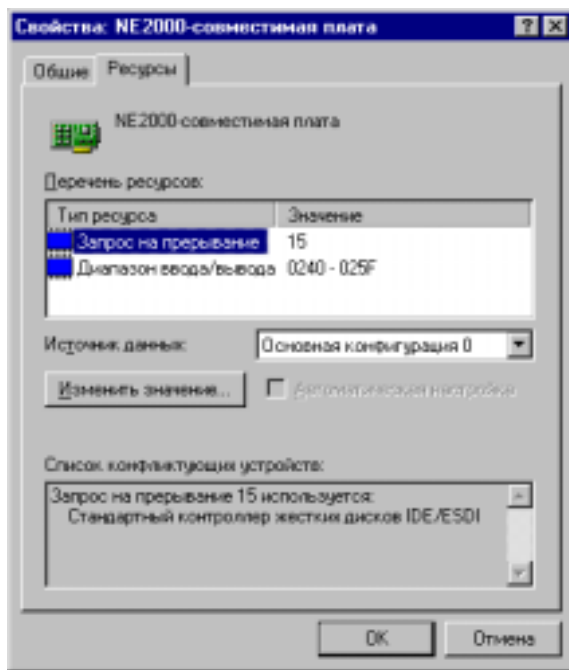


Рис. 7.11. Не тот номер прерывания

Ну, номер прерывания это не так страшно, я могу его поменять на другой, незанятый, только чтобы плата была в состоянии с ним функционировать, так как каждая плата может быть настроена на некоторый довольно широкий, но все же ограниченный набор номеров. Мне просто надо, выбрав тип ресурса (он выделится синим цветом), щелкнуть по экранной кнопке **Изменить значение**. При этом, возможно, необходимо будет предварительно снять галочку с **Автоматической настройки**, если она там есть. Справа от номера видны стрелки вверх-вниз, вот с их помощью я и делаю новый, бесконфликтный выбор.

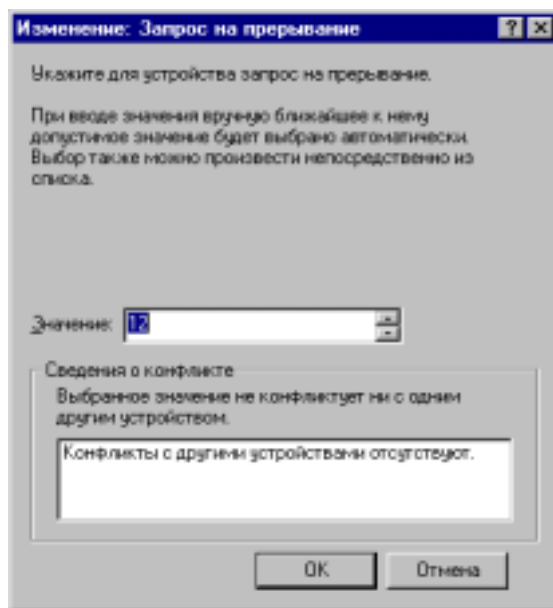


Рис. 7.12. Несколько прерываний свободны

Хуже, если плата не та. Ошибочную плату, то есть ту, что с восклицательным знаком, необходимо удалить из системы (для данной операции имеется соответствующая экранная кнопка), затем нужно запустить **Мастер установки оборудования** — в панели управления есть такой. Он, конечно, предложит сделать все автоматически, но я-то уже знаю, к чему это приведет, и потому должен буду ответить отказом, взяв всю ответственность на себя: в списке оборудования (по типу) укажу на сетевые платы, а среди них выберу, как мне представляется, нужную.

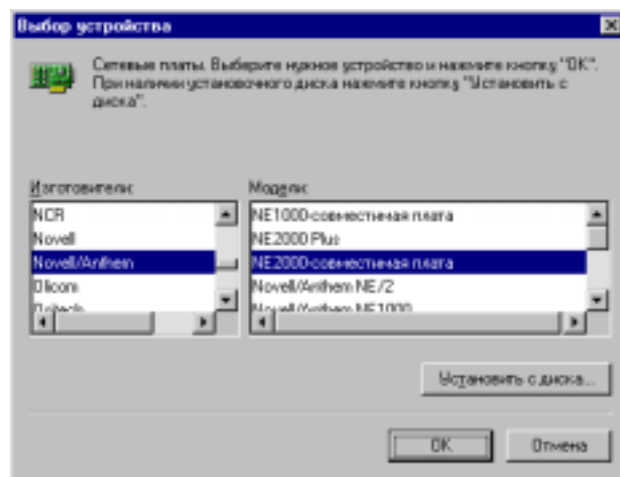


Рис. 7.13. По-моему, так

Не исключено, что в ручном режиме эта повторная, по сути дела, попытка окажется более удачной, но, возможно, потребуется дискета с подходящим исключительно к данной плате драйвером, хотя даже и ее наличие не всегда гарантирует успех. Вообще это дело темное, у меня бывало, что от простой, так сказать, перемены мест слагаемых, например, от изменения очередности установки разных устройств, результат тоже менялся, и тут можно посоветовать только одно — набираться опыта, раз за разом повто-

ря весь процесс и слегка варьируя сопутствующие условия. Лично мне помогало то, что машин в классе много: с одной что-то не получается, переходишь на другую, на третью, а потом, глядишь, и с первой все образуется.

Короче говоря, тем или иным способом добиваемся, чтобы сетевая плата установилась и настроилась правильно. Однако это еще не все. Допустим, плата заработала; тут, наверное, все получится автоматически, не потребуются даже прибежать к ручному режиму. Но вот работает ли сеть? То есть вся сеть целиком, не отдельная плата. А практика покажет: просто попытаюсь выйти на связь с кем-нибудь. На самом деле даже никого и не надо, можно к самому себе обратиться, но не напрямую, через **Мой компьютер**, а как к участнику рабочей группы, через **Сетевое окружение**. Это ведь не одно и то же. Я еще не устанавливал никаких прав доступа к конкретным ресурсам своей машины для других рабочих мест (хотя и заказал соответствующую службу, занимающуюся этим), а потому и когда обращусь сам к себе через сеть, то не получу в распоряжение ни одной папки, ни одного файла, ни одного диска, вообще ничего. Лишь увижу, что да, есть такой компьютер в сети, с именем, скажем, **04**, но и только. Впрочем, этого уже достаточно — увидеть в сети самого себя, чтобы сделать вывод о ее работоспособности.

И что же я получу после того, как двойным щелчком открою **Сетевое окружение**? В его окне появится список всех функционирующих в данный момент компьютеров, которые входят в ту же рабочую группу, что и моя машина (а рабочая группа, напомню, у нас одна, и ее имя **206**).

Причем отображается список почти мгновенно, и уж во всяком случае самого-то себя я должен увидеть, даже если другие компьютеры пока еще в сеть не включены. Ну а если в окне долго ничего нет, кроме стилизованного фонарика, символизирующего поиск, то дело плохо. Единственное, что появится через минуту, — это значок **Вся сеть**, но радости от него будет мало. Попытка войти туда ни к чему хорошему не приведет. Система просто сообщит, что сеть недоступна, и предложит обратиться к справочной системе.

Однако обращаться к ней не надо, уверяю, что толку не будет. Лучше проверить кабель, соединяющий данную машину с другими. Сначала отсоедините его совсем от платы:

себя компьютер должен тогда “увидеть” точно!⁹ Если же и тут неудача, то я уже ничего не понимаю. Конечно, нелишним будет проверить, все ли в порядке со свойствами **Сетевого окружения** (правая кнопка, последняя строка); иногда я замечал у своих машин одну особенность: стоило мне заглянуть в закладку **Компьютер**, только заглянуть, даже не меняя ничего, как сеть начинала после этого работать.

Предположим все-таки, что, будучи изолированным от сети, мой компьютер сам себя видит. Очень хорошо, тогда переходим к соседнему и, добившись от него того же, пробуем соединить машины между собой кабелем. Но только именно между собой и ни с чем больше, так, чтобы снова терминаторы ограничивали линию с обоих ее концов. Если каждая машина себя видит, а другую нет, то, значит, кабель плохой и, вероятнее всего, наконечники были приделаны неудачно. Надо взять другой отрезок кабеля и повторить опыт. Однако если для всех отрезков картина одна и та же, то вряд ли причина в соединении. Беру самый тяжелый случай, в действительности до этого редко доходит, но тем не менее раз уж такое случится, значит, надо проверять все настройки: и протоколы, и прерывания, и службы, и т.д. и т.п. вплоть до переустановки системы! Ничего не поделаешь, на ошибках тоже учатся. К своей сети я знаете сколько приноравливался, потратив при этом много сил, и душевных, и физических. А как иначе, что в нашей жизни дается без труда?

Хотел рассказать еще и о дальнейших своих действиях, связанных с организацией сети, совместным использованием объединяемых ею ресурсов — принтера, привода CD-ROM, жестких дисков, программ, да так устал, преодолевая трудности настройки и добиваясь нормальной работы сети, что вынужден подождать с этим до следующей статьи.

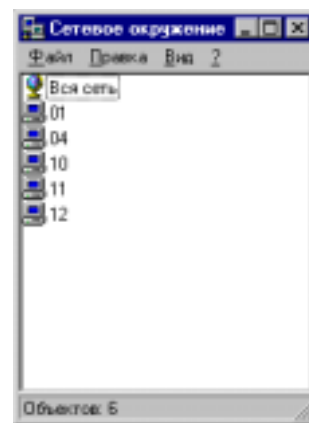


Рис. 7.14. Все в порядке, связь есть

Статья 8

Доступ к ресурсам локальной сети

С прошлого учебного года в моем компьютерном классе работает локальная сеть Windows 95. С ее помощью я могу пересылать файлы с одной машины на другую, а также с любой из них обращаться к удаленным устройствам — в первую очередь к приводу CD-ROM, который у меня один на весь класс, и к принтерам, которых у меня, так уж получилось, два. Предыдущая статья была посвящена тому, как производилась установка программного обеспечения, управляющего сетью (выбор драйвера для сетевой платы и протокола для связи, заказ той или иной службы доступа и способа входа в сеть и т.д.). А до того была статья о монтаже аппаратной части сети. Чаще всего процесс установки проходит плавно, без осложнений: система сама опознает плату и подбирает к ней драйвер и тут же конфигурирует ее, то есть подбирает свободный номер прерывания и адрес ввода-вывода так, чтобы сетевая плата не конфликтовала с ранее

установленными устройствами. В какие-то моменты я тоже принимал участие в настройке различных служб, выбирая состав программных компонентов сети и определяя подходящие режимы их работы.

Бывают, конечно, сбои на этом пути, и некоторые из них я описывал. В принципе эти сбои устраняются посредством ручной (не автоматической) установки; есть ряд хитростей, помогающих обойти опасные этапы. Рано или поздно на **Рабочем столе** появляется значок **Сетевое окружение**, и любой компьютер, включенный в сеть, должен “увидеть” в этом самом окружении и сам себя, и все остальные. Именно такой картинкой заканчивалась предыдущая ста-

⁹ Может быть, для большей уверенности в развязке с окружающей средой на разъем платы надеть T-коннектор, а вместо кабелей на оба его конца — по терминатору?



тья — в раскрытом окошке **Сетевого окружения** были видны несколько одинаковых по виду значков, и каждый из них отображал отдельного участника сети: компьютер по имени 01, компьютер по имени 02, по имени 03 и т.д. — мне казалось удобным называть машины именно так.

А где же все файлы, где принтеры?

Однако “видеть” — еще не означает “иметь доступ”. Попытка войти, к примеру, в компьютер 03 не приведет ни к чему, я просто получу у себя на экране совершенно пустое, без единого объекта окно. То есть видеть-то 03 я вижу, канал связи функционирует, но передавать информацию по этому каналу, оказывается, невозможно. Мне получить от этой машины нечего, ни одного файла я не обнаруживаю в ее раскрытом окне, да и какой-нибудь свой файл послать мне некуда — ни чужого диска я не нахожу, ни хотя бы папки, ни принтера. Почему так? Ведь во время установки сети я, помнится, заказывал такой программный компонент — **Служба доступа к файлам и принтерам сетей Microsoft**, и он работает, в чем легко убедиться, просматривая (с помощью правой клавиши мыши) свойства **Сетевого окружения**.

Дело в том, что служба доступа отвечает лишь за самую техническую возможность иметь сообщение по каналу связи. И связь с 03 действительно установлена. Но согласен ли владелец данного компьютера, чтобы его винчестер, или привод CD-ROM, или принтер были доступны всем участникам сети, — неизвестно¹⁰. Ведь в одноранговой сети Windows 95 решение о том, какими конкретно ресурсами машины и в какой мере смогут воспользоваться другие члены рабочей группы, принимает исключительно ее хозяин, работающий на ней. И это естественно. Никто не имеет права без разрешения считывать файлы с моего жесткого диска, печатать на принтере, подключенном к моему компьютеру, обращаться к моему приводу CD-ROM.

Хотя, если быть таким строгим, то зачем тогда было и сеть входить? Принтер все-таки не мой личный, а общественный, и привод CD-ROM тоже. А по поводу жесткого диска — весь его, конечно, не стоит делать общедоступным, но несколько избранных папок или даже специально для обмена файлами заведенных — это пожалуйста. Причем можно при желании все разделяемые ресурсы защищать паролями. Допустим, чтобы учащиеся не гоняли принтер почем зря, задам-ка я на той машине, к которой подключен принтер, пароль и лично сообщу его только тем, кому доверяю. То же относится и к папкам на моем винчестере, тут даже два вида паролей существует: для полного доступа и только для чтения, что вполне разумно. Правда, пароли имеют обыкновенное раскрываться со временем, но кто же мне мешает менять их раз в месяц, или в неделю, или хоть каждый день?

Но такая ситуация — когда у каждой машины есть свой постоянный владелец — является идеальной. В классе же известно что: за одним и тем же компьютером работают по очереди несколько учеников, на одном уроке — один, на другом — другой, на третьем — третий. И здесь обязанности хозяина, очевидно, должен брать на себя я, называясь при этом администратором сети и определяя, какие диски или папки открыть для сети полностью, какие — только для чтения, а какие совсем не открывать, какие пароли задавать и

как часто их менять. Разумеется, все мои установки легко могут быть изменены первым же, кто этого пожелает. Сев за машину, он откроет для доступа другие папки, введет новые пароли, закроет доступ к принтеру — что захочет, то и сделает. И мне придется постоянно следить за порядком, время от времени восстанавливая исходное состояние. Вообще это целая проблема, хорошо знакомая преподавателям информатики. Причем ладно бы такого рода деятельность учащих касалась лишь безобидных (более или менее) фоновых рисунков или экранных заставок, но ведь они норовят и панель задач расположить как-нибудь иначе, и к мыши шлейф длиною в пол-экрана привязать, а то еще кто-то из озорства скорость двойного щелчка догадается установить максимальной, так что не сможешь открыть ни одного значка. И это, конечно, далеко не все, что способен изобрести пытливым ум, стараясь изменить жизнь к лучшему. Что же, теперь и сетевое взаимодействие бросать на произвол судьбы?

Мои слушатели на занятиях часто спрашивают о способах защиты настроек компьютера от безудержной фантазии школьников; интересуются, знаю ли я такие способы, входят ли такого рода средства в состав самой системы (или это должны быть отдельные программы), где их достать, как с ними работать и т.д. Ответа от меня ждут с таким живым интересом, что сразу видно, насколько это актуально для школы, особенно в последнее время, в связи с распространением домашних компьютеров и повсеместным переходом на Windows 95, прямо-таки подталкивающим пользователя к экспериментам с управлением. Что ж, я знаю и системные средства, и другие программы и хочу об этом рассказать в следующей статье, но сейчас, чтобы не путать два вопроса, касаться защиты не буду. Хотя понимаю, что для сети важность защиты еще более возрастает с увеличением числа автономных рабочих мест.

Все мое — твое

Сегодня у меня другая тема: организация доступа к ресурсам, предоставляемым отдельными компьютерами для сети. Прежде всего я хочу обмениваться файлами — значит, в каждом компьютере надо разрешить доступ к его жесткому диску. Ну и мне к своим. Вообще-то у меня три диска; на С (где содержатся система и прикладные программы) и Е (где хранится дистрибутив Windows 95) го-стям, пожалуй, делать нечего, но вот D (где учащиеся сохраняют свои упражнения, тексты, рисунки и другие документы) как будто специально создан для такого обмена. В окне **Моего компьютера** я щелкаю правой клавишей мыши по значку диска D и в появившемся контекстном меню выбираю строку **Доступ...** В ответ получаю карточку с тремя закладками, причем как раз на нужной она открыта (рис. 8.1).

Правда, поначалу здесь черная точка-переключатель была в верхнем кружке **Локальный ресурс**, это уже я ее поставил пониже, напротив **Общего ресурса**. Вообще говоря, имя, под которым будет фигурировать мой диск D на экранах соседей, может быть заменено на любое другое, лишь бы им было понятно, с кем они имеют дело. Лично я, попробовав разные варианты, пришел к выводу, что удобнее всего оставить то, что предложено, — просто D. В поле **Заметки**, как легко убедиться, можно поместить все, что угодно, хотя, на мой взгляд, заметки там не нужны: они лишь будут сбивать с толку соседей.

¹⁰ В данном случае как раз, наоборот, известно, раз я ничего не увидел в его окне.

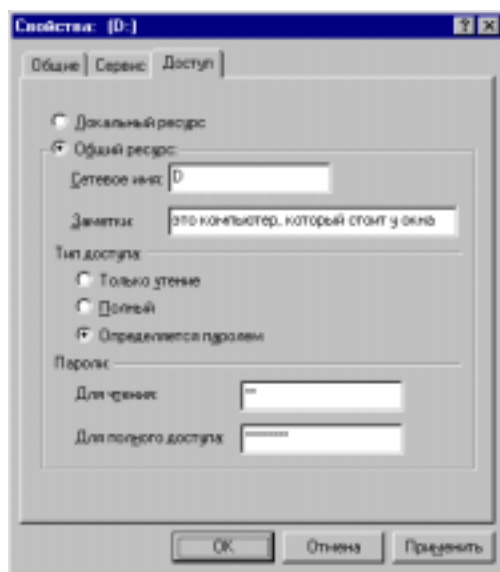


Рис. 8.1. Открываю доступ к диску

Впрочем, это кому как нравится, а вот к паролям следует отнестись более серьезно. Установить переключатель на **Только чтение** представляется не совсем удобным. Хотелось бы дать **Полный доступ**, чтобы на мой диск можно было пересылать файлы с других машин, иначе что же это за общение. Но, с другой стороны, так ведь кто-нибудь возьмет да и сотрет с него пару-тройку моих файлов, что совсем не желательно. Как быть? Ну, например, можно задать не просто полный доступ, но еще и пароль к нему. Наиболее же универсальным является третий вариант, как на рисунке, когда тип доступа **Определяется паролем** и задает не один пароль, а два. Обращается, допустим, к моему диску кто-то со своего рабочего места, а система требует: «Назови пароль!» В зависимости от того, что именно он введет, и права получит соответствующие. Своим людям я сообщу одно секретное слово, малознакомым — другое, а посторонние не будут знать их вообще.

Правда, тут с паролями получается такая канитель, что лучше от них совсем отказаться и сделать вот как. Установить для диска D все же тип доступа **Только чтение**, так будет надежнее. А для присылаемых мне файлов создать на этом диске отдельную папку, назвать ее **Для посылки** или еще короче — **Прием** — и точно так же, как это делалось для диска, установить для нее доступ, только уже полный и тоже без пароля. Надо кому-то переслать мне файл — папка **Прием** к его услугам. Надо что-то от меня получить — пожалуйста, вот мой диск D. В системе все хорошо предусмотрено: тип доступа для всего диска не будет мешать доступу к отдельной папке, даже если она расположена внутри него. Для соседей оба моих ресурса, и диск, и папка, предстанут как два равноправных объекта. И никаких паролей.

Я, пожалуй, так и сделаю, а что получится, как это будет выглядеть с экрана другого компьютера — покажу на рис. 8.5. Перед тем как этот рисунок приводить, я заодно уж и принтер, подключенный к моей машине, сделаю общим, чтобы сразу все было видно. Да и на моем компьютере его надо установить, ведь этого еще не было сделано. Вот три следующих рисунка для иллюстрации этого и послужат.

А принтеров у меня два

Один хороший, струйный, хотя и не цветной, — Canon, он подключен к компьютеру 12, а второй похуже — старый матричный Epson, подключен к 07. Надо сделать так, чтобы со всех рабочих мест имелся доступ к обоим: струйный я берегу и печатать на нем буду только важные бумаги, а Epson использую в качестве рабочей лошадки. Соответственно и права доступа к ним надо будет разграничить.

Итак, я сначала установлю Canon, сидя за машиной 12. В окне **Моего компьютера** имеется папка **Принтеры**, а в ней пока что только один значок программы-мастера **Установка принтера**. Обращаюсь к нему, чтобы запустить на выполнение соответствующую программу. С мастером жизнь хорошая: с помощью серии карточек он тебя за руку проводит по всем этапам сложного пути. Прежде всего спрашивает, локальный ты принтер устанавливаешь или сетевой. До сетевого дело еще не дошло, пока что локальный. Хорошо, локальный, но какой именно — фирма, модель? Я выбираю из списка: фирма Canon, модель VJ-230.

Далее. К какому порту он присоединен? Естественно, к LPT 1. Далее. Какую дать подпись к значку этого принтера? Мастер предлагает указать здесь название модели, и я с ним согласен. Тогда еще один вопрос на этой же карточке: использовать ли данный принтер в качестве принтера по умолчанию? То есть, допустим, если принтеров будет несколько, а я закажу печатать документ, но не укажу специально, на каком принтере, то на этом ли печатать? Да, пусть пока так, потом разберусь, когда их будет два. Далее. Напечатать ли прямо сейчас пробную страницу? Мастер рекомендует напечатать, но я позволю себе с ним не согласиться, чтобы избежать лишних хлопот.

Нет так нет, отвечает Мастер. Тогда уже не **Далее**, а **Готово**. Идет копирование необходимого программного обеспечения с дистрибутива (а дистрибутив у меня всегда под рукой, на диске E). Ну вот и все, в окошке **Принтеры**, откуда я запускал на выполнение Мастера, появляется значок установленного принтера Canon. Сразу же щелкаю по значку правой клавишей мыши и в строке **Доступ** заказываю **Общий ресурс**. Но, в отличие от того, как это было для диска, пароль какой-нибудь поставлю. Между прочим, замечаю, что теперь у значка принтера, как и ранее у значков диска D и папки **Прием**, снизу подрисовывается изображение кисти руки.

Отлично! Все то же самое я делаю и на компьютере 07, пересев туда:

- и у него диск D делаю общим, причем только для чтения;
- и общую папку **Прием** на этом диске создаю с полным доступом;
- и принтер локальный устанавливаю, только теперь не Canon, а Epson;
- и доступ к нему открываю общий, без пароля.

Более того, раз к этому моменту принтер Canon с машины 12 уже доступен всем в сети, то и его можно установить на компьютере 07 как второй. Поэтому я еще раз запускаю на выполнение Мастера установки принтера, только в соответствующей карточке помечаю, что принтер не локальный, а сетевой. А раз так, то Мастер просит указать путь к нему:

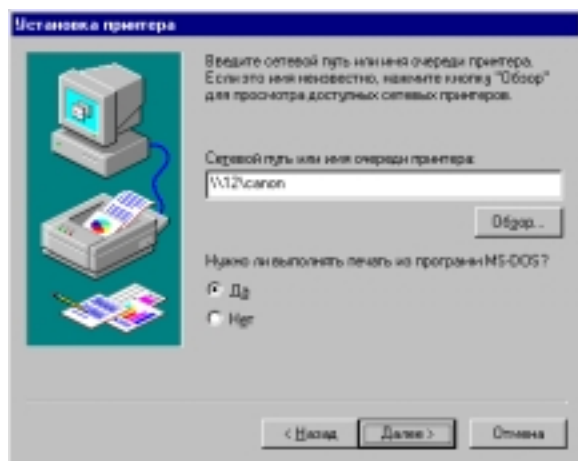


Рис. 8.2. Путь к сетевому принтеру

Путь я, конечно, не задавал вручную, а использовал кнопку **Обзор**. Там были представлены только те два компьютера из всей моей сети, на которых имелся уже установленный локальный принтер. Я указал на 12, а в нем на Canon, после чего Мастер вернул меня к предыдущей карточке, с уже заданным в поле **Сетевой путь** адресом. Сам я вряд ли бы догадался, что путь должен начинаться с двойной наклонной черты, являющейся знаком того, что поиск устройства ведется не на моей машине, а в сети. Оказывается, это принцип всех сетевых имен — двойная наклонная черта впереди.



Рис. 8.3. Принтеров в моей сети два

По поводу переключателя внизу для печати из программ MS-DOS я не знаю, что сказать, данный вопрос остался для меня неясным. На всякий случай ответил “Да”, но что такой ответ значит и в чем тут отличие от предложенного вначале “Нет” — не понимаю. Как бы там ни было, сетевой принтер установлен, и это видно по значкам в папке. Причем между ними имеется небольшое отличие: под сетевым принтером изображен кабель, а под локальным — просто рука.

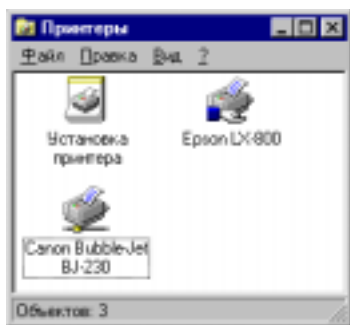


Рис. 8.4. Оба принтера доступны

Взгляд со стороны

На всех других компьютерах я делаю то же самое, что и на машинах с номерами 12 и 07, с одной только поправкой: для них оба принтера — сетевые. Ну и для всех без исключения рабочих мест я теперь укажу (используя правую клавишу мыши), что принтером по умолчанию надо считать Epson.

Пришла пора посмотреть, в каком виде предстанет один компьютер на экране другого. Например, с машины номер 12 я через **Сетевое окружение** обращусь к 07 и вот что увижу в его окошке:



Рис. 8.5. Один компьютер на экране другого

И диск D, и папка **Прием** внешне выглядят одинаково, как папки, несмотря на то, что доступ к диску D — только для чтения, а к папке **Прием** — полный. Чтобы это проверить, надо попытаться записать какой-нибудь свой файл на диск и в папку: в первом случае ничего не выйдет, а во втором запись должна осуществиться. Однако если раскрыть диск D и, увидев там ту же самую папку **Прием**, попробовать записать туда свой файл, то ничего уже не получится (что вполне логично, раз я подхожу к этой папке не напрямую, а через закрытый для записи диск D).

Если бы на 07 имелся привод CD-ROM, обозначенный как F, то и его надо было бы сделать общим ресурсом. С паролем или без, но он тоже выглядел бы в сетевом окне как папка.

Напоследок расскажу о том, как можно облегчить сам процесс доступа к сетевому ресурсу, если приходится часто им пользоваться. К примеру, на компьютере 01 (учительском) я создал на **Рабочем столе** папку, в которую помещаю файлы с заданиями для своих учеников, чтобы они эти файлы переносили к себе на машины и там с ними работали. Папку эту, понятно, надо сделать общей, с доступом, только для чтения без пароля. А еще одну папку, тоже общую, но с полным доступом, создать для присылаемых обратно выполненных работ. Это с моей стороны. А что со стороны учеников? Неужели каждый день добираться до заданий, открывая сетевое окружение, разыскивая компьютер 01, вход в него? Как бы сделать так, чтобы обе папки всегда были наготове?

Отвечаю. Можно с каждого рабочего места присвоить этим папкам имена в виде букв, которыми обычно обозначают диски, и поместить “диски” в окне **Мой компьютер** (или в каком угодно другом). Делается это так. В выбранном окне заказывают действие **Подключить сетевой диск** — щелкают по экранной кнопочке с таким названием на панели инструментов¹¹. Появляется карточка запроса с двумя полями ввода: какой ресурс подключить и какую букву диска назначить.

¹¹ В любом окне эту панель можно отобразить, щелкнув по меню Вид и выбрав в нем первый пункт.

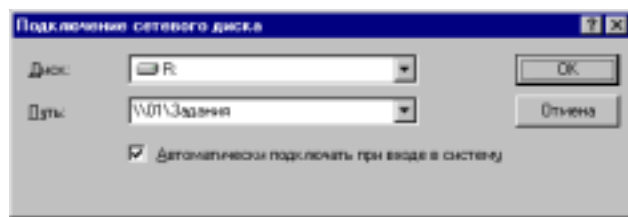


Рис. 8.6. Редко встретишь диск R

Я сознательно беру буквы, далеко отстоящие от C, D, E, F, — чтобы не спутать реальные диски с фиктивными. Впрочем, их и так по внешнему виду не спутаешь.

Подключение делается один только раз и при следующих запусках компьютера будет обеспечиваться автоматически, в соответствии с указанной в запросе галочкой. С диска R файл с заданием ученик копирует в начале урока к себе на рабочий стол и делает с ним, что нужно. Результат потом отправляет на диск S учителю для проверки — и порядок. Довольно удобно. Если же не понравится работать с сетевыми дисками, то можно их отключить, есть соответствующая кнопочка на панели инструментов.

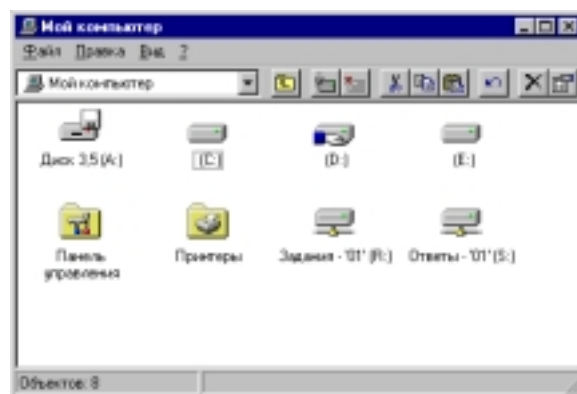


Рис. 8.7. Под сетевыми дисками — кабель

Кому-то понравится, а кому-то нет. Один подключит, другой отключит, а третий вообще закроет доступ к своим ресурсам. Сеть-то одноранговая, каждый “сам себе режиссер”, и порядка не жди! Только и забот будет у учителя, что ходить по рядам да восстанавливать подключения. Впрочем, о защите настроек, как обещал, в следующей статье.

Статья 9

Ограничение прав пользователя

Как защитить компьютеры от перенастройки? Каждый день, заканчивая занятия в своем классе, я говорю учащимся: “Пожалуйста, не выключайте машины, мне нужно привести их в порядок”. Действительно, во время уроков кто-то поменял фоновый рисунок на **Рабочем столе**, кто-то **Панель задач** сдвинул набок да к тому же сделал ее автоматически убирающейся с экрана, кто-то способ переключения раскладок клавиатуры другой установил, а кому-то пришло в голову в локальной сети права доступа к своим дискам изменить. Во всем этом, может, ничего страшного и нет, но мне как преподавателю, воспитанному в традиционном духе коллективной деятельности, на уроке бывает неудобно вести занятия, когда у одного ученика на экране одно, а у другого — другое.

Например, я объявляю классу новую тему и говорю: “Сегодня мы будем работать с такой-то программой. Управлять ею довольно сложно, поэтому будьте внимательны и точно выполняйте мои указания. Для начала, чтобы запустить программу, щелкните мышкой по ее значку”. Но мне тут же с нескольких рабочих мест наперебой чуть ли не кричат: “А где этот значок, у нас на экране его нет!” Как нет, ведь я собственными руками создал на **Рабочем столе** ярлык на кануне, когда устанавливал ту программу. Что же, кто-то его удалил, что ли? Придется открывать корзину, восстанавливать, а если ярлыка там нет, то создавать новый. Но в любом случае уже потеряно время, а начало урока скомкано.

Это, конечно, мелочь, да и вряд ли ученики будут удалять что-либо со стола, скорее они новых ярлычков понаделают, что, впрочем, тоже не хотелось бы пускать на самотек. Но сейчас речь идет не о конкретных проявлениях свободы действий со стороны пользователей, а о моей принципиальной позиции. Для меня необходимым элементом рабочей обстановки в классе является все-таки некоторое единообразие в настройках компьютеров. Включая сюда даже оформление

экрана — хотя различия тут особенно и не мешают преподавать, но и не помогают, поскольку могут отвлекать внимание на посторонние предметы. И это еще в лучшем случае: известно ведь, какие бывают озорные школьники.

Есть такая замечательная утилита!

Короче говоря, прав я или не прав, а хочу, чтобы на рабочих местах все было по возможности одинаково. И здесь прежде всего надо решить в принципе: буду ли я бороться с разного рода самодельностью, расставляя всевозможные преграды и защиты, или пойду по пути непротивления, смиряясь со своей обязанностью постоянно следить за всеми изменениями? Между прочим, второй вариант не так уж безнадежен, как может показаться, и стоит хорошенько подумать, прежде чем браться за оружие. Любое действие рождает противодействие, а потому наличие защиты обязательно активизирует поиск ее преодоления. Соперничество в этой области еще неизвестно чем закончится, не исключено, что и моим поражением. Кто знает: а вдруг ученик такое сильное программное средство найдет, которое не только все установленные мною уровни защиты преодолеет, но и свою собственную защиту поставит? Страшно представить: я, например, чтобы распечатать какой-нибудь текст, должен буду спрашивать разрешения у своих же учеников, если им вздумается в ответ на мои защитные меры поставить собственный пароль для использования принтера!

Наверное, мне лучше совсем не ввязываться в такую борьбу — полагаю, тогда в итоге будет затрачено меньше сил. Правда, придется все время восстанавливать исходные стандартные настройки разных машин, однако это только по началу, а расчет мой на то, что со временем интерес к неконструктивному переустройству сам собой постепенно сойдет на

нет, не встречая сопротивления. Ручаться, конечно, не могу, но по крайней мере моя практика до сих пор была такова.

Но это я такой терпеливый, да и обучаю я все-таки главным образом не детей, а взрослых, которые сами преподают информатику в школе. Судя по их живейшему интересу к данной проблеме, многие, безусловно, хотели бы взять курс на ограничения, и тут есть специальные средства. В частности, не так давно у меня появилась программа Guardian¹² (хранитель, страж, охранник), которая позволяет защитить буквально все, что пожелаешь. Программа эта небольшая (один файл размером 364 Кбайта), а в качестве значка у нее употребляется почему-то микросхема. Работает же “охранник” следующим образом. После записи программы на жесткий диск и запуска на выполнение в первый раз внизу, на панели задач, рядом с переключателем клавиатуры, отобразится маленький висячий замок.

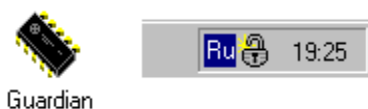


Рис. 9.1. Значок программы и ее эмблемка на панели задач

В дальнейшем этот замок станет автоматически навешиваться при каждом включении компьютера, поскольку все его охраняемые функции загружаются вместе с операционной системой. Какие же это функции? А вот сейчас узнаем — надо только сделать еще один двойной щелчок, но теперь уже не по значку программы, а по самому замочку. В ответ замок потребует ввести пароль на получение права работы с ним.



Рис. 9.2. Если бы еще и пояснений не было!

Сейчас паролем служит слово SHAREWARE — именно так, прописными буквами. Точнее, это вовсе и не пароль, а просто способ его преодолеть — собственно, пароль-то потом можно будет устанавливать, какой пожелаешь. Плохо только, что даже после его установки волшебное слово SHAREWARE все равно будет действовать, так как данная версия программы является свободно распространяемой. Об этом написано прямо в заставке, хотя и по-английски; так сказать, ключ поставляется в комплекте с замком. Следовало бы, конечно, раздобыть полноценную, коммерческую версию, естественно, заплатив за нее. Сумма, наверное, не очень большая, но денежными расчетами через Интернет мне пока заниматься не доводилось. Приходится работать с тем, что есть; да, впрочем, не каждый ведь и обратит внимание на то, что там написано¹³.

¹² Мой коллега “скачал” ее из сети Интернет. Адрес указан на рис. 8.7.

¹³ На худой конец можно эту надпись убрать: в простом текстовом редакторе загрузить файл GUARDIAN.EXE для редактирования, найти в нем фрагмент, начинающийся с “THIS IS A SHAREWARE VERSION...”, и заменить его какими-нибудь иными словами. Правда, редактор здесь подходит не любой, поскольку файл довольно велик, но, скажем, Volkov Commander с ним справляется.

Войдя, таким образом, с помощью пароля или волшебного слова в режим управления замочком, я получаю возможность настраивать охранника на подходящие мне запреты. Только шутить с ним не стоит, о чем заблаговременно предупреждает автор программы.



Рис. 9.3. Не уверен — не обгоняй!

Предлагается щелкнуть по кнопке **Yes**, только если уверен в себе, в своей квалификации, иначе можно таких ограничений понаставить, что и сам потом не разберешься, что к чему. В принципе, если быть осторожным, то бояться нечего, по крайней мере на данном этапе. Ведь совсем не обязательно что-то менять в настройках, которые будут предложены по умолчанию, а потом всегда есть возможность в самый последний момент сказать: **Отмена**. В конце концов волков бояться — в лес не ходить! Хочу просто посмотреть, что будет дальше.

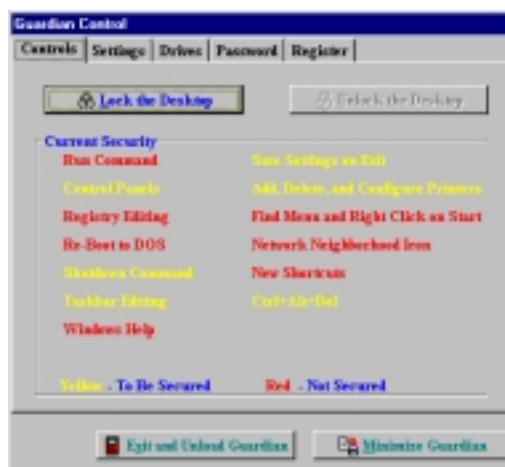


Рис. 9.4. Это ваш заказ?

А посмотреть есть на что. Карточка с пятью закладками, и первая, **Controls**, показывает, какие запреты я могу установить. Хоть и по-английски, а все понятно: одним цветом — что не будет засекречено, **Not Secured**, а другим — что будет, **To Be Secured**. Когда будет? А когда, пройдясь по всем закладкам, я наконец щелкну по экранной кнопке **Lock the Desktop**: мол, выполняй заказ. Вот тут уже спешить не надо. Как говорится, семь раз подумай, перед тем как вводить в действие такие режимы! Ну, пусть, к примеру, я соглашусь с тем, что будет засекречен пункт **Control Panels**, то есть **Панель управления**. Очень хорошо, никто тогда не сможет “войти” в эту самую панель, и не только для того, чтобы по-другому оформить **Рабочий стол** (фоновый рисунок, заставка и т.д.), а вообще чтобы хоть что-то изменить! Это касается и “мышинного” указателя, и звукового сопровождения, и свойства клавиатуры, и даже установки нового оборудования или

нового программного пакета, ввиду того что сам значок **Панели управления** исчезнет как из **Главного меню** (**Пуск — Настройка**), так и из окна **Моего компьютера**. Никак к ней не обратишься, а значит, и управлять не сможешь! Очень сильное ограничение, не правда ли?

Другой пример, пожалуйста. Если засекретить пункт **Shutdown Command**, то никто не сможет выключить компьютер как положено, через **Пуск — Завершение**

работы, только посредством **Ctrl**, **Alt**, **Del**¹⁴. Неизвестно, зачем это нужно, но тем не менее мое право решать. Пожалуй, все же такая строгость мне ни к чему и я исключу ее из своего выбора. Напротив, функцию **New Shortcuts** (новые ярлыки), которая пока еще не засекречена, мне бы хотелось заказать. Ну хотя бы ради пробы: смогу я новые ярлыки создавать или не смогу?

Но как ее заказать, ведь та закладка, где я сейчас нахожусь, не предусматривает смены статуса того или иного пункта, в ней только показано, что будет (или не будет) засекречено — когда щелкнешь по кнопке **Lock the Desktop**. То есть надо либо принять весь тот набор секретности, что предложен, либо нет. А чтобы сделать его другим, каким мне требуется, я должен перейти в следующую закладку, под названием **Settings** (установки), там указать, что нужно, а что не нужно, и только потом вернуться обратно в закладку **Controls**, но уже с новым набором, который затем и ввести в действие. Так и надо поступить.

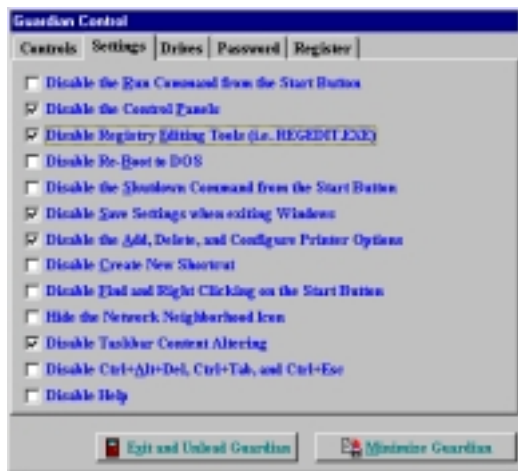


Рис. 9.5. Что надо — отметить

Перечислены те же самые пункты (сформулированные чуть более развернуто), но самое главное — здесь я могу против каждого поставить или, наоборот, снять галочку. Поставить **Disable** — значит “да, запретить” или “закрыть доступ”, снять — значит не запрещать и не закрывать.

Закрывать доступ можно:

- 1) к команде **Выполнить** из **Главного меню** (то есть через кнопку **Пуск**);
- 2) к **Панели управления**;
- 3) к средствам редактирования **Системного реестра**¹⁵;

¹⁴ Данную комбинацию, кстати, тоже можно засекретить. Правда, против **RESET** еще ничего, кажется, не придумано...

¹⁵ Системный реестр — страшная вещь. В двух его файлах, **SYSTEM.DAT** и **USER.DAT**, содержится буквально все настройки компьютера. Редактировать реестр вручную не рекомендуется, поэтому доступ к программе **REGEDIT.EXE** из папки **Windows** я бы обязательно закрыл. Между прочим, и сама-то рассматриваемая программа **GUARDIAN.EXE** — это ведь не что иное, как еще одно средство редактирования реестра!

4) к перезагрузке компьютера в режиме эмуляции **DOS**;

5) к команде **Завершение работы** из **Главного меню**. А вот что еще можно запретить:

6) сохранять настройки системы при выходе из **Windows**;

7) добавлять, удалять или конфигурировать принтеры¹⁶;

8) создавать новые ярлыки.

И еще закрыть доступ:

9) к команде **Поиск** из **Главного меню**¹⁷.

Далее: спрятать, скрыть (**Hide**):

10) значок **Сетевое окружение** на **Рабочем столе**.

И снова запретить:

11) настраивать на свой лад **Панель задач** через пункт **Настройка** в **Главном меню**;

12) использовать “горячие клавиши” **Windows**, типа

Ctrl, **Alt**, **Del**, **Ctrl**, **Tab**, **Ctrl**, **Esc** и т.п.;

13) получать помощь от справочной системы **Windows**.

Некоторые пункты сразу кажутся странными. Ну зачем, скажите пожалуйста, закрывать доступ к команде **Поиск** или, скажем, запрещать помощь справочной системы? Признаю, что бывает целесообразным запрет, касающийся **Панели управления** и **Панели задач**, “удаления” принтера, редактирования системного реестра, но зачем здесь команда **Выполнить**, эмуляция **DOS**, “горячие клавиши”? Не сразу я догадался, каково назначение этих пунктов, но потом понял.

Все дело в **Системном реестре**! Как уже отмечалось, это страшная вещь. Он представляет собой обширную базу данных, в которой хранятся записи обо всех без исключения настройках компьютера. Создается она автоматически, во время установки системы, и в дальнейшем там оперативно отображаются разнообразные изменения, вносимые пользователем в режим работы в связи с добавлением каких-либо аппаратных средств, инсталляцией той или иной программы, выбором понравившегося фоновой рисунка и т.п. Когда изменяют скорость перемещения мыши, устанавливают драйвер сетевой платы или заказывают, чтобы среди стандартных программ был, например, графический редактор **Paint**, то в каждом из указанных случаев редактируют **Системный реестр** (очень часто даже не подозревая об этом). Просто **Панель управления** является средством (а точнее, одним из средств) такого редактирования, причем высокоуровневым.

Среди языков программирования тоже есть удобные: **Бейсик**, **Паскаль**, **Си**, — а есть и ассемблер, и машинный код, дающие полную власть над машиной. На высоком уровне, когда я пользуюсь **Панелью управления**, сама операционная система оберегает меня от неверных действий, сообщает об ошибочном выборе, предлагает подходящие варианты. Но имеется и низкоуровневое средство редактирования реестра — программа **REGEDIT.EXE**, хранящаяся в папке **Windows**. Того же самого быстрого ли, медленного ли перемещения мыши я мог бы добиться и с его помощью, без всякой **Панели управления**, вручную сделав соответствующую запись в базе данных. Однако здесь уже нет никакого контроля, и если записать что-нибудь не то, вообще все может перестать работать.

Понятно, что реестр надо беречь от грубого вмешательства. Причем не так опасно, если на него будут воздействовать

¹⁶ ПечатаТЬ разрешается, только на изменение “свойств” принтера накладывается запрет.

¹⁷ И к аналогичной команде **Найти** из контекстного меню кнопки **Пуск**, вызываемого щелчком правой клавишей мыши.

через **Панель управления** или с помощью той же программы Guardian. Главное — перекрыть все пути к редактору REGEDIT.EXE. Потому-то, кроме специального пункта номер 3, и предложены другие, кажущиеся сперва ненужными. Действительно, пусть я прямой доступ к редактору закрою, но к нему, оказывается, удается добраться и через пункт **Выполнить**, и через **Поиск**, и через **Справку**, и даже по “горячим клавишам”. Правда, здесь необходимо еще такую задачу себе поставить (а, конечно, далеко не каждый школьник о подобных возможностях знает). Что касается меня, то, если уж выбирать запреты, я бы не стал слишком усердствовать и ограничился бы галочками во 2, 3, 6, 7 и 11-м пунктах, а остальные бы снял¹⁸.

После этого, возвратясь в закладку **Controls**, я увижу, что все отмеченное мною окрашено в желтый цвет — **To Be Secured**, то есть будет засекречено, надо только щелкнуть по кнопке **Lock the Desktop**. Однако щелкать по ней еще рано, так как и на других закладках что-нибудь может привлечь мое внимание. Вот, скажем, **Drives**-устройства.



Рис. 9.6. Как много разных дисков!

Тут я снова расставляю или снимаю галочки, теперь уже напротив всех дисков, имеющихся на моей машине. Допустим, первая галочка означает, что из окна **Мой компьютер** исчезнет значок дисководов А и, таким образом, доступ к дисководу будет закрыт — нельзя будет ничего ни записать на дискету, ни прочитать с нее. Будто бы у меня его вообще нет, накопителя на гибких дисках. Хотя, конечно, это и самому неудобно, но вдруг для какого-нибудь класса данная мера — единственный способ уберечься от вирусов или иных нежелательных программ (тех, что ученики имеют обыкновение приносить из дома). Ну а когда мне понадобится произвести запись на дискету, я могу еще раз вызвать “охранника” Guardian, замочек-то всегда под рукой, на **Панели задач**, и перестроить его на время, снять запрет на дисковод А.

Каждый решает, как ему удобнее. Вот я насчет дисководов А еще бы подумал, а жесткий диск Е, где у меня хранится дистрибутив Windows 95, конечно, закрыл. Дистрибутив есть дистрибутив, он всегда должен быть в сохранности, ведь иногда и систему приходится переустанавливать, поэтому лучше, когда на диске Е никто ничего не трогает. Другие жесткие диски закрывать, очевидно, нельзя, иначе как же с ними работать? Впрочем, у моего винчестера есть особенность: два его

¹⁸ А для страховки просто скопировал бы сам реестр, оба его файла, SYSTEM.DAT и USER.DAT, на дискеты и хранил бы их отдельно, на случай, если какой-нибудь умник все-таки найдет окольный путь к REGEDIT.EXE.

логических раздела, а именно С и D, уплотнены с помощью программы DriveSpace, из-за чего на закладке ярко выделены дополнительно еще две буквы — G и H — это так называемые несущие диски для уплотненных С и D, их, разумеется, тоже надо скрыть. Вдобавок слегка проглядывает буква F для привода CD-ROM, но у данной машины такое устройство отсутствует физически. А если бы и присутствовало, все равно не стал бы его засекречивать, снял бы галочку.

Идем дальше. Закладка **Password** с паролями. Здесь никаких трудностей нет, потому и рисунок можно не приводить. Все как обычно: надо ввести действующий пароль, SHAREWARE, потом новый (с подтверждением) и щелкнуть по кнопке **Change Password**. Замечу только, что при этом строчные и прописные буквы не являются эквивалентными, то есть **f** и **F** воспринимаются как разные буквы. Смена пароля, безусловно, не помешает, жаль, повторяю, что и старый всегда позволяет “открыть” программу, но что поделаешь. Вот если бы официально зарегистрироваться. Для того и последняя закладка (**Register**) предусмотрена. На ней, кстати, адрес автора в сети Интернет указан. Кто имеет кредитную карточку, почему бы не зарегистрироваться?



Рис. 9.7. Может быть, действительно зарегистрироваться?

Ну а мне пора переходить от намерений к делу и щелкнуть наконец по кнопке **Lock the Desktop** с первой закладки. Хотя не поздно еще передумать: там же имеется другая кнопка — **Exit and Unload Guardian** (“выйти из программы и выгрузить ее из памяти”), и если ее использовать, все останется как прежде, без запретов. Даже замочек исчезнет с панели задач и при последующих включениях компьютера появляться не будет. Пока не придет искушение еще раз запустить Guardian.

Все работает, но есть и возражения

Откровенно скажу: первый раз было страшновато приступать к столь ответственному делу. А если я что-нибудь не так понял или вдруг окажется, что программа со сбоями, раз она *shareware*? Потом очень нелегко все восстанавливать. Но в конце концов, подумал я, компьютер в классе не один, в крайнем случае использую для загрузки системную дискету¹⁹, отформатирую диск С и установлю систему заново, дистрибутив-то есть. Была не была!

¹⁹ Даже если доступ к дисководу А будет закрыт, поскольку запрет вступает в силу лишь после загрузки системы с винчестера, а я стану осуществлять загрузку не с него, а с дискеты.

Итак, я решился “нажать” эту опасную кнопку **Lock the Desktop**. Что же произошло? Как развивались события дальше? Во-первых, сразу появилось сообщение, предлагающее перезагрузить систему. Я ответил согласием, и через несколько секунд на экране снова возник обычный **Рабочий стол**, а внизу по панели задач поползла по направлению к кнопке **Пуск** надпись “Начните работу с нажатия этой кнопки”. Причем вопреки моему ожиданию внизу справа не появилось никакого замочка и, более того, даже переключателя клавиатуры почему-то не было видно. Меня это, естественно, насторожило. Как же мне теперь управлять охранником, если потребуется снять запрет на что-либо? И потом, с русского на английский как же теперь переходить? Да, непорядок. Надо перезагрузить машину полностью, по-настоящему.

Перезагружаюсь (хорошо еще запрет на **Завершение работы** не установил!) и вздыхаю с облегчением: замочек появился и переключатель клавиатуры тоже, — кажется, все в порядке. Первым делом к замочку и обращаюсь: а нельзя ли, мол, назад? Ну, сначала, конечно, введи пароль (SHAREWARE или какой ты там установил), а после пароля — почему же и нет, можно и назад, можно и вперед!

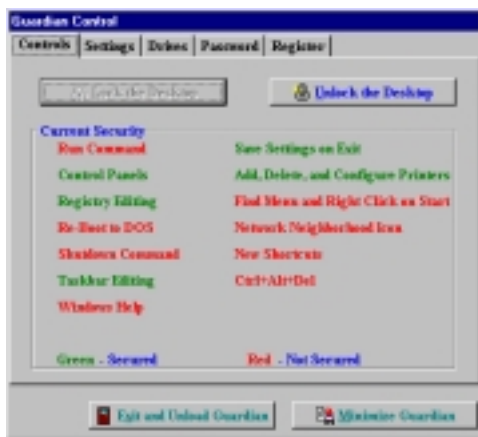


Рис. 9.8. Сам этого хотел

Именно те самые режимы секретности, какие заказывались, только выделенные теперь уже не желтым, а зеленым цветом, и не **To Be Secured**, а просто **Secured**. То есть не “должно быть засекречено”, а уже засекречено. Да еще активная экранная кнопка другая, не **Lock the Desktop**, а **Unlock the Desktop** — не установить запреты, а наоборот, снять. Зачем снять? А чтобы, если потребуется, установить другие, новые. Новые установишь — и опять тогда **Lock the Desktop**. Все в моих руках и прекрасно работает — пробовал.

Вот, например, какие сообщения выдаются на экран, если захочешь испытать защиту. Допустим, поменять фоновый рисунок или поставить какую-нибудь экранную заставку. Щелкаю по привычке правой клавишей мышки по рабочему столу и выбираю нижнюю строку **Свойства** в контекстном меню. А мне в ответ — карточка:

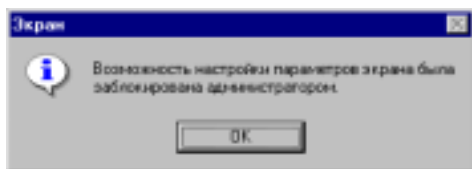


Рис. 9.9. Никак нельзя!

Хорошо, а если попробовать через **Панель управления** в **Главном меню**? Щелкаю **Пуск**, но пункта **Настройка** там нет вообще. Что заказывал, то и получил. То же самое и с дисками. Открываешь **Мой компьютер** — и видишь только лишь С и D, а с дискеты ничего считать не можешь. Даже если до введения запрета был уже создан на рабочем столе ярлык, ведущий к диску А, то толку от него никакого:

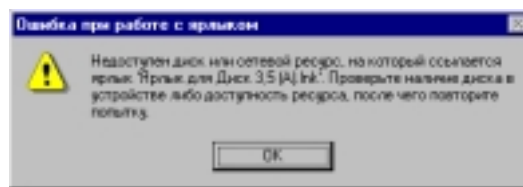


Рис. 9.10. Считаю, что гибкого диска у тебя нет

Ну и так далее. На каждое запрещенное действие — отказ. Впрочем, пылливому уму не так легко поставить непреодолимую преграду. Фоновый рисунок все-таки можно поменять. Просто я вспомнил, что когда работаешь в графическом редакторе Paint, то, загрузив какой-либо рисунок из файла BMP или даже нарисовав свой собственный, имеешь возможность вывести его на рабочий стол: в меню **Файл** есть две строчки — **Заполнить рабочий стол Windows** (используется, если рисунок на весь экран) и **В центр рабочего стола Windows** (употребляется, если рисунок маленький). Вот тебе и запрет!

Такая же история и с диском А. Да, его не видно из среды Windows, однако стоит выйти в Нортон — и пожалуйста: все диски перед тобой. Можно, конечно, и к Нортону путь закрыть, поместив папку с ним на скрытый диск Е, но тогда ведь и самому неудобно будет работать. Получается, что защита есть, но она не стопроцентная. Как я уже напоминал, действие рождает противодействие. На каждую меру найдется контрмера, каждый запрет будут пытаться обойти. Думай, учитель, вступать ли на этот путь борьбы.

Короче говоря, советую попробовать, ну а если не понравится и захочется отказаться от услуг охранника, то ведь кнопка **Exit and Unload Guardian** всегда наготове. Стоит по ней щелкнуть — и действительно все запреты снимаются и сам замочек больше не появляется. В общем, все ОК, пользоваться можно! Вот только нужно ли, все одолевают меня сомнения.

В заключение замечу, что подобных программ-утилит для Windows 95 написано разными авторами довольно много и предназначены они для самых разных целей, а не только для наложения запретов²⁰. Это просто мне под руку попалась утилита Guardian, а так как в предыдущей статье шла речь о правах доступа к ресурсам локальной сети, я и стал рассказывать о защите своего собственного рабочего места, и думаю, что не зря.

Ну а серия статей о модернизации моего компьютерного класса выходит на финишную прямую, причем финиш этот, вообще говоря, является промежуточным, поскольку процесс модернизации может продолжаться бесконечно.

²⁰ Привожу два адреса в сети Интернет для тех, кто хочет “скачать” что-нибудь подходящее: <http://www.winfiles.com/appa/98/> и <http://www.freeware.ru> (особенно рекомендую второй — там все бесплатное и на русском языке).

Театр уж полон...
А.С. Пушкин

В драматическом театре им. Пушкина к юбилею Александра Сергеевича решили поставить оперу “Евгений Онегин”. Артисты театра обладают красивыми, но не очень сильными голосами. По этой причине руководство театра дало указание приобрести радиомикрофоны.

В начале и в конце спектакля все артисты находятся за кулисами. Артисты выходят на сцену и покидают ее через правую или левую кулису. Для того чтобы петь на сцене, артист берет с собой один микрофон. Артист может выходить на сцену с микрофоном (одним), даже если ему не надо петь в этом выходе. Взяв микрофон, артист не может оставить его на сцене или передать другому артисту. При уходе артиста за кулисы микрофон остается за соответствующей кулисой до тех пор, пока его снова не возьмет какой-либо артист, выходящий на сцену.

Очередность выходов артистов на сцену и их уходов за кулисы указывается в режиссерском плане. Кроме того, в этом плане указывается, через какие кулисы выходит (или уходит) артист и поет ли он в данном выходе.

Написать программу, которая по заданному режиссерскому плану определяет минимальное количество требуемых для постановки оперы микрофонов, их начальное размещение по кулисам и для каждого выхода указывает, брать или не брать микрофон.

Формат входных данных

Первая строка входного файла содержит натуральное число N — количество артистов, участвующих в спектакле ($1 \leq N \leq 1000$).

Во второй строке записано натуральное число K — количество выходов артистов на сцену ($1 \leq K \leq 3000$).

Далее идут $2K$ строк, описывающих режиссерский план спектакля. Каждая из них содержит четверку $A_i B_i C_i D_i$ ($1 \leq i \leq 2K$):

- A_i — символ +, если в данный момент артист выходит на сцену, или символ —, если артист со сцены уходит;
- B_i — номер артиста (целое число от 1 до N);
- C_i — символ Л, если артист выходит (уходит) через левые кулисы, или символ П, если он выходит (уходит) через правые кулисы;
- D_i — символ Д, если артист поет в данном выходе (пел перед данным уходом), или символ Н, если он не поет (не пел).

Формат выходных данных

Первая строка выходного файла должна содержать два целых числа. Первое число — количество микрофонов перед началом оперы с левой стороны, второе число — количество микрофонов с правой стороны. В каждой из последующих K строк необходимо вывести 1 или 0 в зависимости от того, берет ли с собой микрофон очередной выходящий на сцену артист (1 — берет, 0 — не берет).

Пример входного файла

```
3
4
+ 1 Л Д
- 1 Л Д
+ 2 Л Н
+ 3 Л Н
- 3 П Н
+ 1 П Д
- 1 Л Д
- 2 П Н
```

Пример выходного файла

```
1 0
1
0
1
1
```

Задача 6. Стихоплет

имя входного файла:	input.txt
имя выходного файла:	output.txt
ограничение времени на каждом тесте:	10 с
баллы:	35

*Четырехстопный ямб мне надоел:
Им пишет всякий. Мальчикам в забаву
Пора б его оставить. Я хотел
Давным-давно приняться за октаву.
А в самом деле: я бы совладел
С тройным созвучием. Пуцусь на славу!
Ведь рифмы запросто со мной живут;
Две придут сами, третью приведут.*
А.С. Пушкин

Любому стихотворению можно поставить в соответствие его схему по следующим правилам. Каждый ударный слог заменяется на символ апостроф “’”, а безударный — на символ тильда “~”. В схеме сохраняется пунктуация (знаки препинания: “,”; “;”, “:”, “.”, “?”, “!”). Строки схемы, соответствующие *рифмующимся* строкам стихотворения, помечаются одинаковыми числами. Для любого стихотворения его схема строится однозначно. Ниже представлен пример четверостишия и его схемы:

```
Мой дядя самых честных правил,   4 ~'~'~'~'~'~',
Когда не в шутку занемог,       1 ~'~'~'~'~'~',
Он уважать себя заставил         4 ~~~'~'~'~'~',
И лучше выдумать не мог.         1 ~'~'~'~'~'~'.
```

Будем считать, что в стихотворении действует следующее правило рифмовки: две строки *рифмуются*, когда в них совпадают все буквы, от последней ударной гласной до последней буквы в строке. Если же ударная гласная — последняя буква в строке, то для *рифмы* необходимо совпадение двух последних букв рифмующихся строк. При этом рифма может порождаться как частью одного слова, так и несколькими подряд идущими словами. Например:

*А подбирать союзы да наречья;
.....
Мне рифмы нужны; все готов сберечь я.*

Отдельные строки стихотворения могут как рифмоваться со всеми строками стихотворения, так и вообще не иметь рифмующихся с ними строк.

Пусть нам известны схема стихотворения, список слов в произвольном порядке каждой отдельной его строки и в многосложных словах указаны ударные слоги. Односложные слова в стихотворениях могут быть как ударными, так и безударными, что отражается только в схеме стихотворения.

Требуется написать программу, переставляющую слова в списке слов каждой строки так, чтобы они соответствовали схеме стихотворения. Считается, что хотя бы одно соответствие списка слов и схемы существует.

Формат входного файла

В первой строке входного файла находится целое число N — количество строк в стихотворении ($1 \leq N \leq 20$). В следующих N строках расположены в произвольном порядке слова соответствующих строк стихотворения. В каждой из них содержится не более 20 слов, разделенных пробелами. Под словом понимается последовательность строчных русских букв (за исключением “ё”) и, при необходимости, знака дефис “-”. Длина слова не превышает 30 символов. Если слово состоит более чем из одного слога, то перед ударной гласной стоит символ “’”.

Далее в N строках располагается схема стихотворения. Каждая строка начинается с числового идентификатора, являющегося произвольным натуральным числом, не превосходящим N (рифмующимся между собой строкам соответствует одно и то же число). За числовым идентификатором через один пробел рас-

положены условные обозначения слогов и знаки препинания соответствующей стихотворной строки оригинала.

Формат выходного файла

Выходной файл должен содержать любой вариант восстановленного стихотворения с сохранением пунктуации. Слова одной строки должны быть отделены друг от друга одним пробелом. Между знаком препинания и предшествующим ему словом должен отсутствовать пробел.

Пример входного файла

```
4
д'ядя мой пр'авил с'амых ч'естных
в занем'ог когд а не ш'утку
заст'авил он себ'я уваж'ать
в'ыдумать и л'учше мог не
4 ~'~'~'~'~,
1 ~'~'~'~'~,
4 ~~~'~'~'~'
1 ~'~'~'~'~'.
```

Пример выходного файла

```
мой самых дядя честных правил,
в когда не шутку занемог,
он уважать себя заставил
не лучше выдумать и мог.
```

Примечание

Оцениваются и такие решения, в которых соблюдены порядок чередования ударных/безударных слогов и пунктуация оригинала, а рифмы нет.

Журнал “Компьютерные инструменты в образовании” совместно с Санкт-петербургским государственным университетом продолжает прием школьников в ЗАОЧНУЮ ШКОЛУ СОВРЕМЕННОГО ПРОГРАММИРОВАНИЯ

В январе 1999 года при издаваемом в Санкт-Петербурге под эгидой Российской академии образования журнале “Компьютерные инструменты в образовании” начала работу Заочная школа современного программирования для учащихся 5—11-х классов, организованная журналом совместно с Санкт-петербургским государственным университетом.

Поступить в школу могут учащиеся 5—11-х классов, как те, кто уже программирует, так и те, кто не имеет о языках программирования никакого представления, но интересуется профессией программиста и готов решать содержательные задачи и читать статьи по современным проблемам информатики.

Прием в школу производится постоянно. Всем учащимся высылаются задания в виде отдельных книжечек.

Задачи имеют несколько уровней:

1) первый уровень рассчитан на тех, кто не изучает алгоритмические языки; решение этих заданий записывается в виде текстового файла в формате ASCII (точная форма записи ответа указывается в условии задачи);

2) второй уровень предназначен для умеющих программировать; решения заданий второго уровня представляются в виде исполнимых exe-файлов (точный формат входных и выходных данных указывается в условии задачи).

Решения в бумажном варианте, на дискетах или электронной почтой пересылаются в школу, где они проверяются и рецензируются. В журнале публикуется подробный разбор всех задач и приводятся лучшие решения. Учениками школы могут быть как отдельные школьники, так и коллективы (классы, кружки и т.д.).

Стоимость обучения в школе — 80 руб. в год, для подписчиков журнала — бесплатно.

Для поступления в школу нужно прислать заявку по адресу: **191025, Санкт-Петербург, ул. Марата, д. 25**

E-mail: pozdnkov@aec.neva.ru

Журнал “Компьютерные инструменты в образовании”. Школа современного программирования.

Телефон для справок: (812) 164-13-55.

Форма заявки:

Прошу принять меня в Школу современного программирования. О себе сообщаю следующее:

1. Фамилия, имя, отчество. Возраст.
2. Место учебы: название и адрес учебного заведения, класс.
3. Фамилии, имена и отчества преподавателей математики и информатики.
4. Домашний адрес, телефон, адрес электронной почты (если есть).
5. Являюсь/не являюсь подписчиком журнала.
6. Личная подпись.

Выравнивающе-развивающая методика преподавания информатики

И.Н. Фалина,

старший преподаватель кафедры информатики СУНЦ МГУ,
Москва

Методика преподавания информатики, о которой пойдет речь в этой статье, родилась в стенах школы-интерната имени А.Н. Колмогорова. Наша школа-интернат была основана в 1963 году выдающимся ученым-математиком Андреем Николаевичем Колмогоровым с целью поиска и поддержки одаренных детей с российской периферии. В 1988 году на базе школы-интерната было создано самостоятельное подразделение МГУ имени М.В. Ломоносова — Специализированный учебно-научный центр Московского государственного университета — СУНЦ МГУ.

Ежегодно СУНЦ МГУ проводит набор в 10-е классы физико-математического, компьютерно-информационного, биофизического и химического профилей (двухгодичное обучение) и в 11-е классы физико-математического профиля (одногодичное обучение).

Специфика набора в профильные классы такова, что дети с одинаковым достаточно высоким уровнем знаний по физике и математике имеют разный уровень подготовки по информатике (с аналогичной проблемой сталкиваются преподаватели информатики в вузах да и во многих школах). Эта особенность только усиливается, если школьники поступают в такие классы из разных регионов.

У школьников, поступивших в СУНЦ, разброс уровня знаний по информатике в силу всем известных причин огромен: к нам поступают и победители олимпиад по информатике, и ребята, у которых отсутствуют минимально приемлемые знания. Среднестатистическая оценка уровня знаний школьников, поступающих в СУНЦ МГУ, по информатике (на основе пятилетнего анкетирования) такова:

- 20—25% — не изучали информатику;
- 11% — плохой уровень знаний;
- 28% — средний;
- 23% — хороший;
- 13—18% — очень хороший.

Таким образом, ежегодно мы начинаем преподавание в классах с различным уровнем знаний по информатике. При этом, учитывая высокий потенциал наших школьников и задачи создания профильных классов, мы читаем в нашей школе достаточно сложный курс информатики. Стержень курса, на который нанизывается все содержание, можно выразить словами Манфреда Броя: “Информатика — это фундаментальная наука, которая занимается вопросами представления и обработки информации”, — добавим: машин-

ной обработки. На преподавание информатики в СУНЦ МГУ отводится 3 часа: 1 час — лекция, 2 часа — семинары (практические занятия).

В результате многолетней работы в СУНЦ МГУ была разработана методика преподавания курса информатики в профильных классах с различными стартовыми знаниями учащихся по предмету. Эту методику мы назвали **выравнивающе-развивающей**. Наша методика во многом переключается с методикой полного усвоения М.В. Кларина (технология полного усвоения), но, естественно, имеет и свои отличия. Мы назвали свою методику **выравнивающей** потому, что она позволяет за один учебный год выровнять уровень знаний школьников по базовому курсу, при этом наша методика позволяет поддерживать достаточно высокий уровень интереса к изучаемому материалу у школьников, имеющих высокие стартовые позиции в информатике. Именно поэтому мы назвали нашу методику **развивающей**.

В статье приводятся конкретные примеры применения методики. Для удобства изложения ниже перечислены основные блоки курса информатики для 10—11-х физико-математических классов СУНЦ МГУ. Основой курса “Структуры данных и алгоритмы” является связка: “структура данных — действия над этой структурой”. На протяжении всего курса идет движение по усложняющимся цепочкам: абстрактная структура данных \Rightarrow алгоритмы обработки этой структуры \Rightarrow компьютерная реализация этой структуры \Rightarrow реализация алгоритмов обработки (см. таблицу на стр. 12).

В рамках курса мы изучаем алгоритмический язык Турбо Паскаль, но обучение программированию — это вспомогательная линия курса. Основная — представление информации в виде абстрактных структур и машинная обработка этих структур.

Область применения методики: классы учеников или группы студентов, имеющих различный стартовый уровень по информатике.

Период применения: год или один семестр, в течение которого происходит выравнивание базового уровня по информатике.

Результаты применения методики:

- 1) достаточно высокий результат усвоения в пределах требований к обязательным результатам обучения;
- 2) за пределами этих обязательных требований, конечно же, мы имеем различие в учебных результатах;
- 3) при освоении обязательного базового курса у всех учеников постоянно поддерживается интерес к изучаемому материалу;

Структуры данных	Реализация в ЭВМ	Основные действия над структурой, алгоритмы
Целые числа	Представление чисел с фиксированной запятой	Изучение систем счисления, целочисленная арифметика в ограниченном числе разрядов
Логические величины	Булева алгебра логики	Составление сложных логических выражений, решение логических задач, изучение конструкций If, Case
Вещественные числа	Представление чисел с плавающей запятой	Вещественная компьютерная арифметика, изучение операторов цикла, численное решение трансцендентных уравнений ($\cos x = x$) с использованием метода деления отрезка пополам, вычисление площади с использованием метода Монте-Карло и т.д.
Последовательность символов (чисел, букв, сложных объектов)	Одномерный массив	Алгоритмы поиска и сортировки, изучение структур типа Record, объявление процедур и функций
Таблицы	Двухмерный массив	Работа с двухмерными матрицами, описание “лабиринтов”, разработка игр в текстовом режиме работы монитора
Строка текста	Строки, одномерный массив символов	Алгоритмы обработки строк
Тексты	Текстовый файл	Изучение принципов построения текстовых редакторов, алгоритмы обработки текстов, работа с текстовыми файлами
Базы данных	Типизированный файл	Изучение стандартных СУБД, написание своих СУБД, работа с типизированными файлами
Графические изображения	Бинарный файл	Работа в графическом режиме, изучение основных алгоритмов графики, создание мультфильмов
Списки, очереди, стеки, деревья, графы	Реализация через массивы и через указатели (динамические структуры данных)	Изучение динамических структур данных, основные алгоритмы обработки простейших структур, например, связанных однонаправленных списков

4) преподаватель имеет возможность работать в режиме “индивидуального подхода” практически на одном и том же методическом материале.

Основные постулаты внедрения методики:

1) все ученики **могут и должны** освоить учебный материал полностью;

2) каждому ученику требуются свой **темп** обучения и **доступные для него способ и глубина** изложения материала. Способности ученика определяются его темпом учения не при усредненных, а при оптимально подобранных для данного ученика условиях. По наблюдениям Б.Блума, 5% учеников (талантливые) могут учиться в высоком темпе; для 90% учеников (обычные) уровень усвоения знаний и умений определяется затратами учебного времени;

3) должны быть выработаны **четкие критерии полного усвоения** темы, раздела, курса. Эти критерии вытекают из целей курса, из осознания планируемых результатов обучения, они должны быть сформулированы в виде **перечня умений, не допускающих расширенного или неоднозначного толкования;**

4) все учащиеся должны быть ознакомлены с основными принципами методики, по которой им предсто-

ит учиться. Таким образом, мы каждого ученика делаем активным участником процесса обучения.

Принципы, на которых основана выравнивающая-развивающая методика:

1) весь учебный курс разбит на блоки, каждый блок посвящен одной теме (содержательная целостность) и невелик по объему (2—4 урока);

2) изучение каждого блока строится с использованием следующих элементов учебного процесса: лекция, практическое занятие, самостоятельная работа учащихся, проверка знаний;

3) к каждому учебному блоку разрабатываются практические задания (практикумы) с задачами различного уровня сложности;

4) к каждой задаче разрабатывается система тестов;

5) к каждому блоку в целом разрабатывается система проверки знаний и умений;

6) вариативность практических заданий: ученику предоставляется возможность в рамках изучения темы выбрать наиболее интересное (полезное) для него задание;

7) использование терминологии, доступной уровню образования, принятой в данной дисциплине и кор-

релирующей с аналогичными терминами в смежных дисциплинах.

Рассмотрим каждый принцип более подробно.

Первый принцип (деление на блоки):

В начале изучения каждого блока ученику ставятся цели, которые он должен достичь. На каждый блок отводится 1—2 лекции для изложения теоретического материала и 2—4 урока практических (семинарских) занятий. Расстояние “стимул — реакция” должно быть по возможности малым для того, чтобы ученик держал в поле своей деятельности планируемый, ожидаемый результат. Из нашего опыта преподавания мы можем поделиться таким наблюдением: в конце 11-го класса ученики выполняют курсовые работы, рассчитанные на 1—2 месяца, и довольно часто мы сталкиваемся с неумением учеников организовать свое время для выполнения поставленной задачи, что приводит к неудовлетворительным результатам. Поэтому для усвоения **базового курса** небольшие по объему блоки программы предпочтительны.

Второй принцип (организация учебного процесса):

Изучение каждого блока строится с использованием следующих элементов учебного процесса: лекция — практическое занятие — самостоятельная работа учащихся — проверка знаний и умений.

Лекция читается для всего потока (нескольких классов), на лекции излагаются основные теоретические вопросы. В нашей школе с содержанием лекций ученик может ознакомиться в школьной компьютерной сети.

На **практических занятиях** разбираются основные вопросы, изложенные на лекциях. Рассматриваются типовые способы решения задач по темам. На этих же занятиях каждому школьнику дается задание для самостоятельной работы (теоретическая задача, написание программы, подготовка доклада). Цель практических занятий — формирование умений применять новые знания на практике.

Самостоятельная работа является необходимым элементом успешного применения методики. Под самостоятельной работой мы подразумеваем не только выполнение домашнего задания. Преподаватель должен создать условия для **возникновения побудительных мотивов** к самостоятельному изучению учащимися вопросов, изучаемых на лекциях и семинарах.

Проверка и контроль. Мы стараемся использовать разнообразные способы контроля степени усвоения основных понятий и ведущих идей, сформированности умений и навыков работы, но в любом случае мы придерживаемся следующих принципов:

1) **немедлительное сообщение результатов** проверки;

2) ни одно задание не должно быть оставлено без проверки и оценивания со стороны преподавателя.

Третий принцип (продуманные задачи):

Для каждого блока мы подбираем задачи, которые объединяются в практикумы.

1) В каждом практикуме есть задачи разного уровня сложности;

2) В каждом практикуме (особенно в начальных) есть задачи, для решения которых не требуется хоро-

ших навыков работы с компилятором, текстовым редактором. Эти задачи рассчитаны на понимание рассматриваемых структур данных и способов их обработки, т.е. они являются базовыми. Для решения таких задач требуются время, ручка и бумага. Когда задача решена, текст программы на Паскале занимает всего несколько строк, и, следовательно, отсутствие навыков работы на ЭВМ и незнание конкретного языка программирования не будут определяющими. Кроме того, при успешном решении такой задачи у начинающего ученика появляется **дополнительная мотивация** изучения предмета: “**Я не самый плохой, я могу**”, — т.к. ученик видит, что он решил ту же задачу, что и “продвинутый в информатике” товарищ по классу.

Пример. В практикуме № 1 “Целочисленная арифметика” есть задачи такого типа: *Ввести с клавиатуры два целых числа $m, n > 0$. Если m делится на n или n делится на m , то вывести 1, иначе — любое другое число. При выполнении задания нельзя использовать условные операторы и операторы цикла; необходимо оставаться в классе целых типов; можно использовать операции целочисленной арифметики MOD, DIV, *, +, - .*

Возможное решение

```
Var m, n, c:integer;
Begin
  write ('Введите два натуральных числа m и n ');
  readln(m, n);
  c:=(m mod n)*(n mod m)+1;
  writeln('Ответ:', c);
End.
```

Для школьников, которые уверенно себя чувствуют в программировании на языке Турбо Паскаль, мы даем эту же задачу, но снимаем ограничение “больше нуля”. В этом случае задача становится значительно сложнее, т.к. возможно возникновение ошибки типа “деление на ноль”. Кроме того, если значение только одного из введенных чисел равно 0, то результат должен быть 1 (0 делится на любое другое число), а если оба введенных числа равны 0, то ответ должен быть иным (0 на 0 не делится). Для того чтобы избежать при делении от 0, можно использовать следующее выражение: $1 \div (1+|n|)$. Это выражение равно 1 при $n=0$ и 0 в любом другом случае. В этом случае решение задачи имеет следующий вид:

```
Var m, n, m0, n0, c:integer;
Begin
  write ('Введите два натуральных числа m и n ');
  readln(m, n);
  n0:=1 div (1+abs(n));
  m0:=1 div (1+abs(m));
  c:=(m mod (n+n0))*(n mod (m+m0))+n0+m0;
  writeln('Ответ:', c);
End.
```

3) Задачи подбираются таким образом, чтобы наработки предыдущих практикумов можно было использовать как заготовки для решения задач последующих практикумов. Это позволяет сокращать время на написание и отладку программы (особенно для начинающих), позволяет концентрировать внимание на разбираемой теме, не тратя значительного времени на технику программирования.

Пример. В практикум № 2 “Булева алгебра логики” включена такая задача.

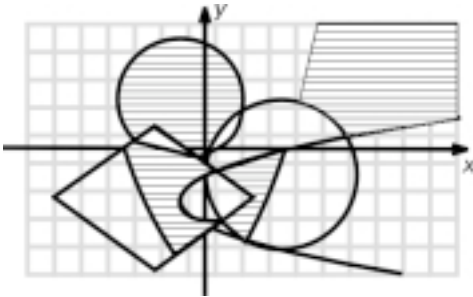
На координатной плоскости изображены линии, описываемые уравнениями окружности, прямой, параболы, ромба, прямоугольника. Несколько областей, ограниченных этими линиями, заштрихованы. Определить, принадлежит ли точка с координатами (x, y) заштрихованной фигуре. Точки, лежащие на границе, не учитывать (для простоты решения их можно считать одновременно как принадлежащими заштрихованной области, так и не принадлежащими ей). Значения x и y вводятся с клавиатуры.

Ответ выдать в следующем виде: True, если точка с координатами (x, y) принадлежит заштрихованной фигуре, False — в противном случае.

Обязательное требование:

- 1) операторы If и Case использовать нельзя;
- 2) для описания каждой области ввести свою булеву переменную (это упрощает процесс отладки и поиска ошибок); основное логическое выражение конструировать только из таких переменных;
- 3) имена переменных должны быть mnemonic, например, In_Circle1 (внутри первой окружности), Upper_Line1 (над первой прямой).

Для этой задачи каждому школьнику дается индивидуальный рисунок, аналогичный приведенному ниже.



А в практикум № 4 “Вещественная арифметика” включена задача: Вычислить площадь заштрихованной фигуры, используя метод Монте-Карло. Рисунок с заштрихованной фигурой взят из практикума № 2. При таком методическом подходе есть возможность сконцентрировать внимания на сущности метода Монте-Карло, не обращая внимания на способ определения попадания точки в требуемую область.

Четвертый принцип (продуманные тесты):

1) Для каждой задачи разработана система тестов. Мы этот принцип называем “каждому уровню знаний — свой тест на “отлично”. Если начинающему ученику ставится цель — решить задачу правильно в целом, то “продвинутому” ученику ставится цель — написать программу так, чтобы она прошла на всех тестах. В последнем случае мы учим школьников уже технике программирования, прививаем культуру программирования, т.е. срабатывает “развивающая” составляющая методики. Таким образом, **на одной и той же задаче мы получаем различные педагогические результаты:** начинающему показываем, что он решает ту же задачу, что и “продвинутый”; интересующемуся даем толчок к повышению умений и навыков; “зазнайке” показываем, что и “простую” задачу надо решать аккуратно, при этом она может стать не такой уж и легкой в реализации.

2) Предлагаемая система продуманных тестов **наиболее комфортна для преподавателя**, т.к. в рамках одной темы можно дать всего лишь 4—5 различных задач, но при этом учесть все уровни подготовки учеников за счет проверки заданий на различных тестах. Заметим, что подготовка тестов к каждой задаче занимает существенно больше времени, чем подбор самих задач.

3) Система тестов должна быть **не констатирующей, а развивающей**.

Пример. В практикуме № 1 “Целочисленная арифметика” есть задача: Для введенных с клавиатуры целых чисел m и n вывести на экран результаты всех возможных операций над этими числами.

При проверке этого задания мы обязательно даем такой тест: $m = 100$, $n = 40$ (допустим, что m , n относятся к типу byte) — и просим объяснить полученный результат.

В идеале от ученика мы хотим услышать, что целые числа m и n в компьютере занимают один байт и имеют следующее машинное представление:

$m =$	0	1	1	0	0	1	0	0
$n =$	0	0	1	0	1	0	0	0

При сложении этих чисел в старшем разряде получается единица, что соответствует отрицательному числу. Целые отрицательные числа представляются в дополнительном коде. Далее, при необходимости, ученика можно попросить восстановить десятичное представление отрицательного числа по его дополнительному коду.

Пятый принцип (“проверяй, развивая и не подавляя”):

Для каждого блока формируется своя система контроля знаний. Но при любом контроле знаний мы придерживаемся **принципа незамедлительного сообщения** ученику результатов оценки его знаний, умений или навыков.

1) Для взаимосвязанных блоков мы проводим **теоретическое тестирование на компьютерах**. Известно, что тестирование относится к виду контроля, при котором скорее выявляется, какие темы ученик не знает, чем глубина знаний по тем вопросам, на которые он ответил правильно. Составление вопросов для тестирования и, главное, вариантов возможных ответов требует тщательного продумывания. Необходимо учитывать и случайность выбора правильного ответа. При всех недостатках тестирования мы считаем возможным и необходимым использовать этот вид контроля, т.к. выполнение теста из 5—10 вопросов занимает 15—20 минут и преподаватель и ученик получают оценку моментально после окончания тестирования, кроме того, соблюдается объективность оценки. На кафедре информатики СУНЦ МГУ разработана программная оболочка для тестирования. Для всех блоков курса разработана система тестов. Нашу программную оболочку для тестирования можно использовать для любой школьной дисциплины.

2) В контрольные работы мы обязательно включаем **“развивающие” вопросы и задачи**, т.е. такие, о которых непосредственно не рассказывалось на лекциях и семинарах, но для решения которых изложенных сведений достаточно. Например, в контрольную работу по системам счисления включается вопрос о переводе периодической дроби из одной системы счисления в

другую. Для решения этой задачи достаточно знать правила перевода конечной дроби и свойств периодических дробей из курса элементарной математики. Заметим, что оценка “развивающего” вопроса не выносится за рамки оценки всей контрольной работы.

3) Мы используем такую форму контроля знаний, как **доклады с тремя участниками** — докладчик, содокладчик и оппонент. Наиболее интересны роли содокладчика и оппонента, т.к. эти ученики должны внимательно выслушать основного докладчика и в первом случае подчеркнуть все положительные моменты доклада, а во втором случае выявить все недостатки и упущения. Система докладов с тремя участниками позволяет не только оценить знания учеников, но и формирует культуру спора, отстаивания своей точки зрения. Мы оцениваем каждого из участников по нескольким критериям: изложение фактического материала, умение работать с литературой, оформление выступления и т.д.

4) При проверке домашних заданий мы придерживаемся системы “одна неделя задержки — минус один балл”. Выполнение каждого практикума рассчитано на определенное время, как правило, 1—2 недели. Если ученик сдает задание в срок, то он получает ту оценку, которую заслуживает. Если ученик по каким-либо причинам (кроме болезни) не сдает свою работу в срок, то отрицательная оценка не выставляется. Но ученик знает, что если через неделю он сдает работу на “отлично”, то получает только “хорошо”, через две недели отметку выше тройки он не получит. Такой подход позволяет при необходимости увеличить время выполнения домашнего задания, но и предупреждает эффект “снежного кома”, так хорошо всем нам знакомый. Задачи практикумов по информатике изданы отдельным сборником. Эти задачки можно взять в школьной библиотеке. В сборнике для каждой задачи даются указания для решения или приводится основной алгоритм или часть программы на Турбо Паскале.

5) В течение учебного года мы проводим четвертные и семестровые **зачеты, или коллоквиумы**, на которые выносятся основные теоретические вопросы.

6) В конце года все классы сдают **экзамен** по информатике. Экзамен проходит в виде теста на компьютере. В тесте от 25 до 30 вопросов.

Шестой принцип (“обозначай и развивай свои интересы”):

При изучении достаточно сложных тем школьникам предоставляется возможность самим сформулировать себе задачу, постановка которой, конечно же, обсуждается с преподавателем. Тем самым школьнику предоставляется возможность заниматься интересующим его вопросом. Например, при изучении принципов построения СУБД школьники могут выбрать следующие практические задания:

- взять задачу из сборника задач;
- написать свою СУБД;
- выполнить практическую работу в стандартной СУБД, например, в dBase III+;
- выполнить практическую работу в Excel.

Седьмой принцип (терминология):

Известно, что использование **продуманной терминологии** существенно повышает усвояемость материала. Мы придерживаемся следующих принципов:

1) Не вводи новое определение, если без него можно обойтись.

2) При введении понятия, ранее введенного на других школьных предметах, необходимо использовать ассоциативную память учеников, это достигается использованием одних и тех же терминов.

3) Если вводится понятие, используемое только в информатике, желательно использовать термины, употребляемые в “настоящей” науке, это повышает значимость данного школьного курса в глазах учеников и в дальнейшем облегчит им изучение этой дисциплины.

4) При выборе терминов в случае вариативности необходимо использовать терминологию, доступную данному возрасту и полученному образованию.

Пример. В процессе преподавания мы пришли к выводу, что слово “переменная” для определения понятия переменной в алгоритмическом языке лучше заменить словом “объект” (с оговоркой, что к объектно-ориентированному программированию это отношения не имеет). Эта замена существенно подняла уровень понимаемости излагаемого материала. Предложение “рассмотрим переменную типа массив” школьники часто не понимают и воспринимают как формальный фон. Предложение “рассмотрим объект типа массив” воспринимается школьниками адекватно.

Выравнивающе-развивающая методика используется в СУНЦ МГУ уже на протяжении 5 лет. Мы считаем ее применение успешным на основании следующих фактов:

1) школьники сдают экзамены по информатике за 10-й и 11-й классы в основном на “хорошо” и “отлично”;

2) выпускники СУНЦ успешно осваивают курсы программирования и информационных технологий на естественно-научных факультетах МГУ, МИФИ, МГТУ им. Баумана и т.п.;

3) наши школьники, как правило, выигрывают командное первенство Москвы по программированию, занимают призовые места на олимпиадах по информатике, включая Российскую (командное и личное первенства);

4) по субъективной оценке преподавателей, наблюдается устойчиво высокий интерес большинства учащихся к информатике на протяжении всего учебного года;

5) результаты анонимного анкетирования учащихся школьным психологом также дают основание утверждать об успешном применении методики. Ниже приведены результаты ответов на вопрос “Оцените по 10-балльной шкале, насколько вам легко, полезно, понятно, интересно на уроках информатики”.

Лекции			
Легко	Понятно	Полезно	Интересно
7,4	6,8	4,8	4,2
7,9	8,8	6,5	5,4
Семинары			
7,6	7,3	6,8	6,2
7,3	8,2	7,6	7,1

1-я строка таблицы — обычные классы.

2-я строка таблицы — классы, преподавание в которых ведется по выравнивающе-развивающей методике.

В заключение надо добавить, что описанную методику можно применять при изучении компьютерных технологий и преподавании курсов программирования в вузах.



А вам не жаль, что кончилось “Жаркое лето 99-го”?

ДОРОГИЕ ЧИТАТЕЛИ!

Летом этого года мы выпустили серию спецвыпусков “Жаркое лето 99-го”. Планируя эту серию, мы старались сделать ее полезной и разнообразной, включив материалы, посвященные различным темам. Надеемся, летние номера пригодятся вам в наступившем учебном году.

Наши постоянные читатели знают, что мы регулярно публикуем анкеты, в которых просим вас оценить номера газеты. Конечно, нам очень интересно и крайне важно знать ваше мнение и о летних номерах.

Пожалуйста, оцените спецвыпуски из серии “Жаркое лето 99-го”, используя пятибалльную систему, по трем критериям:

1. Насколько полезны в наступившем учебном году будут вам материалы данного номера?
2. Насколько интересны они вам были для чтения?
3. Насколько соответствует оформление номера его содержанию?



№ 21

№ 22, 26, 30

№ 23, 28, 31

№ 24, 27, 34

№ 25, 33

№ 29

№ 32

1. __ 2. __ 3. __ 1. __ 2. __ 3. __ 1. __ 2. __ 3. __ 1. __ 2. __ 3. __ 1. __ 2. __ 3. __ 1. __ 2. __ 3. __

Мы благодарим всех читателей, приславших нам анкеты с оценками предыдущих номеров. Как всегда, мы разыграли среди них небольшие призы — книги по информатике. Комплекты номеров “Жаркое лето 99-го” мы разыграли среди тех, кто нашел все изменения, которые были внесены нами в репродукцию картины В.М. Васнецова “Книжная лавка” (это шуточное задание было напечатано в № 14/99). Призы получают:

Ужакин А.И. Алтайский край, Панкрушихинский р-н
Рябых М.И. Орловская обл., Шаблыкинский р-н
Мильтова И.А. Иркутская обл., г. Усть-Кут

Щербакова О.Б. Кемеровская обл., р.п. Яя
Николаева М. Республика Саха (Якутия)
Шалыгин С.В. Новосибирская обл., г. Черепаново

Еще раз большое спасибо всем, приславшим анкеты! Среди тех, кто пришлет нам письма с оценками летних номеров, мы разыграем первые книги по информатике из новой серии “Я иду на урок”, которую начинает выпускать “Первое сентября”.

Гл. редактор С.Л.Островский Зам. гл. редактора Е.Б.Докшицкая Редакция: 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 5000 экз. Заказ №
	ИНДЕКС ПОДПИСКИ для индивидуальных подписчиков 32291 комплекта приложений 32744		Internet: inf@1september.ru Fidonet: 2:5020/69.32 WWW: http://www.1september.ru
Тел. (095)249 3138, 249 3386. Факс (095)249 3184			