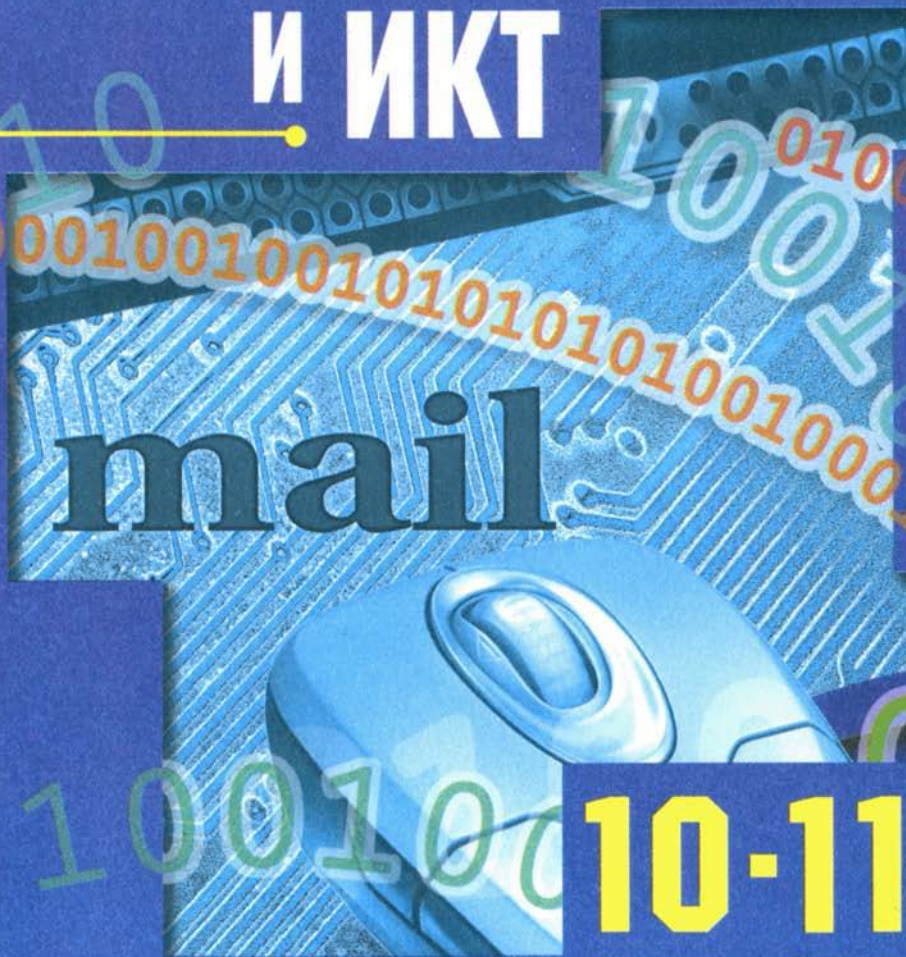


А. Г. ГЕЙН

ИНФОРМАТИКА И ИКТ



mail

10-11

ЗАДАЧНИК-ПРАКТИКУМ


ПРОСВЕЩЕНИЕ
ИЗДАТЕЛЬСТВО

А. Г. ГЕЙН

ИНФОРМАТИКА и ИКТ

**Задачник-
практикум**



10–11 **КЛАССЫ**

**Базовый
и профильный
уровни**

Москва
«Просвещение»
2010

Предисловие

Информатика принадлежит к числу быстро меняющихся дисциплин, и, хотя общие её контуры остаются сравнительно стабильными, в каждом школьном учебнике по-своему расставлены акценты. Данное пособие входит в учебно-методический комплект, разработанный коллективом авторов под руководством А. Г. Гейна. Тем не менее, создавая задачник-практикум, мы ориентировались на общие контуры предмета, очерченные в «Обязательном минимуме образования по информатике». Поэтому данная книга будет, на наш взгляд, полезна при изучении курса информатики по любому учебнику.

Весь материал разделён на разделы, соответствующие основным образовательным линиям курса; разделы делятся на параграфы, которые подразделяются на пункты. Именно для того, чтобы читателю, знакомому с курсом информатики по учебникам других авторов, было комфортно работать с данным пособием, каждый раздел, параграф, пункт открываются краткой сводкой основных определений и сведений; определяемые термины выделены жирным шрифтом. За более подробными разъяснениями в случае необходимости читатель может обратиться к одному из учебников информатики, названия которых приведены в списке рекомендуемой литературы.

Помимо собственно заданий, в книгу включены вопросы по основным теоретическим положениям школьного курса информатики — это, на наш взгляд, поможет читателю проверить себя в знании теории, которая необходима для успешного выполнения последующих заданий.

Задачник призван обслуживать как базовый, так и профильный курс информатики. Задания, относящиеся к профильному уровню, отмечены символом **П**. Кроме того, в задачнике представлены задания разного уровня сложности. Более трудные задания помечены символом *****. Конечно, оценка сложности задания — вещь достаточно субъективная, так что некоторые из указанных заданий могут и не показаться трудными для читателя этой книги.

Ряд заданий ориентирован на их выполнение с помощью компьютера. Такие задания помечены знаком **Э**. В основном такие задания не привязаны к какой-либо конкретной реализации средств информационных технологий.

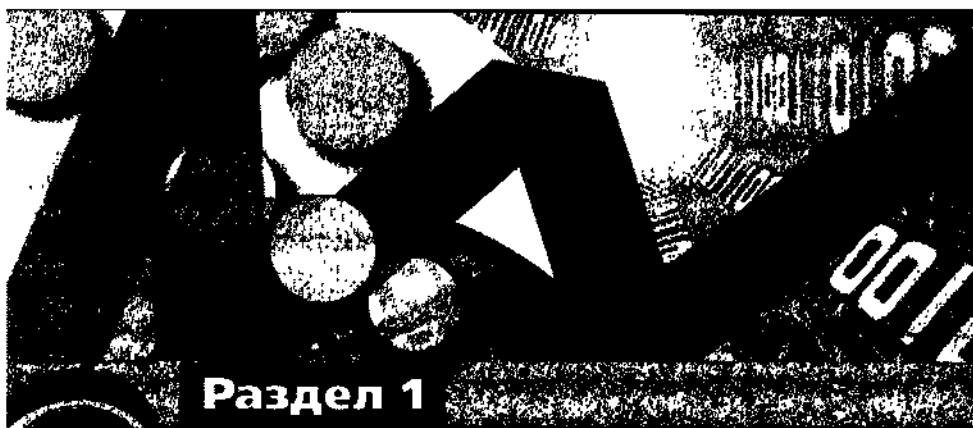
В последнее время наблюдается тенденция переноса некоторых заданий ЕГЭ из части А (задания с выбором ответа) в часть В (задания с кратким ответом). Поэтому в нашем задачнике мы постарались предъявить не только задания,

аналогичные заданиям части В Единого государственного экзамена по информатике, но и возможные трансформации заданий из части А.

В конце задачника к некоторым заданиям приведены ответы. Кроме того, в заданиях, которые предусматривают использование компьютера, для некоторых исходных данных приведены результаты тестирования программ. Это позволяет читателю, самостоятельно работающему с задачником, осуществлять дополнительный контроль за правильностью выполнения заданий.

Мы надеемся, что данная книга будет полезна тем, кто изучает и преподаёт школьную информатику.

Желаем успехов!



Раздел 1



Информация, виды информации и способы её представления

Понятие информации является первичным в информатике и не может быть строго определено в других терминах. Поэтому в информатике понятию «информация» определение не даётся, оно лишь разъясняется и иллюстрируется конкретными примерами применения данного термина. Более того, в разных разделах информатики этот термин понимается более широко или более узко в зависимости от тех проблем, которые в данном разделе рассматриваются. В § 1 приведено несколько таких толкований.

Для передачи информации или её хранения в виде того или иного сообщения требуется определённый способ кодирования. Кодирование своей главной целью имеет *сохранение* информации и придание ей формы, обеспечивающей полноценную (т. е. без потерь и искажений) передачу информации от источника к получателю.

Об информации, представленной последовательностью знаков, говорят, что она *символьная*; информация, представленная посредством какого-либо изображения, называется *видеоинформацией*.



8 Информация и информационные процессы

Информация — это то, что позволяет живому организму (или технической системе) адекватно реагировать на окружающую среду посредством тех или иных механизмов, индуцируя целенаправленную деятельность по сохра-

нению отдельного индивидуума и/или вида (системы) в целом. Такая точка зрения на понятие информации обычно принимается в тех разделах информатики, которые близки к кибернетике.

Нередко говорят, что информация — это сведения, знания об окружающем человека мире и о самом себе. Так обычно разъясняется понятие информации в гуманитарных науках, где центральным объектом является человек как существо социальное.

В теории передачи и хранения информации под информацией понимается последовательность сигналов или символов какого-либо алфавита, кодирующая некоторое сообщение без учёта смыслового содержания этого сообщения. Аналогично понимается термин «информация» и в науках, изучающих компьютерную обработку информации.

Наиболее общее понимание термина «информация» состоит в том, что информация — это отражение разнообразия в существующем мире. Отсутствие разнообразия, когда неразличимы никакие два объекта, явления или процесса, это и есть отсутствие какой бы то ни было информации.

Информацию, зафиксированную каким-либо способом, называют информационным объектом.

Информационный процесс — это процесс, в ходе которого изменяется содержание информации или форма её представления. Среди информационных процессов выделяют следующие основные виды: получение, хранение, передача, обработка, использование информации.

Получение информации — это реализация способности к отражению различных свойств объектов, явлений и процессов в окружающем мире.

Передача информации всегда осуществляется по некоторому каналу связи от источника информации к её приёмнику.

Под обработкой информации понимают любое преобразование её содержания или формы представления. Обработка информации может происходить двумя способами: формально или эвристически. Формальной называется обработка информации, которая производится исполнителем при строгом следовании выданной ему инструкции без учёта смысла обрабатываемой информации. Эвристическая обработка предполагает принятие решений о том, как производится обработка информации в зависимости от самой этой информации. Для конкретной ситуации далеко не всегда можно точно сказать, производится в данной ситуации алгоритмическая или эвристическая обработка информации. Это зависит, в частности, от тех знаний, которыми обладает тот, кто ведёт обработку информации.

Использование информации — обязательный элемент формирования целенаправленной деятельности. При использовании информации выявляются такие её свойства, как новизна, актуальность, достоверность, объективность, ценность, полнота и т. п.

Информация обладает новизной, если её смысловое содержание отличается от смыслового содержания ранее имевшейся информации.

Информация актуальна (иными словами, своевременна), если она оказывает влияние на формирование целенаправленной деятельности именно в данный момент времени.

Информация достоверна, если принимается, что она отражает реальное положение дел, в частности, не вступает в противоречие с уже имеющейся информацией, также признаваемой в качестве достоверной. Вовсе не исключается, что с поступлением новой информации данная информация уже перестанет быть достоверной.

Информация объективна, если она не зависит от свойств источника информации. Надо понимать, что абсолютно не зависеть от свойств источника информация не может, однако при тех или иных условиях можно считать, что такое влияние пренебрежимо мало.

Информация обладает ценностью (или, по-другому, полезностью), если она повышает вероятность достижения цели в целенаправленной деятельности той системы, которая использует эту информацию.

Информация полна, если её достаточно для достижения цели. Полная информация может быть избыточной, если для достижения цели достаточно только части данной информации.

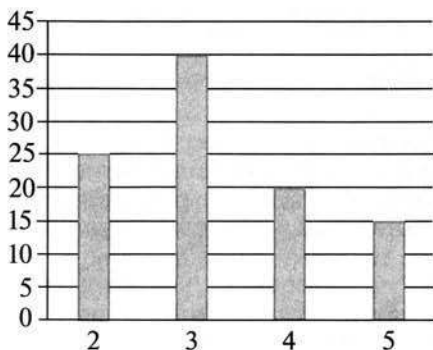
Этими свойствами информация обладает в рамках конкретно протекающего информационного процесса.

Вопросы и задания

- 1) Какую информацию называют символьной? Что такое видеоинформация?
- 2) Кроме символьной и видеоинформации, есть и другие виды информации. Информацию несёт запах; осязание доставляет нам тактильную информацию. Вкусовые рецепторы доставляют информацию о качестве пищи или готовности её к употреблению и т. д.
 - а) Приведите примеры использования человеком информации указанных видов.
 - б) Почему символьную и визуальную формы представления информации считают основными видами информации для человека?

- 3 Назовите основные виды информационных процессов.
- 4 Из перечисленных ниже процессов выделите информационные и укажите, к какому виду информационных процессов они относятся:
- а) производство серной кислоты;
 - б) перевод из единиц измерения в системе СГСЕ в единицы СИ;
 - в) старт спортсменов в беге на дистанцию 100 м;
 - г) фотофиниш спортсменов в беге на дистанцию 100 м;
 - д) объявление победителей в беге на дистанцию 100 м;
 - е) награждение победителей в беге на дистанцию 100 м;
 - ж) измерение уровня кислотности раствора;
 - з) выпечка хлеба;
 - и) поиск грамматических ошибок в тексте;
 - к) уменьшение размеров тела при охлаждении;
 - л) составление меню обеда в школьной столовой;
 - м) выбор блюд из меню обеда в школьной столовой;
 - н) обед в школьной столовой.
- 5 В цехе трудятся рабочие трёх специальностей — токари (Т), слесари (С) и фрезеровщики (Ф). Каждый рабочий имеет разряд не меньше второго и не больше пятого. На диаграмме, изображённой на рисунке 1.1, а, отражено количество рабочих с различными разрядами, а на диаграмме, изображённой на рисунке 1.1, б, — распределение рабочих по специальностям. Каждый рабочий имеет только одну специальность и один разряд по этой специальности. Кроме того, высказаны следующие утверждения, также относящиеся к этому цеху:
- а) Не могут все рабочие третьего разряда быть токарями.
 - б) Среди слесарей есть те, кто имеет разряд выше второго.
 - в) Все рабочие третьего разряда могут быть фрезеровщиками.

а)



б)

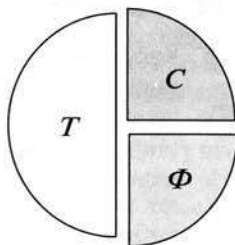


Рис. 1.1

- г) Все слесари могут быть пятого разряда.
- д) Кто-то из токарей имеет разряд ниже четвёртого.
- е) Среди работников третьего и четвёртого разрядов обязательно есть токари.
- ж) Только фрезеровщики имеют четвёртый разряд.
- з) Все рабочие третьего разряда могут оказаться фрезеровщиками.

Укажите, какие из этих утверждений не противоречат данным обеих диаграмм.

- 6 В некотором классе для изучения изменения трудоспособности учащихся было проведено анкетирование. Один из школьников дал следующие ответы на вопросы анкеты:

- а) Лучше всего работается в среду.
- б) Самый непродуктивный день — суббота.
- в) К концу недели продуктивность падает.
- г) В течение недели не бывает больше двух дней с одинаковой продуктивностью.
- д) В понедельник продуктивность такая же, как в пятницу.
- е) В четверг работается хуже, чем во вторник, но лучше, чем в пятницу.
- ж) Вторник — второй по продуктивности день.

Кроме того, в течение той же недели для того же школьника была составлена диаграмма изменения его трудоспособности, измеренной в условных единицах (рис. 1.2).

Укажите, какие из субъективных ощущений школьника совпадают с объективными показателями.

- 7 Тяжёлая атлетика — это прямое соревнование, когда каждый атлет имеет три попытки в рывке, а затем три попытки в толчке. Самый тяжёлый вес поднятой штанги в каждом упражнении суммируется в общем зачёте. Если спортсмен потерпел неудачу во всех трёх попытках в рывке, он может продолжить соревнование в толчке, но уже не сможет занять какое-либо место по сумме двух упражнений.



Рис. 1.2

Если два спортсмена заканчивают состязание с одинаковым итоговым результатом, высшее место присуждается спортсмену с меньшим весом. Если же вес спортсменов одинаков, преимущество отдаётся тому, кто первым поднял победный вес. Рассмотрите таблицу результатов соревнований по тяжёлой атлетике.

Ф.И.О.	Вес спортсмена	Взято в рывке	Рывок с попытки	Взято в толчке	Толчок с попытки
Арестов В. И.	77,1	147,5	2	200,0	2
Борисов Е. В.	79,1	147,5,	1	202,5	1
Воронкин В. А.	78,2	147,5	2	200,0	1
Гончаров С. Т.	78,5	—		202,5	1
Климов К. С.	79,5	150,0	1	200,0	1
Лапин В. А.	77,1	147,5	2	200,0	1
Минин П. С.	78,5	150,0	3	200,0	2
Тарасов О. В.	78,5	147,5	1	202,5	2
Устинов К. Л.	77,6	—		202,5	2
Юдин А. С.	78,2	147,5	3	202,5	3
Якушев М. А.	77,3	150,0	1	—	

- Кто стал победителем по сумме двух упражнений?
- Какое место по сумме двух упражнений занял П. С. Минин?
- У кого лучший результат в толчке?
- Кто выступил лучше по сумме двух упражнений: В. И. Арестов или В. А. Лапин?
- Чей результат по сумме двух упражнений оказался худшим?

- 8 Определите в каких случаях информация, которую намерен получить Петя, избыточна, а в каких недостаточна для того, чтобы сделать требуемый вывод (возможно также, что информации будет достаточно и она не избыточна).
- Требуется определить, является ли данный четырёхугольник прямоугольником. Петя собирается измерить все четыре угла этого четырёхугольника и убедиться, что каждый из них составляет 90° .

б) Требуется определить, принимает ли данный квадратный трёхчлен только положительные значения при всех значениях независимой переменной. Петя намерен проверить отрицательность дискриминанта этого трёхчлена.

9 В приведённых ниже примерах определите, полна ли информация для принятия требуемого решения. Если, на ваш взгляд, она не полна, то какую ещё информацию вы хотели бы иметь?

а) Вы хотите подключиться к сети Интернет. Вы выяснили, какие провайдеры предоставляют такую услугу, сколько стоит подключение к сети и какова абонентская плата у каждого из провайдеров.

б) Вы собираетесь продолжить своё образование в вузе. У вас есть справочник для поступающих, где про каждый вуз сказано, по каким специальностям ведётся подготовка, каковы требования к поступающим, проводятся ли дополнительные вступительные испытания и в какие сроки они проходят, продолжительность обучения, формы обучения (очная, заочная, на платной или бесплатной основе и т. п.).

2

Кодирование символьной информации

Для передачи информации или её хранения в виде того или иного сообщения требуется определённый способ кодирования. Кодирование своей главной целью имеет *сохранение* информации и придание ей формы, обеспечивающей полноценную (т. е. без потерь и искажений) передачу информации от источника к получателю.

Знаковую систему, используемую для представления информации, называют языком. Языки бывают коммуникативные (языки межличностного общения), формализованные (состоящие из терминов и обозначений, за которыми закреплён однозначный смысл) и формальные (определённые формальными синтаксическими правилами). По своему происхождению языки делятся на естественные и искусственные. К формальным искусственным языкам относятся все языки программирования. Примером коммуникативного искусственного языка является эсперанто.

Вся информация, циркулирующая в компьютере, закодирована в двухсимвольном алфавите. Для представления символьной информации в компьютере используются кодовые таблицы. Стандартной является кодовая таблица ASCII (American Standard Code for Information Interchange). В таблице П.1, в Приложении, приведена основная часть этого кода, содержащая латинский алфавит, цифры и ряд вспомогательных символов. Расширения этой таблицы с символами национальных языков получили название кодо-

вых страниц. Для русского языка это прежде всего кодовые страницы CP-866 и CP-1251 (последнюю называют ещё Windows-1251). В таблицах П.2 и П.3 приведены эти страницы. В сети Интернет используется таблица КОИ-8 (Код Информационного Обмена 8-битовый). В Приложении она представлена таблицей П.4. Для экономии места мы не приводим в этих таблицах двоичные коды символов, а только указываем их десятичный порядковый номер.

Вопросы и задания

- 1 В таблице приведён код азбуки Морзе.

А • —	И • •	Р • — •	Ш — — — —
Б — • • •	Й • — — —	С • • •	Щ — — • —
В • — —	К — • —	Т —	Ъ • — — • — •
Г — — •	Л • — • •	У • • —	Ы — • — —
Д — • •	М — —	Ф • • — •	Ь — • • —
Е •	Н — •	Х • • • •	Э • • — • •
Ж • • • —	О — — — —	Ц — • — •	Ю • • —
З — — • •	П • — — •	Ч — — — •	Я • — • —

а) Закодируйте с помощью азбуки Морзе сообщение «ИНФОРМАТИКЕ ПРИНАДЛЕЖИТ БУДУЩЕЕ».

б) Какое сообщение передано с помощью азбуки Морзе?

• — — • • • — • • • — — — • — — — • • • —
 — — • — • • • — • • • • — — • • — • • —
 • • • — — • — • • — • — • — • — • •

- 2 В сообщении, переданном с помощью азбуки Морзе, потеряно разделение на буквы:

— • • • • — — • • — — •

Известно, что в передаче этого сообщения использовалось не более пяти различных символов: А, В, Д, Н, С. Попробуйте восстановить сообщение.

- 3 Пусть стрелка ↑ означает перемещение на одну клетку вверх, стрелка ↓ означает перемещение на одну клетку вниз, стрелка ← — перемещение на одну клетку влево, стрелка → — перемещение на одну клетку вправо.

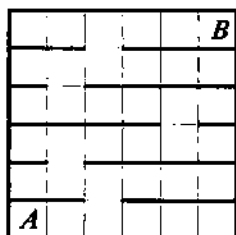


Рис. 2.1

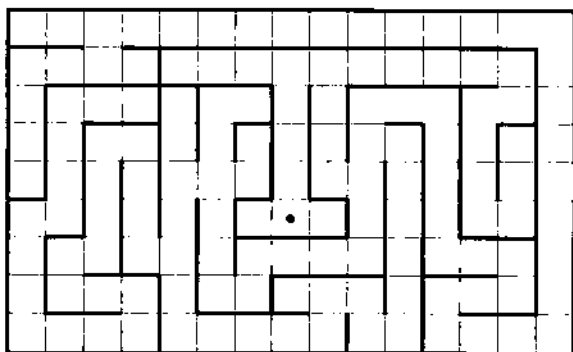


Рис. 2.2

а) Закодируйте последовательностью стрелок кратчайший маршрут из клетки A в клетку B на клетчатом поле с перегородками, изображённом на рисунке 2.1. (За один ход можно переместиться ровно на одну клетку, при этом запрещается проходить сквозь перегородки.)

б) Закодируйте последовательностью стрелок путь от входа лабиринта, изображённого на рисунке 2.2, до его центра (отмеченного точкой).

- 4) Световое табло состоит из лампочек. Каждая лампочка может находиться в одном из трёх состояний («включено», «выключено» или «мигает»). Какое наименьшее количество лампочек должно находиться на табло, чтобы с его помощью можно было передать 18 различных сигналов?

- 5) Для пяти букв русского алфавита заданы их двоичные коды:

A — 001; K — 100; H — 10; O — 101; T — 01.

Какое слово русского языка, состоящее из этих букв, закодировано двоичной строкой 100101100100110001?

- 6) Для пяти букв латинского алфавита заданы их двоичные коды:

A — 10; B — 01; C — 110; D — 101; E — 010.

Декодируйте сообщение 1001010111001, о котором известно, что каждая из пяти букв фигурирует в этом сообщении ровно один раз.

- 7) Представьте двоичным кодом в трёх различных кодировках текст «Время — вперёд!».

- 8) Представьте десятичным кодом в трёх различных кодировках текст «Без труда не вынешь рыбку из пруда!».

- 9) На листе бумаги было записано (по-русски) несколько терминов, относящихся к информатике. В каждом слове каждую букву заменили её порядковым номером в алфавите (вместо «а»

написали 1, вместо «л» — 13 и т. д.). Получились следующие последовательности цифр: 11341618102014, 12161417303220618, 118231031241033. Восстановите закодированные слова.

- 10 Символ «пробел» играет в тексте сообщения важную роль — ведь без него

текст слипся бы, и читать его было бы просто противно.

Но роль этого символа не ограничивается обеспечением удобства чтения текста. Без него иногда просто невозможно однозначно разделить последовательность букв на осмысленные слова. Например, последовательность букв «поленоров» можно разбить пробелом на два осмысленных слова двумя способами — «поле норов» и «полено ров». Впрочем, допустимо ещё и разбиение двумя пробелами в три слова — «поле но ров».

а) Каким образом текст «Поднимите перья так же» можно разбить пробелами на слова так, чтобы получалась осмысленная фраза? Приведите как можно больше вариантов.

б) Придумайте ещё какие-нибудь последовательности букв, допускающих неоднозначное разбиение пробелом на группы слов русского языка.



Кодирование числовой информации

Знаки, посредством которых кодируется числовая информация, называются **цифрами**. Способ представления числовой информации с помощью цифр называется **системой счисления**. Различают **позиционные** и **непозиционные** системы счисления. В позиционной системе значение цифры зависит от того, где эта цифра расположена в записи числа. Это место называют **разрядом**; разряды нумеруются слева направо. В непозиционных системах значение цифры не связано с занимаемым ею местом.

3.1. Позиционные системы счисления с произвольным основанием

Среди позиционных систем особую роль играют системы, в которых каждый следующий разряд в b раз больше предыдущего (b — фиксированное для данной системы число, большее 1). В этом случае b называют **основанием системы**, а саму систему счисления называют **b -ичной**. Для записи любого натурального числа в b -ичной системе требуется b цифр. Обычно используются цифры от 0 до $b - 1$ ¹.

¹ Другие возможные варианты цифр рассматриваются в п. 4.3.

Запись $a_1 a_2 \dots a_{n-1} a_n$ является представлением числа c в позиционной системе счисления с основанием b , если

$$c = a_1 \cdot b^n + a_2 \cdot b^{n-1} + \dots + a_{n-1} \cdot b^1 + a_n \cdot b^0,$$

причём каждый коэффициент a_i неотрицателен и меньше b . Из этого равенства видно, что последняя цифра a_n представляет собой остаток при делении числа c на b . А частное от такого деления используется для нахождения предпоследней цифры a_{n-1} — она получается как остаток при делении этого частного снова на b . И т. д.

Алгоритм обратного перевода заключается в следующем. В одной строке записывается число, которое нужно перевести, а строкой ниже будет вычисляться запись числа в нужной системе счисления. Для этого первая цифра переписывается без изменения, а под каждой следующей цифрой записывается число, полученное сложением этой цифры с произведением слева стоящего числа на основание системы счисления. Этот алгоритм называется схемой Горнера.

Позиционный принцип распространяется и на дробные числа: запись $a_1 a_2 \dots a_{n-1} a_n, a_{n+1} \dots a_m$ является представлением дробного числа c в системе счисления с основанием b , если

$$c = a_1 \cdot b^n + a_2 \cdot b^{n-1} + \dots + a_{n-1} \cdot b^1 + a_n \cdot b^0 + a_{n+1} \cdot b^{-1} + \dots + a_m \cdot b^{-m},$$

где по-прежнему каждый коэффициент a_i неотрицателен и меньше b . Указанные выше алгоритмы перевода соответствующим образом модифицируются.

Главное удобство позиционной нумерации состоит в том, что действия над числами в такой системе счисления выполняются *поразрядно*. Всё, что требуется знать для выполнения действий над многозначными числами, — это таблицы сложения и умножения для однозначных чисел.

Если число записано не в десятичной системе счисления, то основание системы будем записывать в виде нижнего индекса справа от числа, например 13_6 , 357_{11} . В случае, когда основание системы ясно из контекста, мы не будем, как и для десятичной системы, писать этот индекс.

Вопросы и задания

- ① Будет ли позиционной система счисления, в которой для записи чисел используется ровно одна цифра?
- ② Запишите наименьшее и наибольшее n -разрядные числа в системе счисления с основанием b при:
 - а) $b = 3$; $n = 4$; б) $b = 4$; $n = 3$; в) $b = 7$; $n = 7$.

Каждую пару полученных чисел переведите в десятичную систему счисления.

- 3 Сравните числа:
а) 1234_9 и 765_9 ; б) 34543_{11} и 43634_{11} ;
в) $\overline{\text{TTTTTTTTTTTTTT}}_{25}$ и $\overline{\text{TTTTTTTTTTTTTTTTT}}_{25}$;
г) $53535353535353535353_{53}$ и $453535353535353535353_{53}$.
- 4 К чему приводит умножение на 7 числа, записанного в шестеричной системе счисления? А умножение на 49?
- 5 П а) Составьте таблицы сложения и умножения в шестеричной системе счисления.
б) Используя результат пункта а, вычислите $54_6 + 45_6$ и $54_6 \cdot 45_6$.
- 6 Переведите числа 634 и 5321 в системы счисления с основанием: а) 5; б) 7; в) 9.
- 7 Переведите в десятичную систему счисления числа:
а) 754_9 ; б) 2145_{11} ; в) $32\,475_{12}$; г) 2146_7 ; д) $24\,584_9$.
- 8 Переведите в десятичную систему счисления дробные числа:
а) $75,6_9$; б) $23,88_{11}$; в) $345,96_{12}$; г) $65,23_7$; д) $57,063_9$.
Ответ запишите обыкновенными дробями.
- 9 Переведите в пятеричную систему счисления дробные числа:
а) $73,2$; б) $32,87$; в) $30,84$; г) $45,23$; д) $84,268$.
Результат, если необходимо, округлите до шестого разряда после запятой.
- 10 Определите, может ли в какой-либо системе счисления быть верным равенство:
а) $7 + 8 = 13$; б) $7 + 8 = 20$; в) $7 + 8 = 23$.
- 11 Для каждого из следующих равенств определите основание системы счисления, в которой оно справедливо:
а) $232 + 302 + 23 = 1001$; б) $242 + 312 + 224 = 1111$;
в) [Я] $15,5 + 23,6 + 14,3 = 60$.
- 12 а) Укажите все основания системы счисления, после перевода в которую числа 47_{10} получается число, оканчивающееся на цифру 2.
б) Укажите все основания системы счисления, для которой после перевода в неё десятичного числа 41_{10} получается число, оканчивающееся на цифру 6.
- 13 В системе счисления с некоторым основанием десятичное число 12 записывается как 110. Укажите это основание.
- 14 У отца, увлекающегося математикой, спросили о его детях. Он ответил так: «У меня 11 детей. Все они учатся в школе. Младшему из них 100 лет, а старший заканчивает 101-й класс». В какой системе счисления названы все числа в этом рассказе? Сколько детей у этого отца?

- 15 П Чему равно x , если выполнено равенство:
а) $15_x + 27_{2x} = 13_{6x}$; б) $103_x + 24_{2x} = 12_{6x}$?
- 16 П Для двух позиционных систем счисления с основаниями x и y выполняется равенство $486_x = 902_y$. Какое наименьшее значение может иметь x ?
- 17* П а) Для позиционных систем счисления с основаниями x , y и xy выполняется равенство $35_x + 23_y = 16_{xy}$. Найдите x и y .
б) Найдите все такие пары натуральных чисел x и y , чтобы в системах счисления с основаниями x , y и xy выполнялось равенство $23_x + 34_y = 19_{xy}$.
- 18 П а) В системе счисления с основанием x число, записываемое как 54, является квадратом целого числа. Найдите наименьшее возможное значение x .
б) Выполните то же задание для числа, записываемого как 72.
в)* Существует ли такая позиционная система счисления, в которой число, записанное как 32, является квадратом целого числа?
г)* Ответьте на тот же вопрос для чисел, записанных как 42 и 43.
- 19 Натуральное число в десятичной системе счисления оканчивается на цифру 7, а при переводе его в девятеричную и двенадцатеричную системы счисления получаются числа, оканчивающиеся на цифру 5. Какое наименьшее число обладает таким свойством?
- 20 Наиболее известная из непозиционных систем счисления — римская нумерация. В ней семь цифр: I, V, X, C, D, L, M. Они означают соответственно 1, 5, 10, 50, 100, 500, 1000. Если меньшая цифра стоит справа, то значение прибавляется к значению предыдущей, а если слева, то вычитается.
а) Число MDLXI записали в десятичной системе, затем поменяли местами две средние цифры. Запишите получившееся число снова римскими цифрами.
б) Выполните такое же задание для числа MLDCII.

3.2. Системы счисления, используемые в программировании

В компьютере вся информация представляется в двоичном коде. Поэтому для кодирования числовой информации применяется двоичная система счисления. Чтобы уменьшить длину записи, в программировании используются восьмеричная и шестнадцатеричная системы счисления. Десятичные, двоичные, восьмеричные и шестнадцатеричные коды первых шестнадцати натуральных чисел представлены в таблице.

Десятичная система счисления	Двоичная система счисления	Восьмерич- ная система счисления	Шестнадцатерич- ная система счисления
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

Для систем с основанием $b = 2^n$ существует простой алгоритм перевода из двоичной системы в систему с основанием b . Для этого запись числа в двоичной системе разбивают на блоки по n цифр, начиная от разряда единиц, и каждую такую группу заменяют цифрой соответствующей системы счисления (при $b = 8$ такой блок называют триадой, при $b = 16$ его называют тетрадой). Обратный перевод осуществляется расписыванием каждой цифры её представлением в двоичной системе счисления.

Вопросы и задания

- а) Переведите из десятичной в двоичную, восьмеричную и шестнадцатеричную системы числа: 27, 72, 123, 987, 10101.
б) Пользуясь инженерным калькулятором, переведите из десятичной в двоичную, восьмеричную и шестнадцатеричную системы числа: 2347, 32 568, 777 777.
- Переведите из двоичной в десятичную систему числа: 1011, 11001, 11011, 10101101.
- Переведите из восьмеричной в десятичную систему числа: 25, 47, 173, 721, 2345, 7777.

- 4 а) Переведите из шестнадцатеричной в десятичную систему счисления числа: 52, F4, A7B, BCD.
б) ☐ Пользуясь инженерным калькулятором, переведите из шестнадцатеричной в десятичную систему счисления числа: ED1A, F0F0.
- 5 Сколько существует натуральных чисел, меньших 1000, содержащих в своей записи ровно три единицы после перевода их в двоичную систему счисления?
- 6 Переведите из двоичной в восьмеричную и шестнадцатеричную системы счисления следующие числа:
а) 1011; 11001; 11011; 10101101;
б) ☐ 10,01; 1010,1; 11,1011; 1011,1101.
- 7 Переведите из шестнадцатеричной в двоичную систему счисления следующие числа:
а) 31; 6F; 2A9; ACE; D1AE; FFFE;
б) ☐ 3,1; 5,F; 2A,9; A,CE; D1,AE; F,FFF.
- 8 Переведите из шестнадцатеричной в восьмеричную систему счисления следующие числа:
а) 31; 6F; 2A9; ACE; D1AE; FFFF;
б) ☐ 3,1; 5,F; 2A,9; A,CE; D1,AE; F,FFF.
- 9 Переведите из восьмеричной в двоичную систему счисления следующие числа:
а) 31,47; 216; 654; 5432; 7070.
б) ☐ 3,1; 0,47; 2,16; 65,4; 54,32; 7,077.
- 10 Переведите из восьмеричной в шестнадцатеричную систему счисления следующие числа:
а) 31,47; 216; 654; 5432; 7070;
б) ☐ 3,1; 0,47; 2,16; 65,4; 54,32; 7,077.
- 11 ☐ а) Натуральное число A , записанное в шестнадцатеричной системе счисления, четырёхзначно, а будучи записанным в восьмеричной системе счисления становится пятизначным. Какое наименьшее значение может иметь число A ? А наибольшее? Ответ дайте в десятичной системе счисления.
б) Натуральное число A , записанное в восьмеричной системе счисления, семизначно, а будучи записанным в шестнадцатеричной системе счисления становится шестизначным. Какое наименьшее значение может иметь число A ? А наибольшее? Ответ дайте в десятичной системе счисления.
- 12 Сравните между собой числа, записанные в двоичной системе счисления:
а) 10101 и 101010; б) 1101 и 1001; в) 111001 и 110111.
Для пары чисел из пункта в определите, на сколько одно число больше другого.

- 13 П Расположите числа в порядке возрастания:
 а) $2A,3_{16}$; $41,25_{10}$; $51,1_8$; $101010,001_2$;
 б) $1C,6_{16}$; $28,75_{10}$; $34,5_8$; $11100,111_2$;
 в) $13,E_{16}$; $19,7_{10}$; $23,5_8$; $10011,111_2$.
- 14 Среди всех чисел, заключённых между числами 515 и 580, найдите те, которые после перевода в двоичную систему содержат в своей записи наибольшее количество единиц.
- 15 Составьте таблицы сложения и умножения в системе счисления с основанием: а) 2; б) 8; в) 16.
- 16 П Используя результат задания 15а, найдите в двоичной системе счисления:
 а) суммы $1011 + 1101$, $101,01 + 100,01$ и $101,01 + 110,11$;
 б) разности $110\ 101 - 101\ 111$ и $100\ 000 - 11,1$;
 в) произведения $111 \cdot 101$ и $101,01 \cdot 101,11$;
 г) целочисленное частное и остаток при делении $101\ 101$ на 100 и $100\ 001$ на 101 .
- 17 П Используя результат задания 15б, найдите в восьмеричной системе счисления:
 а) суммы $2222 + 6666$ и $671,76 + 370,3$;
 б) разности $101 - 110\ 111$ и $100\ 000 - 11,1$;
 в) произведения $567 \cdot 321$ и $570,25 \cdot 543,04$;
 г) целочисленное частное и остаток при делении $707\ 707$ на 100 и $100\ 000$ на 707 .
- 18 П Используя результат задания 16в, найдите в шестнадцатеричной системе счисления:
 а) суммы $8888 + 5555$ и $ABCD, F + 30\ 70,3$;
 б) разности $111\ 101 - 101\ 111$ и $100\ 000 - 1,11$;
 в) произведения $7A5 \cdot 1B3$ и $C4D,5 \cdot F4A,6B$;
 г) целочисленное частное и остаток при делении $FFFFFF$ на 100 и $100\ 000$ на FFF .
- 19 П а) Найдите значение выражения $110_{16} : 10_2 - 100_8 : 10_2$. Ответ дайте в системе счисления с основанием 4.
- 20 П Решите уравнение $1101_2 + 10_2 \cdot x = 101010_2$. Ответ дайте в системе счисления с основанием 8.
- 21 а) Используя кодовую таблицу CP-1251 (см. Приложение, табл. П.3), закодируйте фразу «Знание — сила!» в двухсимвольном алфавите (0 и 1) и полученный двоичный код запишите шестнадцатеричными цифрами.
 б) Выполните такое же задание для фразы «Никто не забыт, ничто не забыто!», используя кодовую таблицу КОИ-8 (см. Приложение, табл. П.4).
 в) Придумайте какую-нибудь фразу, закодируйте её в двухсимвольном алфавите с помощью кодовой таблицы CP-866 (см. Приложение, табл. П.2) и полученный двоичный код запишите восьмеричными цифрами. Передайте полученное вы-

ражение соседу по парте и предложите восстановить исходную фразу. Проверьте, правильно ли он выполнил задание.

- 22) Замените звёздочки цифрами так, чтобы получились верные записи примеров в двоичной системе счисления:

$$\begin{array}{r} \text{a) } + \quad *1*10* \\ \quad \quad *0*1* \\ \hline \quad \quad ***1000 \end{array}$$

$$\begin{array}{r} \text{б) } - \quad **001* \\ \quad \quad *0*0* \\ \hline \quad \quad *1*01 \end{array}$$

$$\begin{array}{r} \text{в) } \quad *1** \\ \quad \times \quad *** \\ \hline \quad \quad **** \\ + \quad \quad **** \\ \hline \quad \quad \quad \quad **** * 1 \end{array}$$

$$\begin{array}{r} \text{г) } - \quad **0*** \quad | \quad **** \\ \quad \quad **** \quad \quad | \quad *** \\ \hline \quad \quad **** \\ - \quad \quad **** \\ \hline \quad \quad \quad \quad 0 \end{array}$$

3.3. □ Уравновешенные и другие системы счисления

В позиционной системе счисления с нечётным основанием b в качестве цифр могут использоваться целые числа в диапазоне от $-\frac{b-1}{2}$ до $\frac{b-1}{2}$. Такая система счисления называется **уравновешенной**. В уравновешенных системах счисления в записи отрицательных цифр вместо знака «-» обычно пишут черту над цифрой. Алгоритмы перевода из десятичной системы счисления в уравновешенную систему с основанием b (последовательное деление с остатком) и обратно (схема Горнера) остаются теми же — надо только при делении выделять остаток, заключённый между $-\frac{b-1}{2}$ и $\frac{b-1}{2}$.

В нашей стране в 70-х гг. был создан компьютер «Сетунь», выполнявший вычисления в троичной уравновешенной системе счисления.

! Вопросы и задания

- 1) Напишите цифры, которые используются в уравновешенной системе счисления с основанием: а) 3; б) 5; в) 11.
- 2) Запишите 3; 5; 121; 534; -2; -16; -121 в уравновешенной системе счисления:
а) с основанием 3; б) с основанием 5; в) с основанием 11.
- 3) а) Переведите в обычную десятичную систему счисления числа, записанные в уравновешенной троичной системе счисления: 10 $\overline{1}$ 1; 1 $\overline{1}$ 1 $\overline{1}$ 01; $\overline{1}$ 1 $\overline{1}$; 1 $\overline{1}$ 0 $\overline{1}$ 1; $\overline{1}$ 1 $\overline{1}$ 1.
б) Переведите в обычную десятичную систему счисления числа, записанные в уравновешенной системе счисления с основанием 5: 102 $\overline{1}$ 1; 1 $\overline{1}$ 0 $\overline{2}$; $\overline{1}$ 10 $\overline{2}$ 1; 2 $\overline{1}$ 1 $\overline{1}$.

- в) Переведите в обычную десятичную систему счисления числа, записанные в уравновешенной системе счисления с основанием 11: $153\bar{1}1$; $5\bar{1}2\bar{2}$; $1044\bar{2}$; $2\bar{1}5\bar{1}4$.
- 4 а) Сформулируйте правило сравнения двух целых чисел, записанных в одной и той же уравновешенной системе счисления.
 б) Сформулируйте правило, по которому осуществляется переход от числа, записанного в данной уравновешенной системе счисления, к противоположному числу.
- 5 а) Составьте таблицы сложения и умножения однозначных чисел в троичной уравновешенной системе счисления.
 б) Составьте таблицы сложения и умножения однозначных чисел в пятеричной уравновешенной системе счисления.
- 6 а) Сформулируйте правило сложения двух целых чисел, записанных в одной и той же уравновешенной системе счисления.
 б) Сформулируйте правило умножения двух целых чисел, записанных в одной и той же уравновешенной системе счисления.
- 7 Выполните действия в троичной уравновешенной системе счисления:
 а) $1001 + 1111$; б) $110101 + \bar{1}010\bar{1}$;
 в) $\bar{1}11\bar{1}1\bar{1} + \bar{1}\bar{1}1\bar{1}1\bar{1}$; г) $1010\bar{1}11 + \bar{1}0\bar{1}01\bar{1}\bar{1}$;
 д) $10\bar{1}1\bar{1}01 - \bar{1}0110\bar{1}\bar{1}$; е) $10\bar{1}1\bar{1} \cdot \bar{1}10\bar{1}1$;
 ж) $\bar{1}\bar{1}01\bar{1} \cdot \bar{1}10\bar{1}01$; з) $*1\bar{1}11\bar{1} : 111$.
- 8 Выполните действия в пятеричной уравновешенной системе счисления:
 а) $21212 + 121212$; б) $\bar{1}201\bar{2}1 + 22122$;
 в) $\bar{2}\bar{2}\bar{2}\bar{2}\bar{2} + 10201$; г) $\bar{2}\bar{1}10\bar{1}2 + \bar{2}\bar{2}12\bar{1}1$;
 д) $2\bar{1}0\bar{1}\bar{1}01 - \bar{1}020\bar{1}\bar{1}1$; е) $2\bar{1}02 \cdot \bar{1}2\bar{1}1$.

Цветное изображение на экране компьютера образуется смешением трёх основных цветов: красного, синего и зелёного. Изменение интенсивности того или иного цвета и создаёт нужный оттенок, воспринимаемый человеческим глазом как единое целое. В таблице приведено кодирование цветов, если для каждого цвета есть только две возможности — светиться с определённой интенсивностью (кодируется 1) либо не светиться вообще (кодируется 0). Указанное кодирование цвета называется *RGB*-кодировкой (от английского названия цветов — Red, Green, Blue).

Красный	Синий	Зелёный	Цвет
0	0	0	Чёрный
0	0	1	Зелёный
0	1	0	Синий
1	0	0	Красный
0	1	1	Бирюзовый
1	0	1	Жёлтый
1	1	0	Малиновый
1	1	1	Белый

Для кодирования графической информации на экране его делят на множество рядов одинаковых квадратиков. Каждый такой квадратик называется пиксель (от английского PICTure'S ELeмент — элемент картинки). Число строк на экране и количество пикселей в строке называют разрешением заданного графического режима.

В современных компьютерах используется 16-битное (режим Hi-Color) и 24-битное (режим True-Color) кодирование. В первом случае оказывается возможным закодировать $2^{16} = 65\,536$ цветов, во втором — $2^{24} = 16\,777\,216$ цветов. В режиме True-Color на кодирование градаций яркости каждого из основных цветов отводится 1 байт: код 00000000 показывает, что данного цвета нет вообще, а код 11111111 соответствует наибольшей интенсивности (яркости) кодируемого цвета. В случае 16-битного кодирования ситуация иная: на кодирование яркости красного и синего цветов отводится по 5 битов, а на кодирование зелёного цвета — оставшиеся 6 бит.

Вопросы и задания

- 1) Что является пикселем в случае цветного монитора и почему он так называется?
- 2) Для хранения растрового изображения размером 128×128 пикселей отвели 4 Кб памяти. Каково максимально возможное число цветов в палитре изображения?
- 3) Растровый графический файл содержит цветное изображение с палитрой из 256 цветов размером 10×10 пикселей. Каков информационный объём этого файла?

- 4 а) Один из самых первых графических режимов, называвшийся сокращённо CGA (от Crayon Graphics Adaptor — графический адаптер «как цветной карандаш»), предусматривал 320×200 пикселей на экране. Каждый из пикселей мог гореть одним из восьми цветов — именно тем, который указан в таблице. Сколько бит требовалось, чтобы закодировать изображение в этом режиме?
 б) Выполните то же задание для режима VGA (Video Graphics Adaptor) и экрана 640×480 пикселей. При этом каждый элемент триады допускает не две, а 64 градации яркости.
- 5 В процессе преобразования растрового графического файла его объём уменьшился в 1,5 раза. Сколько цветов было в палитре первоначально, если после преобразования было получено растровое изображение того же разрешения в 256-цветной палитре?
- 6 а) Набору яркостей (1/2; 0; 1/2) соответствует фиолетовый цвет. Какой цвет, по вашему мнению, соответствует набору (1/4; 0; 1/4)? А набору (3/4; 0; 3/4)?
 б) Каким набором яркостей характеризуется бирюзовый цвет?
- 7 Пусть используется режим True-Color. Укажите цвет, который задаётся кодом:
 а) 111111111111111100000000;
 б) 100000001000000010000000;
 в)* 000000000111111101111111.
- 8 При кодировании цвета соответствующая битовая последовательность нередко записывается в виде шестнадцатеричного числа. Укажите цвет, который задаётся кодом:
 а) FF00FF; б) CCCCCC; в) 009900; г)* FF9933.
- 9 При кодировании цвета в режиме True-Color битовая последовательность, соответствующая коду каждого основного цвета, рассматривается как натуральное число в двоичной системе и переводится в десятичную систему счисления. В коде всей триады цветов эти десятичные числа отделяются друг от друга точкой. Например, зелёный цвет задаётся как 0.255.0. Укажите цвет, который задаётся десятичным кодом:
 а) 0.255.255; б) 102.102.102; в) 204.0.204; г)* 204.51.255.
- 10 Пусть используется режим Hi-Color. Укажите цвет, который задаётся кодом:
 а) 1111100000000000;
 б) 0011100111100111;
 в)* 0000001111111111.
- 11 **П** При печати цветного изображения на бумаге используется иная кодировка цвета. Связано это с тем, что изображение создаётся лучами, отражёнными от белого листа бумаги, а типографская краска представляет собой светофильтр, погло-

щающий ту или иную составляющую белого цвета (который, как известно из физики, является смесью всех цветов спектра).

а) Возьмите лупу и рассмотрите какую-либо цветную иллюстрацию в книге. Из каких цветов, кроме чёрного, составлено изображение?

б)* Выполнив задание а, вы обнаружили, что, кроме чёрной, при печати используется краска трёх цветов: малинового, бирюзового и жёлтого. Бирюзовый цвет на бумаге поглощает красную составляющую. Синий цвет поглощается жёлтым, а зелёный — малиновым. Чёрный цвет получается, если нанести на бумагу все три поглощающих цвета. Сказанное выше занесено в три строки следующей таблицы.

Бирюзовый (нет красного)	Жёлтый (нет синего)	Малиновый (нет зелёного)	Цвет
0	0	0	
0	0	1	Малиновый
0	1	0	Жёлтый
1	0	0	Бирюзовый
0	1	1	
1	0	1	
1	1	0	
1	1	1	

Заполните остальные строки таблицы. (Совет: догадайтесь, как для этого использовать таблицу, приведённую на с. 23.)

в)* Кодировка, которая рассмотрена в пункте б, называется **СМУ-кодировкой** (по первым буквам английского названия основных цветов — Cyan, Magenta, Yellow). Пусть интенсивность каждого цвета в **RGB**-кодировке принимает значения от 0 до 1. Иными словами, каждый цвет кодируется тройкой чисел, в которой вначале идёт доля красного цвета, затем зелёного и, наконец, синего. Бирюзовый цвет, таким образом, будет иметь **RGB**-код (0, 1, 1). Аналогично в **СМУ**-кодировке каждый цвет задаётся тройкой чисел, показывающей долю бирюзового, жёлтого и малинового цветов. Напишите формулы перехода из **RGB**-кодировки в **СМУ**-кодировку.

II Коды, обнаруживающие и исправляющие ошибки

В ходе передачи информации внешние воздействия могут вносить изменения в передаваемый код. Чтобы устранить воздействие внешних помех, код делают избыточным, т. е. передаваемая информация кодируется не минимально необходимым количеством бит, а с использованием дополнительных двоичных символов. Это позволяет автоматизировать поиск и исправление ошибок.

Базовым инструментом в автоматическом поиске и исправлении ошибок является понятие расстояния Хэмминга. Расстоянием Хэмминга между двумя битовыми последовательностями одинаковой длины называют количество позиций, в которых эти две последовательности различаются. Множество всех битовых последовательностей, используемых для кодирования символов заданного алфавита, называется кодом. Кодовым расстоянием называется минимальное из расстояний между любыми двумя последовательностями, принадлежащим данному коду. Пусть d — кодовое расстояние. Если при передаче кода символа количество искажённых бит оказалось не больше чем $\frac{d-1}{2}$, то имеется ровно одна кодовая последовательность, ближайшая к той последовательности, которая получена приёмником сообщения. В этом случае именно эта последовательность принимается за исходно переданную.

Вопросы и задания

- 1) Чему равно расстояние Хэмминга между последовательностями:
 - а) 10010110 и 10110101; б) 011010101 и 001011111;
 - в) 1011010101 и 001010100101?
- 2) Рассматривается множество всех пятисимвольных слов над алфавитом $\{0, 1\}$.
 - а) Перечислите все слова, находящиеся на расстоянии 1 от слова 11101.
 - б) Перечислите все слова, находящиеся на расстоянии 2 от слова 01011.
 - в) Сколько существует слов, находящихся на расстоянии 3 от слова 11111.
- 3) Для кодирования 15 букв русского алфавита и пробела использовался код, представленный в таблице.

Буква	Код	Буква	Код
Пробел	0000000	З	1000011
А	1001100	И	0001111
Б	1110000	Й	0010110
В	1100110	К	0011001
Г	1011010	Л	0100101
Д	1010101	М	0101010
Е	0111100	Н	0110011
Ж	1101001	О	1111111

а) Найдите кодовое расстояние. Сколько ошибок гарантированно исправляет этот код?

б) Получено сообщение

11100100101100101001110100011110111011101100100111011
11101110000000100011011101110001100000101111

Попытайтесь его декодировать, исправив, если необходимо, ошибки.

в) Выполните такое же задание для следующего сообщения:

11011011010001000110100010001000111110111011110110111
01110010000010000001101110111110101101111101100111011
11110101011011011100

г) Придумайте небольшое сообщение (10—12 символов), составленное из букв, фигурирующих в таблице из задания 3, закодируйте его, внесите в кодировку некоторых символов по одной ошибке и предложите соседу по парте декодировать сообщение. Проверьте, как он выполнил эту работу.

- 4* Попытайтесь обосновать высказанное выше утверждение, что для кода с кодовым расстоянием d любая последовательность, которая не более чем в $(d-1)/2$ местах отличается от некоторой кодовой последовательности, есть ровно одна ближайшая к ней кодовая последовательность.



6

Измерение количества информации

Понимание, что является мерой для количества информации, зависит от трактовки самого понятия информации (см. преамбулу данного раздела). Каждый из получающихся в этом случае подходов к измерению количества инфор-

мации представлен в пунктах этого параграфа. Тем не менее независимо от подхода единицей измерения информации является 1 бит (от английского *Binary digit* — двоичная цифра). Обычно используются более крупные единицы измерения информации — 1 байт, 1 килобайт (Кб), 1 мегабайт (Мб), 1 гигабайт (Гб) и т. д. Соотношение между этими единицам таково:

$$\begin{aligned} 1 \text{ байт} &= 8 \text{ бит} \\ 1 \text{ Кб} &= 1024 \text{ байт} \\ 1 \text{ Мб} &= 1024 \text{ Кб} \\ 1 \text{ Гб} &= 1024 \text{ Мб} \end{aligned}$$

Вопросы и задания

- 1 Каков коэффициент пересчёта байтов в килобайты; килобайтов в мегабайты? А коэффициент пересчёта битов в байты?
- 2 Сколько байт в одном мегабайте? А бит?

6.1. Информационный объём сообщения

Любая информация, представленная в виде сообщения, может быть закодирована в двухсимвольном алфавите. Количество символов в такой кодировке называют **информационным объёмом сообщения**.

Количество информации, закодированной одним символом двоичного алфавита, принимают за 1 бит.

Для кодирования символьной информации в компьютере применяется обычно 8-битный код ASCII или 16-битный код UNICODE.

Если сообщение закодировано в N -символьном алфавите, то информационный объём одного символа принимают равным $\log_2 N$. Этот коэффициент используется при подсчёте количества информации, записанной не в двухсимвольном алфавите.

Вопросы и задания

- 1 а) Информационный объём сообщения равен 40 бит. Выразите объём того же сообщения в байтах.
б) Информационный объём сообщения равен 23 Кб. Выразите объём того же сообщения в байтах и битах.
- 2 а) В сообщении «Компьютер — основное средство информационных технологий.» каждый символ кодируется в системе UNICODE. Подсчитайте информационный объём этого сообщения в байтах.

б) В сообщении «С. А. Лебедев — создатель первой отечественной ЭВМ.» все символы кодируются в системе ASCII. Подсчитайте информационный объем этого сообщения в байтах.

3 Подсчитайте, каков информационный объем этой страницы задачника, округлив полученный результат до целого числа килобайт (колонтитул и номер страницы не учитываются).

4 а) Сколько символов можно закодировать, используя ASCII?

б) Сколько символов можно закодировать, используя UNICODE?

5 а) Автоматическое устройство осуществило перекодировку информационного сообщения длиной в 35 символов, первоначально записанного латинскими буквами в коде UNICODE, в кодировку ASCII. На сколько изменился информационный объем сообщения?

б) Автоматическое устройство осуществило перекодировку информационного сообщения длиной в 35 символов, первоначально записанного русскими буквами в кодировке CP-1251, в UNICODE. На сколько изменился информационный объем сообщения?

6 Для передачи секретного сообщения используется код, состоящий из десятичных цифр. При этом все цифры кодируются одним и тем же (минимально возможным) количеством бит. Определите информационный объем сообщения длиной в 200 символов.

7 В велокроссе участвуют 106 спортсменов. Специальное устройство регистрирует прохождение каждым из участников промежуточного финиша, записывая его номер с использованием минимально возможного количества бит, одинакового для каждого спортсмена. Каков информационный объем сообщения, записанного устройством, после того как промежуточный финиш прошли 95 велосипедистов?

8 В России каждый автомобильный номер состоит из трёх главных букв и трёх цифр (всего в различных номерах задействовано 12 букв русского алфавита, совпадающих по начертанию с буквами латинского алфавита, и все 10 цифр). При этом на первом и двух последних местах стоят буквы, на трёх оставшихся местах стоят цифры. Каждый такой номер записывается минимально возможным и одинаковым целым количеством байт (при этом используется посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством бит). Определите объем памяти, необходимой для записи 700 номеров.

9 а) «Сколькибитное» кодирование вы бы предложили для языка племени «Мумбо-Юмбо», в алфавите которого 32 буквы и все заглавные, а цифр и знаков препинания и вовсе нет?
б) Если в пункте а ваш ответ — 5, то найдите ошибку. Без какого символа нельзя обойтись?

10 ДНК человека (его генетический код) можно представить как слово в четырёхбуквенном алфавите — каждая буква обозначает некоторое звено цепи ДНК. Каков информационный объём ДНК, содержащей примерно $1,5 \cdot 10^{23}$ звеньев?

11 Сообщение, присланное от инопланетян, оказалось записанным с помощью всех символов их алфавита. Вот это сообщение:

$$\otimes_{\text{oc}} \nabla_X \downarrow \otimes \Xi \otimes \nabla_X 0 \bullet \nabla$$

Все символы алфавита кодируются одним и тем же минимально необходимым числом бит. Сколько бит информации в этом сообщении?

12 В марсианском алфавите 15 букв, в венерианском — 255 букв. Кроме того, в каждом языке есть символ «пробел», разделяющий слова. При передаче сообщений применяется двоичное кодирование, в котором каждый символ кодируется одинаковым числом бит. Жители этих планет обменивались сообщениями, содержащими одинаковое количество символов. Каждый при этом писал на своём языке, поэтому информационный объём сообщений оказался разным.

У какого сообщения информационный объём оказался больше и во сколько раз?

13 Рассмотрите свои ответы к заданиям 3 из § 2. Сколько бит, содержит сообщение о маршруте из клетки A в клетку B , о котором шла речь в задании 6а?

14 Сколько информации содержит следующая картинка, напечатанная компьютером?

[illegible]

15 Представьте себе, что из туманности Андромеды (одной из ближайших к нам галактик) пришло сообщение в виде последовательности радиосигналов двух видов (один из них мы обозначили нулём, а другой — единицей):

[illegible]

а) Сколько бит информации содержит это сообщение?

б) Многие учёные считают, что жители высокоразвитых цивилизаций обладают органами чувств, аналогичными нашему зрению. Используя это предположение в качестве предварительной информации о цивилизации, попытайтесь расшифровать сообщение. (Совет: разложите количество бит информации на простые множители.)

- 16 В системе связи скорость передачи — 200 символов в минуту. В течение 5 мин было передано сообщение объемом 375 байт. Сколько символов в алфавите?
- 17 Монитор работает в режиме с разрешением 1024×768 при глубине разрешения 32 бит и частоте обновления экрана 75 Гц. Какую минимальную пропускную способность должен поддерживать видеоадаптер, обеспечивающий работу монитора?
- 18 В системе связи скорость передачи составляет 128 000 бит/с. На передачу файла с несжатой монофонической музыкой потребовалось 2 минуты и 45 секунд. Укажите количество уровней квантования при оцифровке этой музыки, если известно, что её продолжительность составила 1 минуту и оцифровка производилась с частотой дискретизации 22 000 Гц.
- 19 Для хранения данных на диске 3,5" выделяется 2847 секторов, объемом 512 байт каждый. Какова длительность звукового файла, который уместится на такой диске при высоком качестве звука: стерео, 16 бит, 48 000 измерений в секунду? (Совет: имейте в виду, что каждый раз должны быть записаны данные всех 48 000 измерений.)
- 20* Два текста содержат одинаковое количество символов, хотя и записаны в двух разных алфавитах. Количество символов в каждом алфавите не превышает 20. Вася заметил, что, хотя количество символов в первом алфавите в 3 раза больше, чем во втором, информационный объем первого текста всего лишь в 2 раза больше, чем второго. После этого Вася решил подсчитать, насколько в первом алфавите символов больше, чем во втором. Какое максимальное и какое минимальное число могло у него получиться, если известно, что все символы, принадлежащие одному алфавиту, кодируются одним и тем же минимальным целым числом битов, но, возможно, своим для каждого алфавита?

6.2. □ Экономное кодирование. Алгоритмы сжатия информации

Использование одного и того же количества бит для кодирования всех символов алфавита весьма расточительно, ведь некоторые символы употребляются в сообщениях достаточно редко, а другие значительно чаще. Чтобы уменьшить информационный объём передаваемого сообщения, выгодно часто используемые символы кодировать меньшим количеством бит. Но чтобы такой неравномерный код позволял однозначно декодировать переданное сообщение, нужно, чтобы он обладал определёнными свойствами. Одним из условий, обеспечивающих однозначное декодирование, является префиксность кода: код называется префиксным, если код любого символа алфавита не является начальным фрагментом кода какого-либо другого символа. Многие алгоритмы сжатия информационного объёма сообщения построены на том, что более часто встречающиеся символы или комбинации символов кодируются более короткими битовыми последовательностями. Коэффициентом сжатия называется отношение исходного информационного объёма сообщения к информационному объёму того же сообщения после того, как к нему применено сжатие.

Для каждого двоичного кода может быть построено двоичное дерево¹, каждое ребро которого помечено одним из двух кодирующих символов. Последовательность символов от корня к любой вершине даёт слово, записанное в двухсимвольном алфавите. Код является префиксным в том, и только в том случае, когда все кодовые слова заканчиваются на листьях кодового дерева.

Алгоритмы сжатия бывают двух типов — без потери информации (обратимые алгоритмы) и с потерей информации. Алгоритмы первого типа используются, как правило, для сжатия символьной информации, алгоритмы второго типа — для сжатия видео- и аудиоинформации. К обратимым алгоритмам сжатия информации относится алгоритм Хаффмана. Он состоит в следующем:

1-й шаг — для каждого символа определяется количество его вхождений в сообщение;

2-й шаг — строится бинарное дерево, концевыми вершинами которого являются символы сообщения и каждой из них приписано количество вхождений этого символа, при построении новой вершины среди вершин, в которые ещё не входят рёбра, выбираются две вершины с наименьшими значениями; новая вершина соединяется рёбрами

¹ Понятие дерева обсуждается в § 17.

с выбранными двумя вершинами, и ей приписывается число, равное сумме чисел на концах построенных рёбер;

3-й шаг — рёбра дерева размечаются символами 0 и 1, начиная от корня;

4-й шаг — каждому символу исходного сообщения сопоставляется кодовое слово, прочитанное на маршруте от корня к концевой вершине, соответствующей этому символу.

Вопросы и задания

- 1 Какой код называется префиксным?
- 2* Объясните, почему сообщение, записанное в префиксном коде, допускает однозначное декодирование. Опишите алгоритм декодирования для префиксного кода.
- 3 а) На рисунке 6.1 изображено двоичное дерево некоторого кода. Для каждой буквы запишите соответствующее ей кодовое слово. Является ли этот код префиксным?

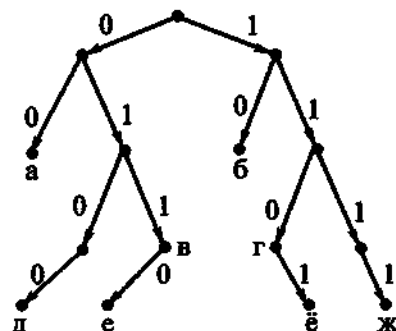


Рис. 6.1

- б) «Перестройте» (т. е. не увеличивая количества вершин) это дерево так, чтобы для тех же 8 букв получился префиксный код.
- 4 а) Дан набор кодовых слов: 00, 10, 010, 101, 110, 1001, 1011, 1101, 1111. Постройте для этого кода соответствующее ему дерево. Является ли этот код префиксным?
б) Выполните такое же задание для кода 00, 01, 10, 011, 100, 101, 1001, 1010, 1111.
в) Выполните такое же задание для кода 00, 10, 010, 110, 0110, 0111, 1110, 1111.
- 5 Для кодирования 7 букв русского алфавита и пробела использовался код, представленный в таблице:

Буква	Пробел	А	В	В	Г	Д	Е	Ж
Код	01	111	001	1100	0000	10	0001	1101

- а) Проверьте, что этот код префиксный.
- б) Закодируйте сообщение «ГДЕ ЖЕ ЕДА».

в) Декодируйте сообщение

10000110010010001101110100001000010100111100001111101.

- 6 Для кодирования цифр предложено пять кодов: А, В, С, D и Е (см. таблицу).

	А	В	С	Д	Е
0	1110	0010	1101	0011	0010
1	1010	101	1011	1101	101
2	111	0011	1010	011	0011
3	000	0000	0100	1100	0001
4	0010	011	111	1110	0101
5	100	11	001	000	1000
6	0011	0100	000	0010	0100
7	1101	100	100	10	11
8	1011	0001	010	010	100
9	01	0101	011	1111	011

- а) Для каждого из этих кодов постройте кодовое дерево.
 б) Укажите, какие из предложенных кодов являются префиксными.
 в) Закодируйте каждым из найденных вами префиксных кодов следующие числа: 213; 1000; 8642.
 г) После кодирования некоторого числа одним из найденных вами префиксных кодов получилась последовательность 100100110010000. Определите, каким кодом осуществлено кодирование, и восстановите исходное число.
 д) Выполните такое же задание, как в пункте г, для последовательности 01101000101110.
 е) Выберите какое-нибудь не более чем четырёхзначное число и закодируйте его одним из префиксных кодов, предложенных в таблице. Передайте получившийся код соседу по парте для декодирования. Проверьте, правильно ли он провёл декодирование (для этого пусть он укажет код, которым пользовался).

- 7 а) Постройте код Хаффмана для фразы «НА ДВОРЕ ТРАВА, НА ТРАВЕ ДРОВА».
 б) Определите коэффициент сжатия для данной фразы, считая, что исходно каждый символ кодировался в ASCII.

- 8 а) Постройте код Хаффмана для фразы «КАК БЕЗДНА ЗВЕЗД БЕЗ ДНА НАД НАМИ НЕБО».
- б) Определите коэффициент сжатия для данной фразы, считая, что исходно каждый символ кодировался в UNICODE.

- 9 При кодировании графической информации довольно много идущих подряд пикселей имеют, как правило, один и тот же цвет и потому кодируются одним и тем же набором бит. Естественной идеей уменьшения информационного объёма является простое указание, сколько пикселей, идущих друг за другом, имеет данный цвет. К примеру, монохромная картинка на условном экране размером 9×10 пикселей, изображённая на рисунке 6.2 (каждая клетка — это один пиксель), может быть представлена следующим сообщением:

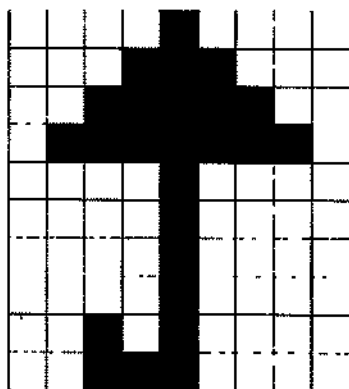


Рис. 6.2

4Б147Б345Б543Б745Б148Б148Б146Б141Б146Б344Б.

а) Проверьте, что указанное сообщение действительно кодирует данное изображение.

б) Какое изображение кодируется на том же экране сообщением

4Б147Б347Б347Б347Б347Б345Б544Б543Б341Б342Б341Б341Б?

А сообщением

3Б345Б543Б741Б941Б742Б243Б242Б243Б242Б243Б242Б742Б741Б ?

в) Нарисуйте какую-нибудь монохромную картинку на таком же экране (9×10 пикселей), закодируйте её соответствующим сообщением и предложите соседу по парте декодировать её. Если результат декодирования не совпал с исходным рисунком, выясните, кто из вас допустил ошибку.

- 10 Если каждый пиксель монохромного экрана кодируется 1 битом (1 — код белого цвета, 0 — код чёрного цвета), то информационный объём сообщения о картинке, изображённой на рисунке 6.2, составит 900 бит.

а) Какой объём имеет сообщение

4Б147Б345Б543Б745Б148Б148Б146Б141Б146Б344Б,

которым представлена та же картинка, если считать, что каждая буква и каждая цифра в нём кодируется 1 байтом? Каков коэффициент сжатия при таком преобразовании исходного сообщения о состоянии данного монохромного экрана?

б) Поскольку используются только буквы Б и Ч и числа от 1

до 9, то каждую пару число — буква можно кодировать пяти-битовой последовательностью, в которой первые четыре бита отведены под код цифры, а последний бит — это код цвета. Например, 4Б кодируется как 01000. Какой объём будет иметь сообщение после такой перекодировки? Чему в этом случае равен коэффициент сжатия по отношению к объёму первоначального сообщения о состоянии данного монохромного экрана?

- 11 Идея метода сжатия информации, высказанная в задании 9, носит название RLE-метода упаковки (от английского *Run-Length Encoding*)¹. Алгоритм сжатия сообщения методом RLE состоит в том, что каждая последовательность повторяющихся байтов заменяется двумя байтами: первый из них — управляющий — начинается с 1, а в остальных семи битах указано число (в двоичном виде) повторений байта из сообщения, второй байт — это точное воспроизведение повторяющегося байта. В начале каждой последовательности неповторяющихся байтов также добавляется управляющий байт, начинающийся с 0, а в остальных семи битах указано число (в двоичном виде) неповторяющихся байтов. После такого управляющего байта воспроизводится цепочка неповторяющихся байтов. Например, в последовательности (для удобства байты в ней отделены друг от друга пробелом, в реальном сообщении пробелов, разумеется, нет)

```
11111111 11111111 11111111 11111111 11111111 10101010 1
1010100 10101010 11010100 11010100 11010100
```

первые пять байтов заменяются на 1000010111111111, последующий блок из трёх неповторяющихся байтов заменяется на 00000011101010101101010010101010, наконец, последний блок из трёх повторяющихся байтов заменяется на 1000001111010100. В результате получается более короткая последовательность 10000101111111110000001110101010110100101010101000001111010100.

а) Используя RLE-метод, уменьшите объём следующего сообщения:

```
010001100100011001000110010001101111111111010101011010
10011010100110101001101010011010100111111111111111111
111111.
```

Каков коэффициент сжатия для этого сообщения?

б) После применения RLE-метода было получено сообщение

```
10000011011101110000000111111111100001101010101000000
0101010101011010100.
```

Распакуйте это сообщение. Подсчитайте коэффициент сжатия.

в) Используя RLE-метод, уменьшите объём сообщения, при-

¹ Этот метод используется в графических форматах PCX и BMP.

ведённого в задании 15 из § 6.1 (чтобы получить целое число байт, добавьте в конец этого сообщения ещё три 1).

г) После применения RLE-метода было получено сообщение
111110110111011100000011111111111000011010101010111
11101010101101011101110110.

Какой информационный объём имело исходное сообщение?
Каков коэффициент сжатия?

д)* Как применить RLE-метод, если количество идущих подряд одинаковых байт больше 127?

- 12) Алгоритм Хаффмана можно применить к любой битовой последовательности. Для этого последовательность разбивают на фрагменты и каждый получивший фрагмент рассматривают как код некоторого символа. Затем для этих «символов» применяют алгоритм Хаффмана и строят новый код. Если, например, последовательность

111111111111111111111111111111111101010101101010010101
010110 11111010011011111010

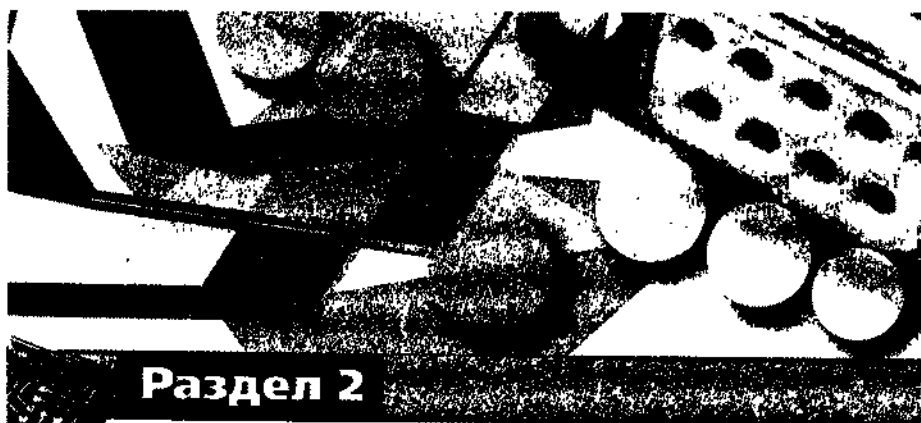
разбить на четырёхбитовые фрагменты, то получится следующий набор: 1111 (встречается 10 раз), 1010 (встречается 4 раза), 1101 (встречается 3 раза), 0100 (встречается 2 раза), 010 («хвостик» из трёх бит — встречается 1 раз).

а) Постройте по этим данным код Хаффмана для указанного набора битовых фрагментов. Закодируйте этим кодом данную последовательность. Подсчитайте коэффициент сжатия.

б) Разбейте данное сообщение на трёхбитовые фрагменты и выполните те же задания, что и в пункте а.

в) Разбейте сообщение, приведённое в задании 15 из § 6.1, на семибитовые фрагменты и выполните для него те же задания, что и в пункте а.

- 13) Для передачи по каналу связи сообщения, состоящего только из букв А, Б, В, Г, решили использовать неравномерный код: А = 01, Б = 110, В = 101. Как нужно закодировать букву Г, чтобы длина кода этой буквы была минимальной, а любое сообщение, переданное в этом коде, допускало однозначное декодирование?



Раздел 2

Основные информационные технологии

Под технологией обычно понимают процесс, обеспечивающий гарантированное получение нужного продукта из исходного материала. Информационная технология — процесс, использующий совокупность средств и методов обработки исходной информации для изменения её содержания или формы её представления.

18

7

Обработка текстовой информации

В информатике текстом принято называть любую последовательность знаков некоторого алфавита. Текст является носителем информации, но содержательное преобразование этой информации, как правило, не относят к обработке текста. Иными словами, под обработкой текста средствами информационной технологии обычно понимают такое его преобразование, которое не меняет, по существу, его информационное содержание. В ходе такой обработки исправляются орфографические, синтаксические и стилистические ошибки, удаляются ненужные повторы, вводятся уточняющие слова и т. п. Текст структурируется, т. е. разбивается на разделы (главы, параграфы, пункты) и абзацы. Наконец, в текст вставляются иллюстрации — чертежи, рисунки, диаграммы, графики и т. п. Структурные элементы текста тем или иным образом располагаются на странице — центрируются заголовки, определённым образом располагаются эпиграфы к главам или параграфам,

выравниваются (или, наоборот, не выравниваются) края текста, выделяются каким-либо образом основные положения и т. д.

Вопросы и задания

- 1  а) Наберите с помощью клавиатуры слова новогодней песенки «Ёлочка» (или какое-нибудь другое стихотворение, которое вы помните), причём заголовок наберите заглавными буквами и каждую строку начинайте тоже с заглавной буквы. Располагайте текст от левого края экрана. Используйте при работе вставку, удаление и т. д. Сохраните набранный текст в файле.
б) Отделите все строки друг от друга, вставляя каждый раз пустую строку.
в) Используя операцию разбиения строки, сделайте так, чтобы в каждой строке осталось только одно слово, и расположите эти слова «лесенкой» (как у Маяковского).
г) Соедините строки так, чтобы они стали такими, как до разбиения. Совпало ли то, что у вас получилось, с первоначально набранным текстом? (Для сравнения можно вызвать в другое окно текст, который вы сохранили, выполнив задание пункта а.)
- 2  Наберите текст детского стихотворения, не забывая в нужный момент переходить на латинский шрифт и обратно.
а) Я шалил: разбилась ваза.
— Кто разбил? — спросил мой father.
— Это бабушкина ваза..., —
Прослезилась моя mother.
Тут за меня вступилась sister:
— Осколки он убрал, всё чисто!
Потом сказала: «Father, mother,
Шалить не будет больше brother».
б) Ну, почему считает папа,
Что все медведи косолапы?
Нет, мой медведь не косолап,
Я попросил его: «Stand up!»
Он встал... на две кривые лапы.
«Садись — sit down!» Прав мой папа...
- 3  Наберите текст благодарственного письма от руководства школы вашим родителям. В оформлении можно следовать ниже приведённому образцу, заменяя слова, стоящие в скобках, на те, которые соответствуют, на ваш взгляд, действительности. Используйте шрифты различной гарнитуры и различного кегля.

БЛАГОДАРСТВЕННОЕ ПИСЬМО

Уважаемые (имя и отчество родителей)!

Дирекция (школы, лицея, гимназии) № ____ (номер) г. ____ (название города) благодарит вас за воспитание вашего (сына, дочери), ____ (фамилия, имя), которое выражается (в хорошей успеваемости, активной общественной деятельности, примерном поведении, добром отношении к товарищам и т. п.).

Директор (школы, гимназии, лицея) № ____ (номер)
____ (Ф.И.О.)

- 4 Создайте таблицу, в которой бы хранилась информация о некоторых ваших друзьях-одноклассниках (например, по приведённому ниже образцу).

Фамилия и имя	Дата рождения	Адрес	Телефон	Имя и отчество родителей
Иванов Саша	13.03.88	ул. Гагарина, д. 15, кв. 107	777-77-77	Тамара Михайловна, Николай Фёдорович
...	...	...	...	...

§

8

Обработка числовой информации с помощью электронной таблицы

Одним из инструментов технологии обработки числовой информации являются электронные таблицы. Основное назначение электронной таблицы — обработка числовой информации, представленной в табличной форме, при этом часть информации является исходной (или, как говорят, независимой), другая — вычисляемой (зависимой). Вся информация в электронной таблице размещается в её ячейках, визуально представленная как клетки таблицы, образованные пересечением строк и столбцов. Столбцы электронной таблицы обычно поименованы буквами латин-

ского алфавита, а строки — перенумерованы. Каждая ячейка, тем самым, получает адрес, состоящий из обозначения столбца и номера строки.

В каждой ячейке может находиться либо текст, либо число, либо формула. Под формулой при этом понимается арифметическое выражение, содержащее числа, адреса ячеек (или их имена, если электронная таблица допускает именование ячеек), и встроенные функции (среди которых имеются логические, поэтому формула может иметь вид оператора ветвления), соединённые знаками арифметических операций. Набор встроенных функций, вообще говоря, для каждой электронной таблицы свой, но, как правило, среди них имеются функции вычисления квадратного корня, экспоненты и логарифма, тригонометрических функций, логических функций (и, или, не, если) и т. д. Кроме того, имеются функции, работающие с блоком ячеек. Таковым называют совокупность всех ячеек, составляющих некоторый прямоугольник. Блок ячеек обычно указывается адресом ячейки в верхнем левом углу и (через двоеточие) адресом ячейки в правом нижнем. К функциям, допускающим обработку блока ячеек, относятся функции нахождения суммы, среднего значения, произведения и т. п.

Среди операций, которые можно выполнять над электронной таблицей, всегда присутствуют операции вставки и удаления строк и столбцов, копирования блока ячеек, очистка содержимого блока ячеек и т. п. При этом в электронной таблице действует так называемый принцип относительной адресации. Он означает, что адреса ячеек в формуле определены не абсолютно, а относительно той ячейки, где расположена формула. Поэтому при копировании содержимого ячейки или блока ячеек в другое место автоматически пересчитываются адреса ячеек, фигурирующих в формулах копируемого фрагмента. Таким образом, относительная адресация проявляет себя в том, что всякое изменение места расположения формулы приводит к автоматическому пересчёту адресов ячеек, фигурирующих в этой формуле.

Если же в копируемых формулах всё время должна использоваться одна и та же ячейка (например, в ней хранится нужная константа) или ячейки одной и той же строки, то адрес такой ячейки или соответственно номер строки помечается в формуле как абсолютный. Обычно для этого применяется зарезервированный символ (довольно часто знак \$).

Одной из важных возможностей современных электронных таблиц является построение графиков и диаграмм по данным, рассчитываемым с помощью этой таблицы.

Вопросы и задания

- 1 Сколько ячеек содержит блок E3:H6? А блок Y7:AB9?
- 2 В некоторой ячейке электронной таблицы записано выражение: а) $=9 - 6*5/3*2$; б) $=(13 - 7)*8/3*2$.
Какое значение будет записано в этой ячейке?
- 3 В ячейке A1 электронной таблицы требуется вычислить значение суммы чисел, стоящих в блоке ячеек от B2 до E8. Какую формулу надо записать в ячейку A1?
- 4 В электронной таблице значение формулы $=CP3HA4(A6:D6)$ равно 2. Чему равно значение формулы $=СУММ(A6:C6)$, если в ячейке D6 записано число -5?
- 5 В электронной таблице в ячейках A7, B7 и C7 записаны соответственно формулы $=МИН(A1:A6)$, $=МИН(B1:B6)$, $=МИН(C1:C6)$, а в ячейку D7 записана формула $=МАКС(A7:C7)$. После того как в ячейке B3 записанное там число поменяли на 7, изменилось и число в ячейке D7. Можно ли по этой информации определить, какое число теперь оказалось в ячейке D7? Если можно, то укажите это число, если нет, то объясните почему.
- 6 В ячейку B1 записана формула:
 $=ЕСЛИ(МИН(A1:A4) < 0; СУММ(A1:A4); ПРОИЗВЕД(A1:A4))$
Определите, чему равно значение в ячейке B1, если о числах, записанных в ячейках A1—A4, известно, что:
а) их сумма равна -3, а произведение равно 12;
б) их сумма равна 3, а произведение равно -6;
в) их сумма равна 0, а произведение равно 4;
г)* их сумма равна 5, а произведение равно 2,5.
- 7 Первоначально ячейки электронной таблицы были заполнены числами так, как показано в таблице. Затем в ячейку A1 записали формулу $=C\$1 + \$B3$ и скопировали её в ячейку B2. Какое значение будет записано в ячейке B2?

	A	B	C	D	E
1	2	3	5	7	11
2	1	4	9	12	8
3	3	7	1	6	5
4	5	2	8	5	1
5	2	6	4	2	1

- 8 Дан фрагмент таблицы в режиме отображения формул. Затем формула из ячейки A5 была скопирована в ячейки A6, B5 и B6. Какое значение окажется в ячейке B6 после того, как будут выполнены вычисления?

	A	B
1	1	2
2	2	4
3	3	6
4	4	8
5	=СУММ(\$A1:A4)	

- 9 а) С помощью электронной таблицы составьте для аргумента x , меняющегося на отрезке от -3 до 2 с шагом $0,1$, таблицу значений функции $y = x^4 + 6x^3 + 2x^2 - 18x + 4$.
 б) Используя составленную таблицу значений, постройте с помощью электронной таблицы график этой функции. Для каждого из корней уравнения $x^4 + 6x^3 + 2x^2 - 18x + 4 = 0$, имеющих на отрезке от -3 до 2 , определите промежутки, на которых они находятся.
 в) П Используя режим «Подбор параметра», найдите все корни уравнения $y = x^4 + 6x^3 + 2x^2 - 18x + 4 = 0$, расположенные на отрезке от -3 до 2 . Точность корня должна быть не менее $0,001$.
 г) П У этого уравнения имеется корень, расположенный за пределами отрезка $[-3; 2]$. Найдите этот корень с точностью до $0,001$.
- 10 П С помощью электронной таблицы найдите с точностью до $0,001$ все решения уравнения $x = 3 \cos x$.
- 11 П а) Требуется найти максимальное значение выражения $5x^2 - 6y^2$ для тех положительных чисел x и y , которые удовлетворяют неравенствам $x - 2y \leq 4$, $2xy \geq 7$, $x + y \leq 100$. С помощью электронной таблицы решите эту задачу. (Совет: воспользуйтесь надстройкой «Поиск решения».)



9

Базы данных и информационно-поисковые системы

База данных — организованная совокупность данных, предназначенная для длительного хранения во внешней памяти компьютера и многократного использования.

Каждая база данных состоит из двух компонентов: содержимого и программной оболочки. Оболочка представ-

ляет постоянную часть базы данных и предназначена для обработки содержимого базы. Эту оболочку обычно называют системой управления базой данных (СУБД). Система управления базой данных играет роль, аналогичную языку программирования. Можно считать, что это некоторая система команд, обеспечивающая организацию данных и манипулирование с ними. Главная функция СУБД — давать ответы на запросы пользователя, иными словами, удовлетворять его информационную потребность. Кроме того, СУБД позволяет изменять (как говорят, редактировать) записи, хранящиеся в базе данных, вносить новые сведения и выполнять другие операции над данными, например упорядочивать их по значениям заданного атрибута (так называемый режим сортировки данных).

Каждая база данных ориентирована на определённую предметную область, в рамках которой рассматриваемые в ней объекты или явления характеризуются конкретным набором свойств, или, как говорят в теории баз данных, набором атрибутов (полей).

Некоторые из атрибутов (полей) могут быть объявлены ключевыми. По таким атрибутам можно, в частности, делать сортировку.

По способу организации связей между атрибутами различают три вида баз данных: иерархические, реляционные и сетевые. В иерархических базах данных атрибуты соподчинены между собой так, что данные образуют древовидную структуру.

Сетевая база данных представляет собой более общую структуру, нежели иерархическая база, — в ней элементы связаны не только от одного уровня к следующему, а, вообще говоря, произвольным образом.

Реляционная база данных — это совокупность связанных между собой таблиц, столбцы которых задаются атрибутами данных, а каждая строка состоит из значений атрибутов для конкретного объекта.

Разновидностью баз данных являются информационно-поисковые системы (ИПС). Так обычно называют базы данных, в которых пользователю предоставляется возможность лишь получать ответы на запросы, при этом форма запросов фиксирована разработчиками и тоже не может изменяться пользователем.

Вопросы и задания

- 1 Что такое база данных? Что такое СУБД?
- 2 Для чего предназначена операция сортировки в БД?

- 3 В ИПС «Государства» хранятся следующие сведения о государствах Европы:

- название государства — атрибут **Название**;
- площадь, км² — атрибут **Площадь**;
- население, тыс. чел. — атрибут **Население**;
- название столицы — атрибут **Столица**;
- наименование денежной единицы — атрибут **Деньги**;
- государственный язык — атрибут **Язык**;
- форма государственного устройства — атрибут **Правление**;
- вхождение в военно-политический или экономический блок (НАТО, ЕЭС и т. п.) — атрибут **Блок**.

В запросе к этой ИПС указывается название атрибута и значения, по которым требуется провести поиск. Отношения между атрибутом и значением описываются знаками =, ≠, а для атрибутов с числовыми значениями могут также использоваться знаки >, <, ≥ и ≤. Например, чтобы найти все страны с населением больше 100 000 человек, составляется запрос

Население > 100,

а для нахождения стран с парламентской формой правления, не входящих ни в какие блоки, потребуется запрос

Правление = "парламентская республика" и **Блок** = "нет".

Составьте запросы, позволяющие найти

- а) все страны-карлики (площадь меньше 50 000 км²);
- б) все страны, которые не пользуются евро в качестве государственной валюты;
- в) все страны с площадью более 10 000 км² и населением менее 1 млн человек;
- г) все страны с населением больше 2 млн человек, кроме франкоговорящих;
- д) все страны ЕЭС, не входящие в НАТО;
- е) все европейские монархии (напомним, что в Европе есть четыре вида монархий: абсолютная, конституционная, парламентская и княжеская).

- 4 Какой запрос ИПС «Государства» (см. задание 3) надо сформулировать, чтобы выяснить:

- а) все ли государства с населением больше 1 млн человек входят в ЕЭС;
- б) во всех ли государствах, входящих в ЕЭС, проживает больше 1 млн человек?

- 5 Однажды школьник решил воспользоваться ИПС «Государства» (см. задание 3), чтобы найти все конституционные и абсолютные монархии. С этой целью он составил следующий запрос:

Правление = "Конституционная монархия" и

Правление = "Абсолютная монархия".

В ИПС не оказалось стран, удовлетворяющих этому запросу. Объясните почему. Какой запрос надо составить, чтобы получить нужную школьнику информацию?

- 6 База данных о поступивших в магазин фруктах представлена таблицей

№ п/п	Товар	Страна	Цена	Общий вес
1	Мандарины	Пакистан	57	200
2	Апельсины	Марокко	38	450
3	Ананасы	Мадагаскар	65	450
4	Апельсины	Испания	42	450
5	Киви	Индонезия	33	250
6	Маракуйя	Алжир	112	200
7	Киви	Бразилия	35	350
8	Бананы	Индонезия	28	200
9	Мандарины	Алжир	48	350

Данная база была отсортирована по следующему принципу: сначала по возрастанию поля *Количество*, затем для одинаковых значений в поле *Количество* — по убыванию поля *Товар*, затем для одинаковых значений в поле *Товар* — по возрастанию поля *Цена*. Какой товар окажется в строке под номером 6?

- 7 П Представьте, что вы руководитель небольшой фирмы. Вам необходимо иметь ту или иную информацию о своих работниках. Разработайте двухтабличную базу данных: одна таблица — «Штатное расписание» — содержит информацию о должностях, окладах, отделе, которому приписана данная должность; другая таблица — «Сотрудники» — содержит информацию о работающих у вас сотрудниках (дата приёма на работу, образование, дата рождения), об их должностях и рабочем месте (название отдела, номер комнаты, рабочий телефон). Свяжите эти таблицы по подходящему ключевому атрибуту. С помощью данной базы данных получите ответы на вопросы:
- У кого из сотрудников имеется высшее образование?
 - Кто самый молодой сотрудник вашей фирмы?
 - Какие вакансии (т. е. незанятые должности) имеются в вашей фирме?
 - Кто из работников в этом году начнёт получать пенсию по старости (женщины выходят на пенсию в 55 лет, а мужчины — в 60)?

д) У кого из сотрудников вашей фирмы стаж работы не меньше 10 лет?

е) Вам срочно потребовалась консультация одного из ваших специалистов, а у него сегодня выходной. Найдите номер его домашнего телефона (заодно узнайте его имя и отчество).

ж) К празднику 8 Марта вы решили лично поздравить женщин — начальников отделов вашей фирмы. Поручите СУБД представить вам нужный список.

Ваша фирма (см. задание 7) ведёт активную работу с иногородними клиентами. Поэтому ваши сотрудники ведут много междугородних переговоров. Прежде всего это сотрудники отделов сбыта и рекламы, но не исключено, что и у сотрудников других отделов (например, бухгалтерии) тоже возникает необходимость в телефонных переговорах. Информация о всех состоявшихся телефонных звонках ежемесячно поступает из телефонной компании: сообщается дата, номер телефона, продолжительность разговора и пункт, с которым вёлся телефонный разговор. Дополните уже разработанную вами базу данных таблицей о телефонных разговорах с указанной информацией. Установите связь новой таблицы с уже имеющимися. Создайте запросы и выполните действия, позволяющие получить ответы на следующие вопросы:

а) С какими пунктами вёл телефонные переговоры ваш отдел сбыта?

б) В какие дни были переговоры с нужным вам пунктом и кто из сотрудников эти переговоры вёл?

в) Велась ли с телефонов вашей фирмы неслужебные разговоры (т. е. с пунктами, в которых нет ваших партнёров и клиентов) и кто из сотрудников такие разговоры вёл?

г) С каким из клиентов переговоры были наиболее интенсивными (по количеству переговоров за месяц)?

д) С каким из клиентов переговоры были наиболее продолжительными (по суммарному времени всех переговоров за месяц)?



Раздел 3

Алгоритмизация, структуры данных и элементы программирования

181

10

Понятие алгоритма и исполнителя. Линейные алгоритмы

Алгоритм — приводящая к определённому результату последовательность действий, допустимых для некоторого исполнителя.

Исполнитель — субъект или объект, выполняющий действия согласно предписанной ему инструкции. Исполнитель может быть формальным или эвристическим. Формальный исполнитель выполняет инструкцию, не вникая в её смысл и не принимая во внимание цель, с которой она составлена. Эвристический исполнитель согласовывает свои действия с целевыми установками и принимает решение о том, как выполнить то или иное действие, сообразуясь со смыслом инструкции.

Каждый исполнитель полностью характеризуется своим набором допустимых действий и средой, в которой он существует.

Запись алгоритма состоит из команд, каждая из которых указывает исполнителю, какое допустимое действие тот должен совершить, получив эту команду. Набор всех команд исполнителя называется системой команд исполнителя (СКИ). Кроме этого, в записи алгоритма присутствуют порядки алгоритмических конструкций, определяющие порядок выполнения команд алгоритма. При отсутствии таких операторов команды, из которых составлен алгоритм, будут исполняться одна за другой в порядке их записи в алгоритме. Такой алгоритм называется линейным.

В записи алгоритма могут присутствовать комментарии, поясняющие человеку, читающему данный алгоритм, для чего предназначено данное действие в алгоритме. Комментарии в записи алгоритма условимся выделять круглыми скобками со звёздочками, например: (* комментарий *).

Среда исполнителя — это совокупность объектов и связей между ними, над которыми данный исполнитель может совершать допустимые для него действия.

Совокупность тех результатов, которые можно получить с помощью данного исполнителя, называется его **достижимыми целями**.

Вопросы и задания

- 1 Попытайтесь определить, что является средой для:
 - а) системы управления базами данных;
 - б) станка с числовым программным управлением;
 - в) автопилота;
 - г) графического редактора;
 - д) программируемого видеомэгнитофона.
- 2 Насколько можно говорить о том, плох формальный исполнитель или хорош? Например, что означает фраза «Я недоволен электронной таблицей»? Означает ли это, что формальный исполнитель не справляется с предназначенными для него алгоритмами?
- 3 а) В приведённых ниже примерах укажите преимущественный, на ваш взгляд, вид обработки информации — алгоритмический или эвристический:
 - вывод формулы корней кубического уравнения;
 - определение кислотности среды с помощью лакмуса;
 - определение очередного хода в середине шахматной партии;
 - определение величины сопротивления цепи электрического тока при наличии амперметра и вольтметра;
 - нахождение корней кубического уравнения по заданным формулам.
 б) Проанализируйте свою обычную ежедневную деятельность по обработке информации. С каким видом обработки — алгоритмическим или эвристическим — вам чаще приходится иметь дело? Проанализируйте с той же точки зрения вашу учебную деятельность на уроках по разным предметам.
- 4 Пусть дан угол ABC (B — вершина угла, A и C — точки на сторонах угла). Определите, для решения какой задачи предназначен следующий алгоритм:
Поставить ножку циркуля в точку B .

Установить раствор циркуля равным длине отрезка BA .

Провести окружность.

Поставить ножку циркуля в точку пересечения этой окружности с лучом BC .

Провести окружность.

Поставить ножку циркуля в точку A .

Провести окружность.

Провести прямую через точку B и вторую точку пересечения тех же окружностей.

- 5 Робот записывает цепочки символов. Каждая цепочка записывается в отдельную строку. Первая строка содержит цепочку из четырёх цифр: 1357. Каждая последующая строка содержит цепочку, которая получается из предшествующей цепочки по следующему правилу: записывается предшествующая цепочка и к ней справа приписывается такая же цепочка, но в которой каждый символ дважды подряд повторяется, затем последний символ заменяется на номер строки. Вот первые три строки:

1357

135711335572

135711335572113355771111333355557723

а) Какая пара цифр будет стоять на 999-м и 1000-м местах в цепочке, записанной в шестой строке?

б) Какая пара цифр будет стоять на 9999-м и 10 000-м местах в цепочке, записанной в девятой строке?

- 6 а) Имеется два кувшина ёмкостью 7 и 12 литров. Исполнитель ВОДОЛЕЙ может набирать воду из реки в каждый кувшин, выливать из него воду и определять, налита ли вода в кувшине доверху. Составьте алгоритм, выполнив который ВОДОЛЕЙ наберёт из реки 10 литров воды.

б) Может ли ВОДОЛЕЙ набрать 7 литров, если в его распоряжении два кувшина ёмкостью 6 и 10 литров?

в)* Имеется два кувшина ёмкостью a литров и b литров (a и b — натуральные числа). Исполнитель ВОДОЛЕЙ может набирать воду из реки в каждый кувшин, выливать из него воду и определять, налита ли вода в кувшине доверху. Результатом работы исполнителя ВОДОЛЕЙ является количество воды, которое он может получить в меньшем по ёмкости кувшине. Укажите достижимые цели для данного исполнителя в зависимости от исходных значений a и b .

- 7 Имеется три кувшина ёмкостью 4, 9 и 12 литров. С ними управляется исполнитель ВОДОЛЕЙ, который описан в задании 6. Составьте алгоритм, выполнив который ВОДОЛЕЙ наберёт из реки 11 литров воды.

- 8 а) На клетчатой бумаге изображено несколько многоугольников, вершины которых располагаются в вершинах клеток (рис. 10.1). Составьте для каждой из этих фигур алгоритм

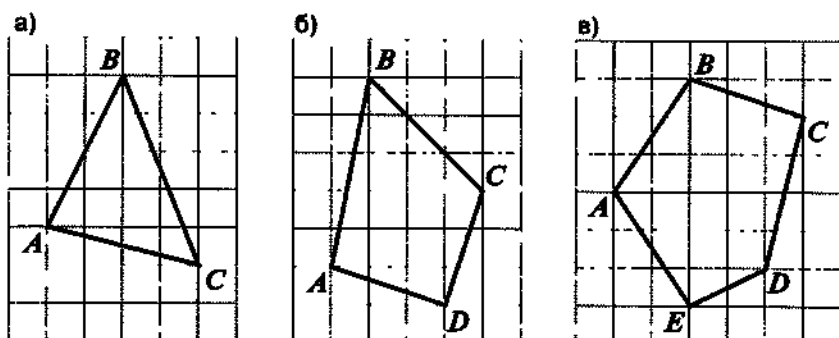


Рис. 10.1

нахождения с помощью карандаша и линейки без делений центра тяжести фигуры. (Совет: воспользуйтесь тем, что центр тяжести треугольника — это точка пересечения его медиан.)

б) Составьте алгоритм нахождения с помощью карандаша и линейки без делений центра тяжести любого треугольника, изображённого на клетчатой бумаге так, что его вершины располагаются в вершинах клеток.

в) Нарисуйте на клетчатой бумаге какой-либо треугольник, вершины которого располагаются в вершинах клеток. Предложите соседу по парте найти центр тяжести треугольника с помощью составленного им алгоритма. Проверьте правильность исполнения алгоритма и правильность результата.

9 Робот перемещается по клетчатому полю, переходя из клетки в соседнюю клетку по соответствующей команде: вверх, вниз, вправо, влево. Между некоторыми клетками поля установлены стены. Если Робот попытается пройти сквозь стену, то он разрушится и исполнение алгоритма прервётся.

а) Робот, начав с некоторой клетки, успешно исполнил следующий алгоритм:

влево
влево
вверх
вправо
вниз
вверх
вправо

Составьте алгоритм, содержащий наименьшее число команд, после исполнения которого Робот не разрушится и вернётся из той клетки, в которой оказался, в исходную клетку.

б) Выполните задание, аналогичное пункту а, для следующего алгоритма:

вправо
вниз

вправо
вниз
влево
вверх
влево
вниз
влево
вверх
вправо
влево
вверх

- 10) Исполнитель умеет умножать число на 3 и увеличивать число на 1.
а) Составьте для этого исполнителя алгоритм получения числа 1000 из единицы.
б) Сколько действий в самом коротком из таких алгоритмов? Ответ обоснуйте.
- 11) Исполнитель умеет умножать число на 3 и уменьшать число на 1.
а) Составьте для этого исполнителя алгоритм получения числа 2000 из единицы.
б) Сколько действий в самом коротком из таких алгоритмов? Ответ обоснуйте.
- 12* Исполнитель КВАДР имеет два допустимых действия:
а) возвести число, записанное на табло исполнителя, в квадрат;
б) вычесть 3 из числа, записанного на табло исполнителя.
После исполнения каждого действия на табло записывается результат. Команду на исполнение первого действия будем обозначать **К**, второго — **В**. Последовательность **КВ** означает, что сначала число возводится в квадрат, а из результата вычитается 3; запись **ВК** означает, что сначала из числа вычитается 3, а затем результат возводится в квадрат.
а) Составьте для этого исполнителя алгоритм получения числа 34 из числа 5.
б) Сколько действий в самом коротком из таких алгоритмов? Ответ обоснуйте.

Среда, в которой действует исполнитель, может накладывать ограничения на возможность выполнения того или иного действия или последовательности действий. Чтобы исполнитель мог учитывать обстановку, сложившуюся в окружающей его среде, в том числе и в результате его

собственных действий, в число допустимых действий исполнителя должны входить действия по проверке условий, относящихся к обстановке, имеющейся к данному моменту в среде, где находится исполнитель. Если условие выполнено, то исполнитель осуществляет некоторую последовательность действий, если же условие не выполнено, то эта последовательность действий не исполняется, а исполнитель либо совершает другую последовательность действий, либо вообще переходит к исполнению действий, предусмотренных алгоритмом после первой последовательности. Конструкция, реализующая в алгоритме указанную возможность выбора исполняемой последовательности действий в зависимости от условия, называется ветвлением.

Кроме того, к ветвлениям относится и конструкция **выбор**. Суть этой конструкции состоит в том, что в зависимости от значения условия-селектора исполняется та последовательность действий, которая помечена данным значением.

В алгоритмах мы будем записывать ветвление в одной из двух форм:

Если (условие) **то** { действие; действие; ... }

— неполная форма ветвления, или

Если (условие) **то** { действие; действие; ... }

иначе { действие; действие; ... }

— полная форма ветвления.

Фигурные скобки показывают, какая последовательность действий исполняется, если условие выполнено — она записана после слова **то**, и какая исполняется, если условие не выполнено — эта последовательность записана после слова **иначе**. Фигурные скобки, используемые для обрамления группы действий в алгоритме, называют операторными скобками.

В записи алгоритма конструкцию **выбор** договоримся оформлять так:

Выбор

при (условие_1): { действие; действие; ... }

при (условие_2): { действие; действие; ... }

...

при (условие_n): { действие; действие; ... }

иначе { действие; действие; ... }

Ветвь, начинающаяся словом **иначе**, может отсутствовать.

Нередко исполнителю необходимо повторить несколько раз какую-либо последовательность действий. То, сколько раз придётся повторить данную последовательность,

может быть известно заранее, а может определяться выполнением условия, которое проверяется каждый раз перед выполнением последовательности действий или сразу после этого.

Алгоритмическая конструкция, позволяющая указать исполнителю на повторяемость последовательности действий, называется **циклом**. Последовательность действий, которую исполнитель будет выполнять в цикле, называется **телом цикла**.

Основные виды циклов: цикл простого повторения (цикл повторить n раз), цикл с предусловием (цикл пока), цикл с постусловием (цикл до), цикл со счётчиком (цикл от... до... с шагом...). Вид цикла указывается в его заголовке.

Цикл с предусловием — это цикл, в котором действия, составляющие тело цикла, исполняются, пока остаётся истинным условие, записанное в заголовке цикла. В алгоритмах цикл с предусловием будем записывать в следующей форме:

Делать пока (условие)

```
{ действие;
  ...
  действие;
}
```

Цикл с постусловием — это цикл, в котором действия, составляющие тело цикла, исполняются, пока не станет истинным условие, записанное в окончании цикла. В алгоритмах цикл с постусловием будем записывать в следующей форме:

Повторять

```
{ действие;
  ...
  действие;
}
до (условие);
```

В качестве условия обычно используется какое-либо высказывание, проверка истинности которого является допустимым действием исполнителя. В цикле с предусловием эта проверка осуществляется всегда перед началом очередного исполнения тела цикла, а в цикле с постусловием — всегда после очередного исполнения тела цикла и никогда не совершается во время исполнения тела цикла (даже если в результате исполнения какого-либо действия высказывание становится ложным).

Две указанные формы цикла с условием полностью взаимозаменяемы, т. е. использование одного из них мо-

жет быть заменено другим. Однако при решении задач на алгоритмизацию нужно стараться выбрать ту форму, которая более естественна (например, приводит к выполнению исполнителем меньшего числа действий).

В записи алгоритма конструкцию цикл повторить n раз договоримся оформлять так:

Повторить n раз

```
{ действие;  
  ...  
  действие;  
}
```

Здесь n — числовая переменная целого типа либо арифметическое выражение, значение которого также имеет целый тип (см. п. 12.1). Некоторые языки программирования допускают в качестве ограничителя n числовую переменную и вещественного типа; в этом случае тело цикла исполняется $[n]$ раз.¹

Цикл со счётчиком — это цикл, в котором известно, сколько раз нужно исполнить тело цикла, причём значение счётчика влияет на исполнение действий в теле цикла. В алгоритмах цикл со счётчиком будем записывать в следующей форме:

Делать от $K := \dots$ до ... с шагом ...

```
{ действие;  
  ...  
  действие;  
}
```

Здесь K — числовая переменная (см. п. 12.1); после знака $:=$, слов **до** и **с шагом** могут стоять числовые константы или арифметические выражения. Слова **с шагом** могут отсутствовать; в этом случае после очередного исполнения тела цикла значение переменной K автоматически увеличивается на 1. Тело цикла исполняется до тех пор, пока значение переменной K не станет больше константы или значения арифметического выражения, стоящего после слова **до**.

Чтобы наглядно представлять формы организации действий, используют так называемые схемы.

Достаточно часто внутри одной алгоритмической конструкции может потребоваться использование одной или нескольких других алгоритмических конструкций. Такие конструкции называются вложенными друг в друга.

¹ Через $[n]$ обозначают наибольшее целое число, не превосходящее n ; это число называют целой частью действительного числа n .

Вопросы и задания

- 1 Составьте алгоритм размена суммы денег (k рублей) двух- и пятирублёвыми монетами.
- 2 Составьте алгоритм, который по запрашиваемой оценке в баллах сообщил бы её словесный эквивалент:
 - 1 балл — плохо;
 - 2 балла — неудовлетворительно;
 - 3 балла — удовлетворительно;
 - 4 балла — хорошо;
 - 5 баллов — отлично.
- 3 а) Составьте алгоритм, который по запрашиваемому номеру месяца сообщал бы количество дней в этом месяце (для невисокосного года).
 б) Составьте алгоритм, который по запрашиваемому номеру месяца и номеру года сообщал бы количество дней в этом месяце.
- 4 Робот перемещается по клетчатому полю, переходя из клетки в соседнюю клетку по соответствующей команде: вверх, вниз, вправо, влево. Между некоторыми клетками поля установлены стены, сквозь которые Робот пройти не может. Робот умеет также проверять истинность условия отсутствия стены у каждой стороны той клетки, где он находится. Если Робот попытается пройти сквозь стену, то он разрушится и исполнение программы прервётся.
 а) Робот исполняет следующий алгоритм, находясь на поле, изображённом на рисунке 11.1.

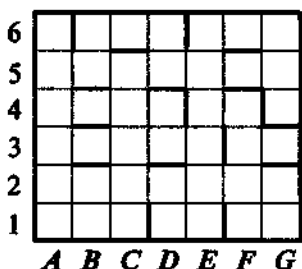


Рис. 11.1

Алгоритм

```

{ Делать пока (не снизу стена)
  { Вниз;
    Вправо;
  }
  Делать пока (не сверху стена)
  { Вверх;
    Вправо;
  }
}

```

Длиной пути Робота считается количество переходов из клетки в соседнюю клетку при успешном завершении исполнения алгоритма. Робот может начать исполнение алгоритма с лю-

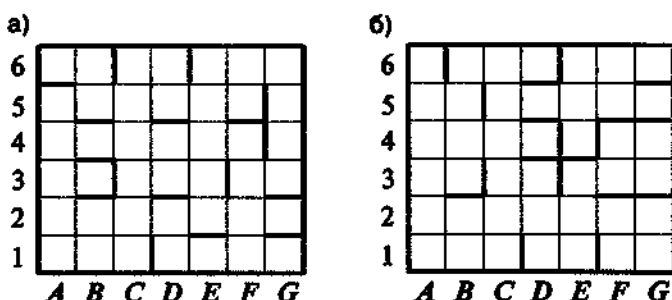


Рис. 11.2

бой клетки поля. Укажите, для какой начальной клетки путь робота окажется самым длинным.

б) Выполните то же задание, что и в пункте а, для полей, изображённых на рисунке 11.2.

в) Нарисуйте своё поле тех же размеров, но с иным расположением стен. Предложите соседу по парте выполнить то же задание, что и в пункте а, на придуманном вами поле. Проверьте его ответ.

- 5 а) На поле, изображённом на рисунке 11.3, Робот, описанный в задании 4, исполняет следующий алгоритм:

Алгоритм

```
{ Повторить  $k$  раз
  { Вверх;
  }
  Повторить  $m$  раз
  { Влево;
  }
  Повторить  $n$  раз
  { Вниз;
  }
  Повторить  $s$  раз
  { Вправо;
  }
}
```

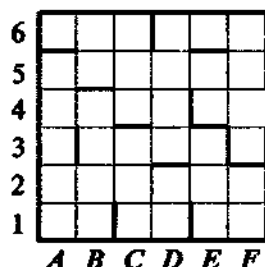


Рис. 11.3

Какое наибольшее значение может иметь сумма $k + m + n + s$, для которой Робот, начав исполнение предложенного алгоритма с подходящей клетки, успешно его закончит? Для этого значения суммы укажите клетку, с которой Робот должен начать исполнение алгоритма.

б) Выполните то же задание, что и в пункте а, для полей, изображённых на рисунке 11.4.

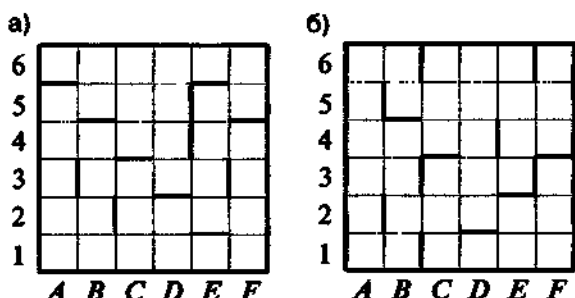


Рис. 11.4

- 6 На поле, изображённом на рисунке 11.3, Робот, описанный в задании 4, исполняет следующий алгоритм:

Алгоритм

```
{ Повторить  $k$  раз
  { Вверх;
    Повторить  $m$  раз
      { Влево;
        Повторить  $n$  раз
            { Вниз;
              Повторить  $s$  раз
                  { Вправо;
                    }
              }
            }
        }
      }
    }
  }
```

Какое наибольшее значение может иметь сумма $k + m + n + s$, для которой Робот, начав исполнение предложенного алгоритма с подходящей клетки, успешно его закончит? Для этого значения суммы укажите клетку, с которой Робот должен начать исполнение алгоритма.

- 7 а) На поле, изображённом на рисунке 11.5, Робот, описанный в задании 4, исполняет следующий алгоритм:

Алгоритм

```
{ Делать пока (не справа стена)
  { Вверх;
  }
  Делать пока (не сверху стена)
  { Влево;
  }
```

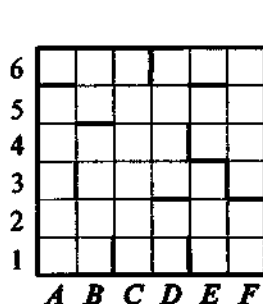


Рис. 11.5

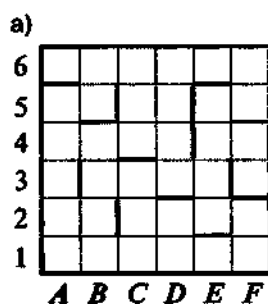
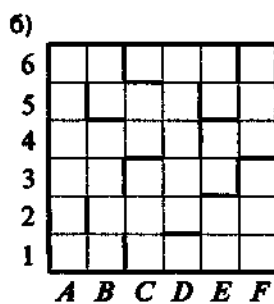


Рис. 11.6



Делать пока (не слева стена)

{ Вниз;

}

Делать пока (не внизу стена)

{ Вправо;

}

}

Перечислите те клетки лабиринта, для которых Робот, исполнив предложенный алгоритм, уцелеет и остановится в той же клетке, с которой он начал движение.

б) Выполните то же задание, что и в пункте а, для полей, изображённых на рисунке 11.6.

- 8 На поле, изображённом на рисунке 11.7, Робот, описанный в задании 4, исполняет следующий алгоритм:

Алгоритм

{ **Делать пока (не сверху стена)**

{ Вверх;

Если (не справа стена) то { Вправо; }

Если (не снизу стена) то { Вниз; }

Если (не слева стена) то { Влево; }

}

}

а) Для каждой клетки первого ряда укажите, в какой клетке окажется Робот после исполнения данного алгоритма.

б) Имеются ли на этом поле жизненно опасные клетки, т. е. такие клетки, начав с которых Робот разобьётся при исполнении данного алгоритма?

в) Перечислите те клетки данного поля, начав с которых Робот никогда не закончит исполнение данного алгоритма.

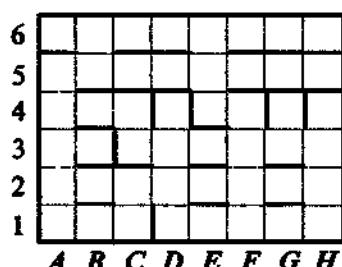


Рис. 11.7

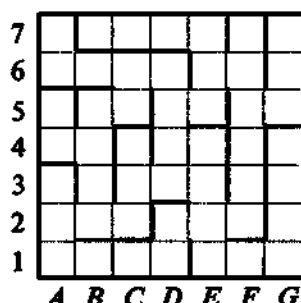


Рис. 11.8

г) Выполните задания, сформулированные в пунктах а и в, для поля, изображённого на рисунке 11.8.

- 9 На поле, изображённом на рисунке 11.7, Робот, описанный в задании 4, исполняет следующий алгоритм:

Алгоритм

```
{ Делать пока (не сверху стена)
  { Верх;
    Если (не слева стена) то { Влево; }
    Делать пока (не справа стена)
      { Вправо;
        Если (не внизу стена) то { Вниз; }
      }
    }
}
```

а) Для каждой клетки первого ряда укажите, в какой клетке окажется Робот после исполнения данного алгоритма.

б) Перечислите те клетки, начав с которых Робот никогда не закончит исполнение данного алгоритма.

в) Выполните задания, сформулированные в пунктах а и б, для поля, изображённого на рисунке 11.8.

- 10 Имеется полоска клетчатой бумаги шириной в 1 клетку и длиной в n клеток. Робот может закрасить клетку, на которой он находится, в красный, жёлтый или зелёный цвет (не важно, каким цветом была клетка закрашена до этого), а также переместиться на одну клетку вправо или влево. Робот умеет также проверять, какого цвета клетка, на которой он находится, и не закончилась ли полоска. Если Робот стоит у края полоски и получает команду переместиться за край, то он выходит из строя. Робот стоит на самой левой клетке и исполняет следующий алгоритм:

Алгоритм

```

{ Если (находится на зелёной клетке) то
  { Закрасить клетку в жёлтый цвет;
  }
  Делать пока (находится не на зелёной клетке и справа
               не край полосы)
  { Перейти вправо;
    Делать пока (находится на зелёной клетке и слева не
                 край полосы)
    { Закрасить клетку в жёлтый цвет;
      Перейти влево;
    }
    Если (находится на жёлтой клетке и справа не край
          полосы) то
      { Закрасить клетку в красный цвет;
        Перейти вправо;
      }
    Делать пока (находится на красной клетке и слева
                 не край полосы) то
      { Закрасить клетку в зелёный цвет;
        Перейти влево;
      }
    }
  }
  Перейти влево;
}

```

а) Пусть $n = 11$ и первоначально клетки полосы (по порядку слева направо) закрашены так: зелёная, жёлтая, красная, жёлтая, красная, зелёная, зелёная, жёлтая, красная, жёлтая, зелёная. Сколько красных, жёлтых и зелёных клеток будет в полоске после исполнения данного алгоритма?

б) Пусть первоначально вся полоска закрашена красным цветом. Сколько красных, жёлтых и зелёных клеток будет в полоске после исполнения данного алгоритма?

в) Пусть первоначально вся полоска закрашена зелёным цветом. Сколько красных, жёлтых и зелёных клеток будет в полоске после исполнения данного алгоритма?

г) Пусть первоначально вся полоска закрашена жёлтым цветом. Сколько красных, жёлтых и зелёных клеток будет в полоске после исполнения данного алгоритма?

д) Придумайте свою раскраску полосы. Предложите соседу по парте выяснить, сколько красных, жёлтых и зелёных клеток будет в полоске после исполнения данного алгоритма при предложенной вами исходной раскраске. Проверьте его ответ.

- 11* Автомат за один шаг преобразует данное ему натуральное число следующим образом: если число делится на 3, то он делит его на 3, в противном случае вычитает из него 1. Сколько существует натуральных чисел N , каждое из которых данный автомат за 10 последовательных шагов превратит в 1?

12.1. Переменные числового типа

Выражение, в котором фигурируют константы и переменные только числовых типов, называется арифметическим. При вычислении значения арифметического выражения действуют стандартные правила, устанавливающие старшинство операций: первой выполняется возведение в степень, затем умножение и деление и, наконец, сложение и вычитание.

В таблицах приведены обозначения операций, принятые в большинстве языков программирования, их взаимодействие с типами переменных, а также обозначения стандартных функций.

Обозначение основных операций

Операция	Выражение	Типы операндов	Тип результата
Сложение	$A + B$	Оба целые	Целый
		Хотя бы один вещественный	Вещественный
Умножение	$A * B$	Оба целые	Целый
		Хотя бы один вещественный	Вещественный
Вычитание	$A - B$	Оба целые	Целый
		Хотя бы один вещественный	Вещественный
Деление	A / B	Целый или вещественный	Вещественный
Целое деление	$A \text{ div } B$	Оба целые	Целый
Остаток при целом делении	$A \text{ mod } B$	Оба целые	Целый

Обозначения стандартных функций¹

Функция	Обраще- ние	Тип аргумента	Тип результата
Квадрат	$\text{sq}(x)$	Целый или вещественный	Вещественный
Корень квадратный	$\text{sqrt}(x)$	Целый или вещественный	Вещественный
Синус	$\text{sin}(x)$	Целый или вещественный (в радианах)	Вещественный
Косинус	$\text{cos}(x)$	Целый или вещественный (в радианах)	Вещественный
Арктангенс	$\text{arctan}(x)$	Целый или вещественный	Вещественный
Экспонента (показательная функция)	$\text{exp}(x)$	Целый или вещественный	Вещественный
Логарифм натуральный	$\text{ln}(x)$	Целый или вещественный	Вещественный
Модуль аргумента	$\text{abs}(x)$	Целый	Целый
		Вещественный	Вещественный
Округление	$\text{round}(x)$	Вещественный	Целый
Дробная часть	$\text{frac}(x)$	Целый или вещественный	Вещественный
Случайное число в интервале $[0; 1)$	random	—	Вещественный

¹ В таблице приведены обозначения стандартных функций, принятые в Турбо Паскале. Если учащиеся изучают другой язык программирования, то при выполнении заданий данного и последующих параграфов следует использовать обозначения того языка, который реально изучается. Ввиду возможных разночтений мы, в частности, в записях алгоритмов с переменными не придерживаемся строго правил какого-либо языка программирования, а используем общеупотребительную математическую запись.

Функция	Обраще- ние	Тип аргумента	Тип результата
Случайное це- лое число из интервала $[0; x)$	<code>random(x)</code>	Целый	Целый
Целая часть	<code>int(x)</code>	Целый или ве- щественный	Целый

Вопросы и задания

- В k -этажном доме m подъездов, по n квартир в каждом. Составьте алгоритм, позволяющий по номеру квартиры определить подъезд и этаж, где располагается данная квартира. Значения k , m и n запрашиваются у пользователя.
- Банкомат выдаёт запрашиваемую сумму банкнотами достоинством в 50, 100, 500 и 1000 рублей. Составьте алгоритм, позволяющий по запрашиваемой сумме S определить, сколько банкнот каждого достоинства надо выдать, если выдача происходит начиная с самых крупных купюр, а сумма S всегда кратна 50.
- Дан алгоритм, схема которого изображена на рисунке 12.1. Какой результат будет выведен после исполнения этого алгоритма, если в качестве начальных значений будут введены следующие наборы чисел:
 а) $x = 2, y = 0, z = 3$; б) $x = 1, y = 2, z = 0$;
 в) $x = 5, y = 5, z = 2$; г) $x = 8, y = 8, z = 8$?
- а) На координатной плоскости начерчена некоторая фигура. Дан алгоритм, позволяющий распознавать, принадлежит ли этой фигуре точка с координатами $(x; y)$:

Алгоритм Распознавание_1

вещ: x, y ;

{ Запросить x ;

Запросить y ;

Если $((x^2 + y^2 \leq 25) \text{ и } (x \geq -3) \text{ и } (x \leq 4))$ то

{ Сообщить "Точка с координатами (x, y) принадлежит фигуре";

}

}

Найдите площадь фигуры.

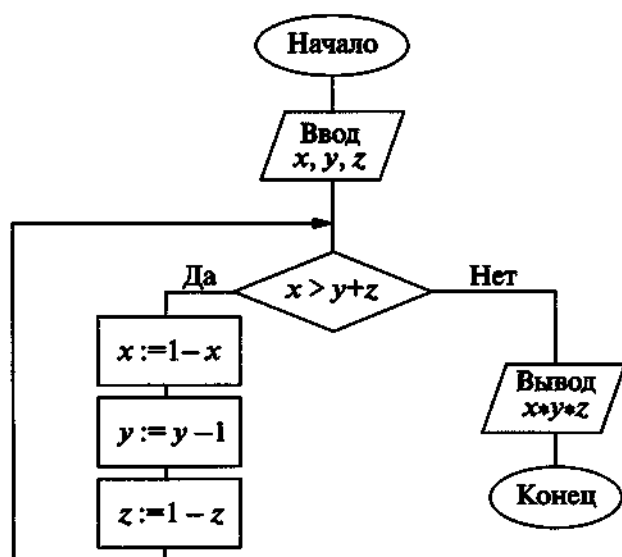


Рис. 12.1

б) Выполните такое же задание, как в пункте а, при следующем алгоритме распознавания:

Алгоритм Распознавание_2

вещ: x, y, z ;

{ **Запросить** x ;

Запросить y ;

$z := x^2 + y^2$;

Если $(z \leq 4x)$ **и** $(z \geq 2x)$ **то**

{ **Сообщить** "Точка с координатами (x , y) принадлежит фигуре";

}

}

- 5 П Составьте алгоритм решения уравнения $a|x| + b = 0$; значения a и b запрашиваются у пользователя.
- 6 а) Составьте алгоритм, при исполнении которого будет запрошено 4 числа и сообщено наибольшее и наименьшее из них.
б) Составьте алгоритм, при исполнении которого будет запрошено 4 числа и сообщено, образуют ли эти числа арифметическую прогрессию.
- 7 Составьте алгоритм, при исполнении которого будет запрошено 3 числа и сообщён периметр треугольника, для которого эти

3 числа являются длинами сторон. Если треугольника с такими сторонами не существует, об этом нужно сообщить.

- 8 Последовательность значений переменной X строится так:

$$X_1 = 1; X_2 = 3; \dots, X_n = 2X_{n-2} - 3X_{n-1}$$

для каждого $n > 2$.

а) Чему равны X_3 ; X_4 ; X_7 ?

б) Составьте алгоритмы:

- нахождения первого значения, большего 1000;
- нахождения суммы первых 30 значений переменной X ;
- нахождения первых десяти положительных значений переменной X ;
- нахождения наибольшего из первых 50 значений переменной X .

в)  Напишите на изучаемом вами языке программирования программы, реализующие составленные вами алгоритмы, и отладьте их.

- 9  Составьте алгоритм, при исполнении которого будет запрошено натуральное число и сообщена сумма кубов его цифр. Напишите на изучаемом вами языке программирования программу, реализующую этот алгоритм, и отладьте её.

- 10  Дано натуральное число n . Требуется переставить цифры этого числа так, чтобы получилось наибольшее возможное число. Составьте алгоритм, решающий эту задачу. Напишите на изучаемом вами языке программирования программу, реализующую этот алгоритм, и отладьте её.

- 11  Простым называется натуральное число, имеющее ровно два делителя.

а) Составьте алгоритм, при исполнении которого будет запрошено натуральное число n и сообщён его наименьший простой делитель. (Совет: обдумайте случай, когда исходное натуральное число равно 1.)

б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её. Проведите вычислительный эксперимент для $n = 637$; 4937; 51 983; 1 192 463.

- 12  Евклид доказал, что простых чисел бесконечно много.

а) Пусть все простые числа занумерованы в порядке возрастания. Составьте алгоритм, при исполнении которого будет сообщено простое число, имеющее номер n . Значение n запрашивается у пользователя.

б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её. Проведите вычислительный эксперимент для $n = 20$; 300; 7000; 10 000.

в)  Пусть $p(n)$ — простое число, имеющее номер n . Ис-

следуйте с помощью компьютера поведение функции $\frac{p(n)}{n \ln n}$

для n от 3000 до 100 000.

Математики доказали, что значение этой функции с ростом n всё меньше отличается от 1. Наблюдается ли это явление в вашем компьютерном эксперименте?

- 13 **П** а) До сих пор неизвестно, конечно ли количество простых чисел, которые лишь на единицу отличаются от квадрата натурального числа. Таких чисел, меньших 100, всего 4. Вот они: 3, 5, 17, 37. Составьте алгоритм, при исполнении которого будет сообщено, сколько имеется таких простых чисел, не превосходящих число n . Значение n запрашивается у пользователя. (Совет: обдумайте, много ли существует простых чисел, которые на единицу меньше квадрата целого числа.)
 б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её. Проведите вычислительный эксперимент для $n = 100; 500; 7000; 10\,000$.
- 14 **П** а) Вещественное число a образовано следующим образом. Сначала написана цифра 0, после которой поставлена запятая, а затем подряд без пробелов записаны по порядку все натуральные числа. Вот как выглядит запись этого числа, в которой приведены первые 20 цифр дробной части: $a = 0,12345678910111213141\dots$. Составьте алгоритм, позволяющий определить, какая цифра записана в этом числе на n -м месте после запятой. Значение n запрашивается у пользователя.
 б)* Вещественное число b получается аналогичным образом, но для его записи используются только чётные натуральные числа. Вот как выглядит запись числа b , в которой приведены первые 20 цифр его дробной части: $b = 0,24681012141618202224\dots$. Составьте алгоритм, позволяющий определить, какая цифра записана в этом числе на n -м месте после запятой. Значение n запрашивается у пользователя.
 в)* Вещественное число c получается аналогичным образом, но для его записи используются последовательные квадраты натуральных чисел. Вот как выглядит запись числа c , в которой приведены только первые 20 цифр его дробной части: $c = 0,14916253649648110012\dots$. Составьте алгоритм, позволяющий определить, какая цифра записана в этом числе на n -м месте после запятой. Значение n запрашивается у пользователя.
 г)  Напишите на изучаемом вами языке программирования программы, реализующие составленные вами алгоритмы, и отладьте их. Проведите с ними вычислительный эксперимент для $n = 20; 300; 7000; 10\,000$.

- 15) П а) Перечитайте задачу 5 из § 10. Составьте алгоритм, позволяющий по номеру строки определить, какая цифра стоит на n -м месте цепочки, записанной в этой строке.
 б) Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм. Проведите вычислительный эксперимент для строк с номерами 7, 13, 100 и $n = 100, 1000$ и $10\,000$.
- 16)* П Пусть a и b — два заданных целых числа, а x и y — пара целых чисел, удовлетворяющих уравнению $ax + by = ab$. Каждая такая пара $(x; y)$ задаёт на координатной плоскости точку. Составьте алгоритм, позволяющий найти x и y , для которых соответствующая этой паре точка была бы ближайшей к началу координат. Если таких точек несколько, то нужно сообщить координаты всех таких точек.
 б) ■ Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её. Используйте вашу программу, найдите x и y для $a = 6, b = 10; a = 0, b = 5; a = 23, b = -37; a = -210, b = -126$.
- 17) П Составьте алгоритм нахождения максимального значения выражения $3x^2 - 4y^2 + z^2$, если натуральные числа x, y и z удовлетворяют неравенствам $4x - 5y + z < 42, x(3y + z) < 100$.
 б) ■ Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её. Найдите требуемое максимальное значение.

12.2. Символьные и строковые переменные

Для хранения информации, представляющей собой один символ или последовательность символов, применяют переменные, которые соответственно имеют тип «символ» или «строка». Сокращённо эти типы мы будем записывать как сим и стр. Последовательность символов нередко называют словом (в абстрактном смысле).

Для переменных этого типа обычно рассматриваются операции соединения (по-другому — конкатенации) и вырезки, т. е. выделения части слова.

Операция соединения состоит в том, что к первому слову приписывается второе. Команду соединения будем записывать знаком $+$, например:

$$A + B,$$

здесь через A и B обозначены переменные строкового типа.

Команду вырезки будем записывать так:

Часть ("ИНФОРМАТИКА", 3, 5),

где на первом месте указывается слово, из которого вырезается часть, на втором — номер места, с которого начинается вырезаемая часть, наконец, на третьем месте указыва-

ется число символов в вырезаемой части. Скажем, в результате выполнения написанной команды получится слово **ФОРМА**. Вместо самого слова в качестве первого аргумента может указываться символьная переменная; операция тогда будет выполняться над значением этой переменной.

Команду нахождения длины слова, т. е. числа символов, из которых оно составлено, будем записывать так:

Длина ("ИНФОРМАТИКА").

Для этой команды также вместо слова может быть указана символьная переменная, к значению которой применяется данная операция. Легко понять, что в приведённом примере результатом является число 11. Для обеих команд кавычки являются служебным знаком, показывающим, что мы имеем дело со словом, а не с именем какой-либо переменной. Поэтому кавычки не входят в число символов, участвующих в данном слове. В частности, слово может вообще не содержать символов. Такое слово называется **пустым**, оно обозначается как "", а его длина равна 0. Пустое слово надо отличать от слова " ", которое состоит из одного символа — пробела.

Всевозможные слова над заданным алфавитом считаются упорядоченными в алфавитном порядке. Это означает, что значения строковых переменных можно сравнивать не только на совпадение и несовпадение (равенство и соответственно неравенство слов), но и относительно заданного упорядочения, индуцированного порядком символов в алфавите.

Вопросы и задания

- 1 Какие операции применяются для действий над данными символьного и строкового типов?
- 2 При исполнении программ, использующих символьные переменные, символы считаются упорядоченными в соответствии с тем, как они располагаются в соответствующей кодовой таблице (см. Приложение, Таблицы П.1—П.4).
 - а) Расположите в порядке возрастания слова «?:?!», «:!?», «?:!?', «:!?!».
 - б) Расположите в порядке возрастания слова «банк», «Саратов», «СНГ», «ЮАР», «ярмо», «Ярославль», если используется кодовая таблица CP-1251.
 - в) Выполните то же задание, что и в пункте б, если используется кодовая таблица КОИ-8.
- 3 а) Найдите слово русского языка, которое больше, чем слово ПАРА, и меньше, чем слово ПАРАБОЛА.

б) Два слова называются соседними, если не существует слова, большего одного из них и меньшего другого. Являются ли соседними в русском языке слова ПАПА и ПАПАША? Какое слово русского языка является соседним к слову ТРИБУНА?

- 4) Рассмотрите следующий алгоритм:

Алгоритм Преобразование

цел: n ; **стр:** A ;

```
{ Запросить  $n$ ;
   $A := \text{"ABCDEF"}$ ;
  Повторить  $n$  раз
  { Если ( $\text{Часть}(A, 1, 1) = \text{"A"}$  или  $\text{Часть}(A, 1, 1) = \text{"E"}$ ) то
    {  $A := \text{Часть}(A, 2, 4) + \text{Часть}(A, 1, 1) + \text{Часть}(A, 6, 1)$ ;
    }
  }
  иначе
  {  $A := \text{Часть}(A, 2, 5) + \text{Часть}(A, 1, 1)$ ;
  }
}
Сообщить  $A$ ;
```

Укажите наименьшее значение n , чтобы после исполнения этого алгоритма было сообщено: а) FBCEAD; б) BCEADF; в) FBEACD.

- 5)* Рассмотрите следующий алгоритм:

Алгоритм Преобразование+

цел: n ; **стр:** A ;

```
{ Запросить  $n$ ;
   $A := \text{"ABCDEF"}$ ;
  Повторить  $n$  раз
  { Если ( $\text{Часть}(A, 1, 1) = \text{"A"}$  или  $\text{Часть}(A, 1, 1) = \text{"E"}$ ) то
    {  $A := \text{Часть}(A, 2, 5) + \text{Часть}(A, 1, 1)$ ;
    }
  }
  иначе
  {  $A := \text{Часть}(A, 6, 1) + \text{Часть}(A, 2, 4) + \text{Часть}(A, 1, 1)$ ;
  }
}
Сообщить  $A$ ;
```

Укажите, при каких значениях n будет сообщено: а) ABCDEF; б) FBCDAE; в) EBDCAF.

- 6 Рассмотрите следующий алгоритм:

Алгоритм Укорачивание

цел: n ; **стр:** A ;

{ **Запросить** A ;

$n := \text{Длина}(A)$;

Делать пока ($n > 1$)

{ $n := n - 1$;

$A := \text{Часть}(A, 2, n)$;

Сообщить A ;

}

}

Сколько слов русского языка будет напечатано в результате исполнения алгоритма, если на запрос о значении переменной A будет введено слово: а) БОРОВ; б) ПОДВОДА; в) ПОВОРОТ?

- 7 а) Составьте алгоритм, при исполнении которого будет запрошено натуральное число от 1 до 150 — возраст пользователя — и напечатана фраза «Вам ... лет» (вместо многоточия печатается число, указанное пользователем). Фраза должна быть грамотной, т. е. для числа 22 печатается «Вам 22 года», для числа 51 печатается «Вам 51 год» и т. д.

б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.

- 8 а) Дана строка, состоящая из нескольких слов русского языка, разделённых пробелами. Составьте алгоритм, позволяющий определить, сколько слов в этой строке.

б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.

- 9  а) Дана строка, состоящая из нескольких слов русского языка, разделённых пробелами. Требуется сообщить слово, содержащее наибольшее количество букв (если таких слов несколько, сообщить все). Составьте соответствующий алгоритм.

б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.

- 10  а) Составьте алгоритмы для определения:

- является данная буква гласной, согласной или одной из букв $\bar{y}$, $\bar{y}$;
- обозначает ли данная согласная буква звонкий или глухой звук. (Совет: используйте символьные переменные $A = \text{"аеиоуыёюя"}$, $Z = \text{"бвгджзлмнр"}$, $G = \text{"кпстфхцщшц"}$, $W = \text{"ъь"}$.)

- б)  Напишите на изучаемом вами языке программирования программы, реализующие составленные вами алгоритмы, и отладьте их.
- 11 **а)** Составьте и отладьте программу для проверки правописания приставок *воз-* и *вос-* в словах русского языка. (Совет: используйте решение предыдущей задачи.)
 б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.
- 12 **а)** Дана строка. Составьте алгоритм, исполнив который компьютер сообщит те символы, которые встречаются в этой строке более одного раза.
 б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.
- 13 **а)** Пусть имеется два слова, составленных из символов некоторого алфавита. Из них составляются новые слова так, что символы одного слова вставляются между символами другого слова с сохранением порядка следования символов в каждом из исходных слов. Например, из слов ПИЛ и АРКА можно получить слово ПАРИЛКА, а из слов ВОР и ПОТ слово ПОВТОР. Составьте алгоритм, который по трём запрашиваемым словам определяет, может ли третье слово быть получено из первых двух с помощью указанной процедуры.
 б)  Напишите на изучаемом вами языке программирования программы, реализующие составленный вами алгоритм, и отладьте их.
- 14* **а)** Пусть имеется строка, составленная из заглавных букв латинского алфавита. Составьте алгоритм, который удаляет наименьшее число букв, после чего остаётся строка, в которой все буквы различны и идут в алфавитном порядке. Строка запрашивается у пользователя. Например, для строки UGKFLFVOABCSTDWEZG должна получиться строка GKLOSTWZ.
 б)  Напишите на изучаемом вами языке программирования программы, реализующие составленный вами алгоритм, и отладьте их.
- 15* **а)** Перечитайте задачу 11 из § 10. Будем обозначать буквой **У** команду, по которой исполнитель умножает число на 3, а буквой **В** команду, по которой исполнитель вычитает из числа 1. Последовательность **УВ** означает, что сначала число умножается на 3, а затем из результата вычитается 1; запись **ВУ** означает, что сначала из числа вычитается 1, а затем результат умножается на 3. Составьте алгоритм, который для каждого натурального числа n будет составлять последовательность команд, исполняя которую можно получить число n из числа 1 за наименьшее количество действий.

- б) ■ Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм. Проведите компьютерный эксперимент для $n = 100$, 2000 и $n = 100\,000$.

12.3. Переменные логического типа

Логические, или, по-другому, булевы, переменные — это переменные, принимающие только два значения. Одно из этих значений обычно обозначают Истина, другое — Ложь. Применяются также обозначения True и False, 1 и 0.

Над логическими переменными выполняются логические операции *и*, *или*, *не*, результат применения которых задаётся следующими равенствами:

Истина <i>и</i> Истина = Истина	Истина <i>или</i> Истина = Истина
Истина <i>и</i> Ложь = Ложь	Истина <i>или</i> Ложь = Истина
Ложь <i>и</i> Истина = Ложь	Ложь <i>или</i> Истина = Истина
Ложь <i>и</i> Ложь = Ложь	Ложь <i>или</i> Ложь = Ложь
<i>не</i> Истина = Ложь	
<i>не</i> Ложь = Истина	

В языках программирования эти операции часто обозначаются соответственно как AND, OR и NOT.

Логические значения вырабатываются при сравнении значений переменных. Например, для числовых переменных a и b при сравнении $a > b$, для числовых переменных c и d при сравнении $c \neq d$, для логической переменной f при сравнении $f = \text{Истина}$. Логическое значение вырабатывается и в комбинациях операций сравнения, скажем, в такой: $(x^2 + 7x - 15 < 0) \neq (x > 1)$. Указанная комбинация имеет значение Истина, например, при $x = 1,5$ и при $x = -9$, а при $x = 0$ имеет значение Ложь.

Вопросы и задания

- Пусть x — переменная целого типа. Перечислите её значения, при которых истинно следующее логическое выражение:
 - $(x^2 + 4x < 15)$ или $(x^2 - 10x < 20)$;
 - $(x^2 - 4x < 23)$ и $(x^3 - 6x > 4)$;
 - Часть ("потолок", x , 1) = Часть ("потолок", $8 - x$, 1);
 - $((x^2 + 7x - 17 < 0) \neq (x > 0))$ и $(x < 4)$.
- Дан алгоритм, вычисляющий члены последовательности, заданной правилом $X_n = X_{n-2} - 2X_{n-1}$ для каждого $n > 2$; значения X_1 и X_2 запрашиваются у пользователя.

Алгоритм 1**лог:** q ; **цел:** K, L, M, N ;{ **Запросить** L ; (*запрашивается значение X_1 *)**Запросить** M ; (*запрашивается значение X_2 *)**Если** $(L < 0$ **или** $M < 0)$ **то** { $q :=$ Истина; }**иначе** | $q :=$ Ложь; }**Делать от** $K := 1$ **до** 6{ $N := M - 2 * L$; $M := L$; $L := N$;**Если** $(N < 0)$ **то** { $q :=$ Истина; }

}

Если $(q = \text{Истина})$ **то**{ **Сообщить** "Последовательность содержит отрицательный член"}**иначе**{ **Сообщить** "Все члены последовательности положительны" }

}

а) Определите, каким будет результат исполнения алгоритма, если $X_1 = 3$; $X_2 = 1$.б) Определите, каким будет результат исполнения алгоритма, если $X_1 = 75$; $X_2 = 31$.в)* **П** Существуют ли такие X_1 и X_2 , для которых все шесть вычисляемых членов последовательности положительны?г) **П** При любых ли значениях X_1 и X_2 алгоритм даёт верный ответ? Если ваш ответ «Да», то его надо обосновать; если ответ «Нет», то нужно привести пример значений X_1 и X_2 , для которых алгоритм даёт ошибочный ответ; укажите также, как в этом случае исправить алгоритм, чтобы получить верный ответ при любых целочисленных начальных данных.

- 3 Дан алгоритм, вычисляющий члены последовательности, заданной правилом $X_n = X_{n-2} + X_{n-1} - n$ для каждого $n > 2$; значения X_1 и X_2 запрашиваются у пользователя.

Алгоритм 1**лог:** q ; **цел:** K, L, M, N ;{ **Сообщить** "Введите два целых положительных числа"**Запросить** L ; (*запрашивается значение X_1 *)**Запросить** M ; (*запрашивается значение X_2 *) $q :=$ Истина;**Делать от** $K := 3$ **до** 10{ $N := M + L - K$;**Если** $(N + L \Leftarrow M$ **или** $N + M \Leftarrow L)$ **то** { $q :=$ Ложь }

```

    M := L;
    L := N;
}
Если (q = Истина) то
{ Сообщить "Из отрезков, длины которых равны
любым трём подряд идущим членам последователь-
ности, можно построить треугольник"}
иначе
{ Сообщить "Есть три члена последовательности,
которые не могут быть длинами сторон треугольника"}
}
Определите, каким будет результат исполнения алгоритма,
если  $X_1 = 3$ ;  $X_2 = 5$ . А если  $X_1 = 2$ ;  $X_2 = 5$ ?

```

- ④ **Палиндромом** называется слово, которое читается справа налево так же, как и слева направо.
- а) Составьте алгоритм, который по введённой строке символов определит, можно ли в этой строке переставить символы так, чтобы получившаяся строка была палиндромом.
- б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.



Вспомогательные алгоритмы и подпрограммы

13.1. Вспомогательный алгоритм-процедура

Разновидность вспомогательных алгоритмов, называемых в языках программирования **процедурами**, предусматривает возможность указания прямо в самом алгоритме переменных, которым будут присваиваться результаты исполнения вспомогательного алгоритма. Вот как выглядит описание вспомогательного алгоритма данного типа:

```

Алгоритм Имя (арг тип: переменная, ...; тип: переменная, ...; рез тип: переменная, ...)
тип: переменная, ....;
{ действие;
...;
}

```

Вместо простых переменных здесь могут употребляться иные структуры данных (массивы, стеки, графы и т. п.; о них см. в § 14 и 15).

Слово **Алгоритм**, его имя и указанные в скобках имена аргументов и результатов (вместе с их описанием) образуют заголовок вспомогательного алгоритма. Фигурирующие в заголовке алгоритма переменные называются **формальными параметрами**.

Обращение к вспомогательному алгоритму мы будем указывать оператором **Вызвать**. Записывать обращение к вспомогательному алгоритму договоримся так:

Вызвать Имя (выражение, выражение, ...; переменная, переменная, ...)

Выражения, стоящие на местах формальных аргументов, и переменные, указанные на местах результатов, называются **фактическими параметрами**. Типы выражений и переменных, являющихся фактическими параметрами, должны совпадать с типами формальных переменных, на местах которых они стоят.

Вспомогательный алгоритм для своей работы может использовать собственные (внутренние) структуры данных. Такие структуры данных называются **локальными**, описание их типов задаётся отдельной строкой между заголовком и телом алгоритма. Их значения не идентичны значениям одноимённых структур данных в основном алгоритме. Во вспомогательном алгоритме локальные переменные могут отсутствовать. В этом случае строка с описанием их типов тоже отсутствует.

Вопросы и задания

- 1) а) Выполнив задание 8, б из § 10, вы составили алгоритм нахождения центра тяжести любого треугольника, вершины которого располагаются в вершинах клеток листа клетчатой бумаги. Используя этот алгоритм как вспомогательный, составьте алгоритм нахождения центра тяжести с помощью карандаша и линейки, пригодный для любого четырёхугольника с вершинами в вершинах клеток.
б) Нарисуйте на клетчатой бумаге какой-либо четырёхугольник, вершины которого располагаются в вершинах клеток. Предложите соседу по парте найти центр тяжести четырёхугольника с помощью составленного им алгоритма. Проверьте правильность исполнения алгоритма и правильность результата.
- 2) а)* Имеются два треугольника одинаковой площади. Составьте алгоритм, после исполнения которого один из треугольников будет разрезан на несколько таких частей, из которых можно составить другой треугольник.

б) Предложите соседу по парте нарисовать два равновеликих, но не равных треугольника и с помощью составленного вами алгоритма преобразовать один треугольник в другой. Проверьте, правильно ли был исполнен алгоритм и действительно ли из частей, на которые был разрезан первый треугольник, можно сложить второй.

в) Имеются треугольник и равновеликий ему многоугольник. Используя алгоритм, составленный при выполнении пункта б, как вспомогательный, составьте алгоритм, позволяющий разрезать треугольник на несколько частей, из которых можно составить данный многоугольник.

г) Имеются два равновеликих многоугольника. Используя алгоритм, составленный при выполнении пункта в, как вспомогательный, составьте алгоритм, позволяющий разрезать один многоугольник на несколько частей, из которых можно составить другой многоугольник.

13.2. Вспомогательный алгоритм-функция

Алгоритм-функция — это вспомогательный алгоритм, результат которого присваивается переменной не внутри этого алгоритма, а в основном алгоритме. Запись этой конструкции в нашем задачнике оформлена следующим образом.

Алгоритм Имя (тип: переменная, переменная,...): тип

тип: переменная,;

{ действие;

...;

действие;

}

В скобках после имени перечислены формальные аргументы алгоритма-функции с указанием их типов. Вместо переменных здесь могут употребляться иные структуры данных. В конце заголовка указывается тип результата. Имя алгоритма функции одновременно служит именем локальной переменной, тип которой совпадает с типом результата.

Вызов алгоритма-функции оформляется командой присваивания:

$a := \text{Имя (переменная, переменная, ...)}$

Здесь переменные — это фактические параметры. Вместо переменных могут стоять выражения, допустимые для заданного типа.

Вопросы и задания

- 1) Рассмотрите алгоритм A , в котором использован алгоритм-функция B :

Алгоритм A

цел: x, y, s ;

{ Запросить x ;

Запросить y ;

$s := 0$;

Делать пока $(x + y \neq 0)$

 { $s := B(x) * B(y) - s$;

$x := B(x)$;

$y := B(y)$;

 }

Сообщить s ;

}

Алгоритм B (цел: x): цел

цел: k ;

{ $B := 0$;

Делать от $k = 2$ до $x - 2$

 { Если $x \bmod k = 0$ то

 { $B := k$; }

 }

}

Какое число будет сообщено, если на запросы при исполнении алгоритма A будут введены: а) $x = 40, y = 36$; б) $x = -63, y = 20$; в) $x = 91, y = 77$; г) $x = -12, y = 12$?

- 2) Ниже приведены тексты алгоритма C и алгоритмов-функций A и B :

Алгоритм C

цел: k, m, n ;

{ Запросить n ;

$m := B(n)$;

Делать от $k = n + 1$ до $2 * n$

 { Если $(B(k) > m)$ то { $m := B(k)$; }

 }

Сообщить m ;

}

Алгоритм B (цел: x): цел

цел: k ;

{ $B := 0$;

```

    Делать от  $k = 1$  до  $x$ 
    { Если  $(A(k) = k)$  то {  $B := B + 1;$  }
    }

```

Алгоритм A (цел: x): цел

```

цел:  $k;$ 
{  $A := 1;$ 
  Делать от  $k = 1$  до  $x$ 
  {  $A := (A * k) \bmod (x + 1);$ 
  }
}

```

Какое число будет сообщено в результате исполнения алгоритма C , если: а) $n = 8$; б) $n = 12$; в) $n = 100$?

- 3 а) Составьте алгоритм-функцию $INV(n)$ целого типа от целочисленного аргумента n , значением которого является число, записанное теми же цифрами, что и число n , но в обратном порядке (если на первых местах оказались нули, то их писать не следует).
 б) Рассмотрите следующий алгоритм:

Алгоритм AP

```

цел:  $x, y, s;$ 
{ Сообщить "Введите натуральное число  $x$ ";
  Запросить  $x$ ;
  Делать пока  $(x > INV(x))$ 
  {  $x := x + INV(x);$ 
  }
  Сообщить  $x$ ;
}

```

Исполните этот алгоритм для $x = 57$ и $x = 86$.

в)  Реализуйте составленный вами алгоритм-функцию и алгоритм из пункта б на изучаемом вами языке программирования и отладьте получившиеся программы.

г)*  Проведите вычислительный эксперимент для нескольких трёхзначных, четырёхзначных и пятизначных чисел. Всегда ли программа успешно завершала работу? Если нет, то укажите причину и объясните, почему произошёл сбой в работе программы. Попытайтесь устранить воздействие этой причины.

- 4 а) Составьте алгоритм-функцию, позволяющий найти сумму четвёртых степеней цифр заданного натурального числа.
 б) Используя алгоритм-функцию, составленный вами при выполнении пункта а, составьте алгоритм, позволяющий найти все числа, равные сумме четвёртых степеней своих цифр.

- 5 а) Выполнив задание 12а из пункта 12.1, вы составили алгоритм нахождения n -го простого числа. Используя этот алгоритм как основу, составьте алгоритм-функцию, аргументом которого служит натуральное число n , а результатом — простое число с номером n .
- б) Вещественное число a образовано следующим образом. Сначала написана цифра 0, после которой поставлена запятая, а затем подряд без пробелов записаны простые числа в порядке возрастания. Вот как выглядит запись числа a , в которой приведены первые 20 цифр его дробной части: $a = 0,23571113171923293137\dots$. Используя алгоритм-функцию, составленный вами в пункте а, составьте алгоритм, позволяющий определить, какая цифра записана в этом числе на n -м месте после запятой. Значение n запрашивается у пользователя.
- в)  Реализуйте составленные вами алгоритмы на изучаемом вами языке программирования и отладьте получившиеся программы. Проведите вычислительный эксперимент для $n = 20; 300; 7000; 10\,000$.

13.3. Рекурсия

В общем случае рекурсией называется ситуация, когда какой-то алгоритм непосредственно или через другие алгоритмы вызывает себя в качестве вспомогательного. Сам алгоритм при этом называется рекурсивным.

Вопросы и задания

- 1 Рассмотрите следующий алгоритм, использующий вспомогательный алгоритм-функцию $A(x)$:

Алгоритм

цел: x ;

{ Запросить x ;

$x := A(x)$;

Сообщить x ;

}

Алгоритм A (цел: x): цел

{ Если $(x = 1)$ то { $A := 1$; }

иначе

{ Если $(x \bmod 2 = 1)$ то { $A := 3 * A(x \div 2) - 1$; }

иначе { $A := 2 * A(x \div 2) + 1$; }

}

Каков будет результат работы алгоритма для чисел 1; 10; 15, 100?

- 2 Ниже приведены тексты алгоритма C и алгоритмов-функций A и B .

Алгоритм C

цел: k, m, n, s ;

{ Запросить m ;

Запросить n ;

$s := 0$;

Делать от $k = 1$ до $A(m, n)$

{ $s := s + B(m, k) + B(k, n)$;

}

Сообщить s ;

}

Алгоритм B (цел: x, y): цел

{ Если $(x < 5)$ то { $B := x$; }

иначе { $B := x - A(x - 3, y + 1)$; }

}

Алгоритм A (цел: x, y): цел

{ Если $(y < x)$ то { $A := x - y$; }

иначе { $A := y + B(x, y - x)$; }

}

Какое число будет сообщено, если на запросы при исполнении алгоритма C будут введены:

а) $m = 1, n = 2$; б) $m = 10, n = 3$; в) $m = 10, n = 10$;

г) $m = 10, n = 0$; д) $m = 0, n = 10$; е) $m = -1, n = -2$?

- 3* Рассмотрите ещё раз алгоритмы, представленные в задании 2. Запишите результат исполнения алгоритма C , если при исполнении этого алгоритма на его запросы будет введено значением n , равное значению m .

- 4 Рассмотрите следующий алгоритм, использующий алгоритм-функцию $P(x)$:

Алгоритм

цел: x ;

{ Запросить x ;

$x := P(x)$;

Сообщить x ;

}

Алгоритм P (цел: x): цел

цел: k ;

{ Если $(x = 1)$ то { $P := 2$; }

иначе

{ $P := P(x - 1) + 1$;

$k := 1$;

```

Делать пока ( $k < x$ )
{ Если ( $P \bmod P(k) = 0$ ) то
  {  $P := P + 1$ ;
     $k := 1$ ; }
  иначе
    {  $k := k + 1$ ; }
}

```

а) Какой будет результат работы алгоритма для чисел 1; 4; 9?
 б)* Определите, для решения какой задачи предназначен данный алгоритм. Ответ обоснуйте.

- 5 Составьте рекурсивный алгоритм, подсчитывающий для запрашиваемого натурального числа n сумму кубов его цифр. Сравните его с нерекурсивным алгоритмом, который вы составили, выполнив задание 9 из пункта 14.1.
- 6 а) Составьте алгоритм для подсчёта количества способов, которыми можно выдать сумму в n рублей монетами достоинством в 1 рубль и 2 рубля.
 б) Составьте алгоритм для подсчёта количества способов, которыми можно выдать сумму в n рублей монетами достоинством в 2 рубля и 5 рублей.
 в)* Составьте алгоритм для подсчёта количества способов, которыми можно выдать сумму в n рублей монетами достоинством в 1 рубль, 2 рубля и 5 рублей.
- 7* На столе стоят два столбика из чёрных и белых кубиков: в первом из них сверху n чёрных кубиков, потом n белых, а во втором столько же чёрных и белых кубиков, но они чередуются — чёрный, белый, чёрный, белый и т. д. Коля складывает из всех этих кубиков один столбик, беря по одному кубику сверху из любого столбика наугад. Сколько различных вариантов столбика, содержащего 4 n кубиков, может получиться у Коли? Составьте алгоритм, позволяющий ответить на этот вопрос.

13.4. Нисходящее и восходящее программирование

Основной метод решения сложной задачи состоит в её разбиении на более простые задачи и/или выделение в ней таких задач, решение которых уже известно. В алгоритмизации подход, связанный с разбиением на более простые задачи, называют методом нисходящего проектирования (проектированием сверху вниз); подход, при котором требующийся алгоритм собирается из готовых вспомогательных алгоритмов, называют методом восходящего проекти-

рования (проектированием снизу вверх). Готовые вспомогательные алгоритмы, оформленные так, чтобы ими можно было легко пользоваться при конструировании других алгоритмов (нередко они оформлены как алгоритмы-функции), обычно называются стандартными и размещаются в библиотеке стандартных алгоритмов. Соответствующие подпрограммы тоже называются стандартными и размещаются в библиотеке стандартных подпрограмм.

Вопросы и задания

- 1 а) Разработайте проект решения следующей задачи с помощью компьютера, который может обрабатывать не более чем 10-разрядные целые числа.
Напечатать число $n!$ в десятичной записи для заданного $n > 30$. (Через $n!$ обозначают произведение всех натуральных чисел от 1 до n включительно.)
Реализуйте этот проект, написав соответствующие основной и вспомогательные алгоритмы. (Совет: обдумайте, какую структуру данных здесь целесообразно использовать для хранения цифр этого числа.)
- б) ■ На изучаемом вами языке программирования напишите программу и комплекс подпрограмм, соответствующих алгоритмам вашего проекта. Протестируйте работу вашего программного комплекса. Получите результаты для n в диапазоне от 31 до 40.
- 2* а) Пусть $p_1, p_2, \dots, p_n$ — это n первых простых чисел, занумерованных в порядке возрастания. Тогда число $p_1 p_2 \dots p_n + 1$ заведомо не делится ни на одно из чисел $p_1, p_2, \dots, p_n$. При $n = 1, n = 2$ и $n = 3$ оно равно 3, 7 и 31 соответственно, т. е. является простым числом. Всегда ли будет так, и если нет, то как часто среди чисел такой природы встречаются составные числа? Реализуйте проект, позволяющий для как можно большего n выяснить, какую долю составляют составные числа среди всех чисел такого вида. Напишите для этого соответствующие основной и вспомогательные алгоритмы. (Совет: воспользуйтесь решениями задач 11 и 12 из пункта 12.1.)
- б) ■ На изучаемом вами языке программирования напишите программу и комплекс подпрограмм, соответствующих алгоритмам вашего проекта. Проведите вычислительный эксперимент. Для какого наибольшего n вам удалось обследовать числа указанного вида? (Результат можно признать хорошим, если вам удалось обследовать не менее 25 таких чисел.)
- 3* а) Пусть $p_1, p_2, \dots, p_n$ — это n первых простых чисел, занумерованных в порядке возрастания. Тогда число $(p_1 p_2 \dots p_n)^2 + 1$ тоже (как и числа, рассматривавшиеся в задании 2) не делит-

ся ни на одно из чисел $p_1, p_2, \dots, p_n$. При $n = 1$, $n = 2$ и $n = 3$ оно равно 2, 37 и 901 соответственно. Первые два числа простые, а третье — составное: $901 = 17 \cdot 53$. Реализуйте проект, позволяющий для как можно большего n выяснить, какую долю составляют составные числа среди всех чисел такого вида, написав для этого соответствующие основной и вспомогательные алгоритмы.

б) На изучаемом вами языке программирования напишите программу и комплекс подпрограмм, соответствующих алгоритмам вашего проекта. Проведите вычислительный эксперимент. Для какого наибольшего n вам удалось обследовать числа указанного вида? (Результат можно признать хорошим, если вам удалось обследовать не менее 15 таких чисел.)

- 4* На П. Ферма открыл следующую теорему: если p — простое число, то для любого натурального числа a число $a^p - a$ делится на p нацело. Числа, для которых неверно обратное утверждение, называются числами Кармайкла. Среди первых ста миллионов имеется 255 чисел Кармайкла; наименьшее из них — 561. Разработайте проект, позволяющий находить числа Кармайкла. Проверьте, используя написанные вами программы, что число 561 действительно является наименьшим числом Кармайкла, и попытайтесь найти четыре числа Кармайкла, следующие за числом 561. (Совет: учтите, что число Кармайкла не является простым; обдумайте, может ли число Кармайкла быть чётным и как ограничить количество чисел a , для которых нужно проводить проверку делимости $a^p - a$ на p , ведь перепробовать все натуральные числа невозможно!)

- 5* Перечитайте задачу 12 из § 10. Разработайте проект решения следующей задачи:

Для любой пары натуральных чисел m и n составьте последовательность команд для исполнителя КВАДР, исполнив которую он за наименьшее число действий из числа m получит число n .

Реализуйте этот проект, написав соответствующие основной и вспомогательные алгоритмы.

б) На изучаемом вами языке программирования напишите программу и комплекс подпрограмм, соответствующих алгоритмам вашего проекта. Протестируйте работу вашего программного комплекса для $m = 3$, $n = 6$; $m = 8$, $n = 28$; $m = 7$, $n = 25$; $m = 1$, $n = 16$; $m = 5$, $n = 31$. Получите результаты для $m = 10$, $n = 60$; $m = 100$, $n = 101$.

- 6* Игра «Жизнь в пространстве». Эту игру на плоскости придумал Д. Конуей. Вам предлагается разработать трёхмерный аналог этой игры.

Всё пространство разбито на кубики с ребром, равным 1. В нём живут колонии покрашенных кубиков. Эти кубики рож-

даются и умирают (по определённым правилам), а их колонии эволюционируют во времени. Вот эти правила. У каждого кубика 6 соседних кубиков пространства, каждый из которых имеет с ним общую грань.

Правило 1. Если покрашенный кубик имеет четырёх или более покрашенных соседей, то он умирает (т. е. перестаёт быть покрашенным) от перенаселённости.

Правило 2. Если покрашенный кубик не имеет покрашенных соседей или имеет ровно одного такого соседа, то он умирает от недостатка общения.

Правило 3. Если не покрашенный кубик имеет ровно трёх покрашенных соседей, то он окрашивается (в этом случае говорят, что родился новый покрашенный кубик).

Время в этой игре течёт дискретно, т. е. по шагам. За один шаг с каждым кубиком происходит (или не происходит) изменение, описанное тремя выше указанными правилами.

Пусть на поле имеется некоторая колония покрашенных кубиков. Разработайте проект, позволяющий наблюдать на экране компьютера жизнь данной колонии кубиков. Продумайте, каким цветом окрашивать кубик и как именно следует его закрасить, чтобы было удобнее наблюдать за жизнью колонии.

Проследите, как эволюционируют разные колонии из четырёх, пяти, шести, семи, восемью кубиков.

Массив — это набор однотипных данных, снабжённых системой из одного или нескольких индексов, каждый из которых принимает последовательные целые значения¹. Количество используемых индексов называют **размерностью массива**, тип входящих в него элементов — **типом данного массива**.

Чтобы задать массив, надо указать его имя, тип, диапазон изменения каждого индекса. В задачнике массив будем описывать так: **вещ:** $M [1:14, 1:25, 1:70]$ ². Элемент массива также имеет имя, состоящее из имени массива и значений индексов, определяющих данный элемент.

¹ Более точно в качестве индексов может выступать любой перечисляемый тип. Поэтому, кроме целого типа, для индексирования элементов могут использоваться символьный (char) и логический (boolean) типы. Однако для простоты мы в дальнейшем в описаниях массивов будем использовать только целочисленный тип.

² В некоторых языках программирования индекс первого элемента имеет фиксированное значение. Это необходимо иметь в виду при реализации алгоритма в виде соответствующей программы.

Каждый элемент массива фактически представляет собой отдельную переменную. Поэтому к нему можно обратиться и присвоить то или иное значение точно так же, как и для самостоятельной переменной.

Одномерный массив часто называют линейным и обычно представляют в виде строки.

Двумерный массив можно представлять себе как прямоугольную таблицу; первый индекс соответствует номеру строки, второй — номеру столбца. В таблице показано, как располагаются элементы двумерного массива A [1:3, 1:4].

$A(1, 1)$	$A(1, 2)$	$A(1, 3)$	$A(1, 4)$
$A(2, 1)$	$A(2, 2)$	$A(2, 3)$	$A(2, 4)$
$A(3, 1)$	$A(3, 2)$	$A(3, 3)$	$A(3, 4)$

В алгоритмах ввод массива M мы для краткости будем записывать одной командой:

Запросить M ;

В реальных языках программирования ввод массива обычно оформляется циклом (вложенным, если массив не одномерен) в форме **Делать от ... до**.

Вопросы и задания

- 1 Сколько элементов содержит массив:
а) $A[2:7]$; б) $B[-3:6]$; в) $C[0:3, -5:0]$?
- 2 Дан алгоритм, схема которого изображена на рисунке 14.1. Какое значение имеет переменная S после исполнения этого алгоритма для массива M , описанного таблицей:

1,2	-1,5	1,6	0,8	-1	1,1	1,9	0	-0,5	0,3
-----	------	-----	-----	----	-----	-----	---	------	-----

- 3 Какие значения будут сообщены при исполнении следующего алгоритма для линейного массива M , заданного:

а) таблицей $\begin{vmatrix} -1 & 2 & -3 & 0 & 1 & -1 \end{vmatrix} ?$

б) таблицей $\begin{vmatrix} 2 & -1 & 2 & -2 & 1 & 0 \end{vmatrix} ?$

Алгоритм

```

цел  $I, M[1:6]$ ;
{ Запросить  $M[1:6]$ ;
  Делать от  $I := 2$  до 6
  {  $M[I] := M[I] + I * M[I - 1]$ ;
  }
  Делать от  $I := 1$  до 6
  { Сообщить  $M[I]$ ;
  }
}

```

- 4 Какое значение переменной K будет сообщено при исполнении следующего алгоритма для одномерного массива M , заданного: а) в задании 3а; б) в задании 3б?

Алгоритм

```

цел  $I, K, M[1:6]$ ;
{ Запросить  $M[1:6]$ ;
   $K := 0$ ;

```

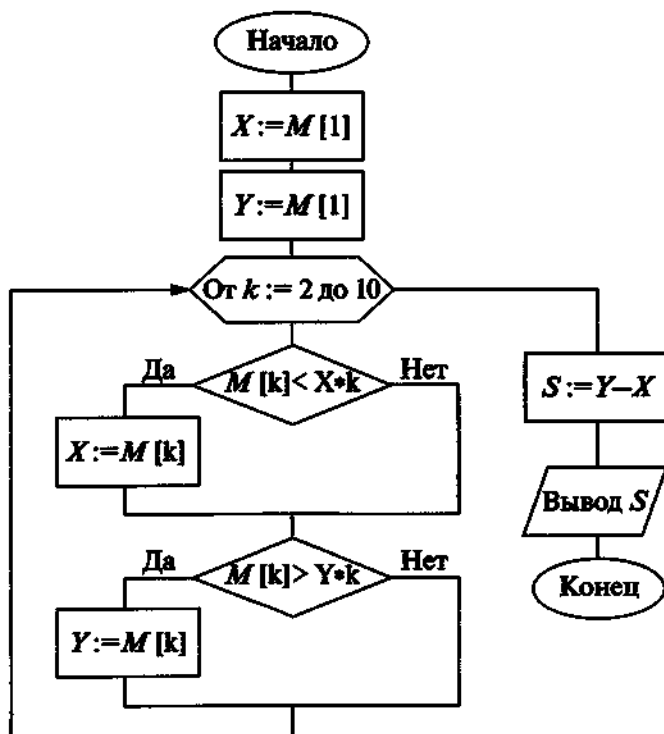


Рис. 14.1

```

    Делать от  $I := 1$  до 6
    { Если  $(M[I] < I * I)$  то {  $K := K + 1$ ; }
    }
    { Сообщить  $K$ ;
    }
}

```

- 5 Дан двумерный числовой массив $A[1:15, 1:10]$ целого типа.
- Составьте алгоритм для подсчёта количества чётных чисел в этом массиве.
 - Составьте алгоритм для подсчёта количества таких чисел в этом массиве, у которых первая цифра совпадает с последней.
 - Составьте алгоритм для подсчёта количества чисел в этом массиве, в записи которых нет других цифр, кроме 0 и 1.
 - Составьте алгоритм для подсчёта количества положительных чисел в этом массиве, у которых первая цифра больше второй.
 - Составьте алгоритм для подсчёта количества чисел, у которых цифры идут в порядке возрастания.
 - Составьте алгоритм для подсчёта количества пар равных между собой чисел в этом массиве.
 - Напишите на изучаемом вами языке программирования программы, реализующие составленные вами алгоритмы, и отладьте их.
- 6 а) Дан массив $A[1:50]$, заполненный положительными действительными числами. Составьте алгоритм, который позволяет найти наибольшее значение среди отношений $A[k]/A[m]$ для всевозможных пар элементов $A[k]$ и $A[m]$ этого массива.
- б) ■ Напишите на изученном вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.
- 7 Натуральное число называется **составным**, если оно представимо в виде произведения двух натуральных чисел, каждое из которых больше 1.
- Составьте алгоритм, с помощью которого массив $A[1:100]$ целого типа будет заполнен первыми ста составными числами.
 - Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.
- 8 а) Составьте алгоритм, с помощью которого в заданном массиве $A[1:100]$ целого типа будут найдены те элементы, которые являются степенями какого-либо простого числа¹ (не обязательно одного и того же), и сообщены их индексы или сообщено, что таких элементов нет.

¹ Определение простого числа дано в задании 11 из пункта 12.1.

б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.

- 9 а) Дан массив $A[1:50]$ логического типа. Составьте алгоритм, при помощи которого будет сообщено значение Истина, если такое значение имеет чётное число элементов массива, и значение Ложь в противном случае.

б) Решите такую же задачу для двумерного массива $B[1:15, 1:10]$.

- 10 Дан массив $A[1:50]$ логического типа. Составьте алгоритм, при помощи которого будет сообщено значение Истина, если такое значение имеет больше половины элементов массива, и значение Ложь в противном случае.

б) Решите такую же задачу для двумерного массива $B[1:10, 1:8]$.

- 11 Гусеница, сидящая на дереве, в солнечный день поднимается на 5 см вверх, если предыдущий день тоже был солнечным, а в пасмурный день спускается на 3 см вниз, если предыдущий день тоже был пасмурным, в остальных случаях она остаётся на месте. Если в какой-то момент гусеница доползает до вершины или до корня дерева, то она окукливается и больше ни в этот день, ни в последующие дни никуда не ползёт. Дан линейный массив, содержащий 30 элементов (результаты наблюдений погоды в течение 30 последовательных дней); каждый элемент — это буква П или С в зависимости от того, пасмурным или солнечным был день. Составьте алгоритм, позволяющий по начальному местоположению гусеницы на дереве высотой 70 см определить её местонахождение и состояние по истечении 30 дней.

- 12 П Дан двумерный числовой массив $A[1:25, 1:20]$ вещественного типа.

а) Составьте алгоритм нахождения такого элемента этого массива, который меньше всего отличался бы от среднего арифметического значения всех элементов массива.

б) Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.

- 13 П Дан двумерный целочисленный массив $A[1:m, 1:n]$.

а) Составьте алгоритм для подсчёта количества элементов массива, меньших среднего арифметического значения всех элементов массива, и количества элементов, больших среднего арифметического значения всех элементов массива.

б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её. Протестируйте вашу программу на массивах, заданных таблицами:

1)

1	3	5	-7
2	-4	6	8

2)

-1	2	2
5	3	7

- 14) а) Из n сопротивлений собрана электрическая цепь, в которой все сопротивления соединены параллельно (рис. 14.2). Значения сопротивлений хранятся в одномерном массиве вещественного типа. Составьте алгоритм, позволяющий найти сопротивление данной цепи.

б) Составьте алгоритм, позволяющий найти сопротивление цепи, изображённой на рисунке 14.3. Значения сопротивлений хранятся в одномерном массиве вещественного типа. (Совет: составьте подходящее рекуррентное соотношение.)

- 15) **П** Дан двумерный массив A [1:25, 1:3], в котором записаны координаты 25 точек пространства. Составить алгоритм поиска двух самых удалённых друг от друга точек.

- 16) а) Дан линейный массив A [1:100] целого типа. Составьте алгоритм, позволяющий в этом массиве найти самую длинную последовательность, состоящую из подряд идущих положительных чисел.

б) Представьте, что этот массив «склеен» в кольцо, т. е. после последнего элемента снова идёт первый. Выполните задание пункта а для такого кольца.

- 17) а) Дан линейный массив A [1:100] целого типа. Последовательность элементов массива, идущих подряд в порядке возрастания индексов, называется **возрастающей**, если в ней каждый следующий элемент больше предыдущего. Составьте алгоритм, позволяющий в этом массиве найти самую длинную возрастающую последовательность.

б) Представьте, что этот массив «склеен» в кольцо, т. е. после последнего элемента снова идёт первый. Выполните задание пункта а для такого кольца.

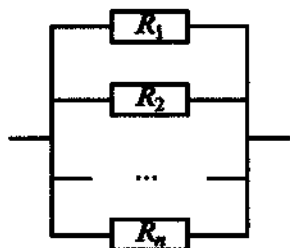


Рис. 14.2

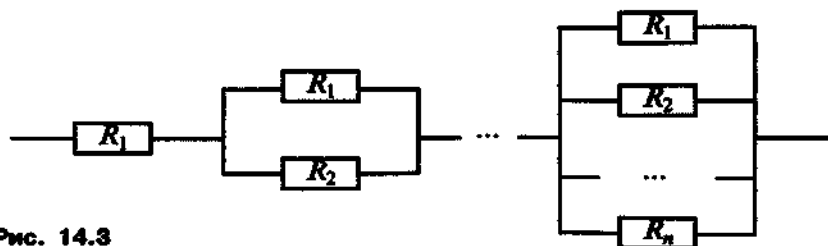


Рис. 14.3

- 18 а) Дан линейный массив $A[1:100]$ целого типа. Составьте алгоритм, позволяющий в этом массиве найти самую длинную **монотонную** последовательность, т. е. такую, которая является либо возрастающей, либо убывающей.
- б) Представьте, что этот массив «склеен» в кольцо, т. е. после последнего элемента снова идёт первый. Выполните задание пункта а для такого кольца.

§ 15

II Графы и алгоритмы на графах

Граф — это совокупность точек, называемых **вершинами**, некоторые из которых соединены линиями. Две вершины называются **смежными**, если имеется соединяющая их линия, на которой нет других вершин. Если на каждой линии, соединяющей две смежные вершины, выбрано направление, то такой граф называется **ориентированным** или сокращённо **орграфом**. Линия, соединяющая две смежные вершины, для орграфа называется **дугой**, а для неориентированного графа — **ребром**.

Допускается, чтобы дуга или ребро (в данном случае безразлично) соединяли вершину саму с собой¹.

Вершина называется **изолированной**, если она не соединена линией ни с какой другой вершиной. Вершина графа называется **инцидентной** данному ребру, если она является одним из концов этого ребра. В этом случае говорят также, что ребро инцидентно данной вершине.

Граф или орграф, у которого каждому ребру или соответственно каждой дуге сопоставлено некоторое число, называется **нагруженным**. В зависимости от рассматриваемой задачи это число может интерпретироваться как расстояние между вершинами, или как время перехода от одной вер-

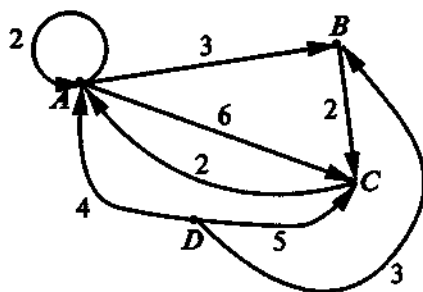


Рис. 15.1

¹ Вообще говоря, две смежные вершины могут быть соединены несколькими рёбрами. В этом случае граф называют мультиграфом. В этом параграфе мультиграфы не рассматриваются.

шины к другой, или как пропускная способность канала, соединяющего две данные вершины, или ещё каким-либо образом. Иногда бывает удобно рассматривать ненагруженный граф (орграф) как нагруженный, в котором каждому ребру (дуге) поставлено в соответствие число 1.

Обычно граф (орграф) задают одним из двух способов: перечислением всех его рёбер (дуг соответственно) или таблицей, где в клетке на пересечении строки и столбца, соответствующих данным вершинам, указано, соединены эти вершины ребром (дугой) или нет. Для нагруженного графа (орграфа) для каждого ребра (дуги) указывается нагрузка. Нагруженный орграф, изображённый на рисунке 15.1, можно задать списком дуг: $(AA; 2)$, $(AB; 3)$, $(AC; 6)$, $(BC; 2)$, $(CA; 2)$, $(DA; 4)$, $(DB; 3)$, $(DC; 5)$. В таблице приведена таблица смежности для этого графа.

	A	B	C	D
A	2	3	6	0
B	0	0	2	0
C	2	0	0	0
D	4	3	5	0

В таблице смежности ненагруженного графа везде вместо чисел, указывающих нагрузку (т. е. отличных от 0), стояло бы число 1.

В заданиях к этому пункту мы для простоты будем считать вершины графа перенумерованными натуральными числами от 1 до n (без пропусков и повторений). Список дуг для нагруженного орграфа будем задавать как двумерный массив $A[1:n, 1:3]$, где в первой строке соответствующей этому массиву таблицы указывается первый конец дуги, во второй — второй конец, а в третьей — величина нагрузки. Для неориентированного графа соответствующий массив содержит только первые две строки. Если орграф (граф) задаётся таблицей смежности, то договоримся считать значение первого индекса номером первой вершины, а второго индекса номером второй вершины; сами номера вершин в массиве не присутствуют.

Вопросы и задания

- 1) Какие вершины графа называются смежными?
- 2) Нарисуйте какой-либо граф с девятью вершинами. Составьте для него таблицу смежности. Передайте эту таблицу смежно-

сти соседу по парте и предложите ему нарисовать граф. Проверьте, правильно ли он справился с заданием.

- 3 Граф без изолированных вершин задан следующим списком рёбер: PQ; PS; PT; QS; QT; RS.
а) Изобразите этот граф.
б) Задайте этот граф таблицей смежности.
- 4 Нарисуйте какой-либо семивершинный граф без изолированных вершин. Запишите его списком рёбер. Передайте этот список соседу по парте и предложите ему нарисовать граф. Проверьте, правильно ли он справился с заданием.
- 5 а) Орграф, имеющий n вершин, задан списком рёбер. Составьте алгоритм, создающий по этому списку таблицу смежности.
б) Орграф, имеющий n вершин, задан таблицей смежности. Составьте алгоритм, создающий по этой таблице список рёбер.
в)  Напишите на изучаемом вами языке программирования программы, реализующие алгоритмы, составленные вами в пунктах а и б, и отладьте их.
- 6 Дана таблица смежности графа, но неизвестно, ориентированного ли. Составьте алгоритм, позволяющий ответить, ориентирован данный граф или нет.

15.1. Свойства графов

Путь в графе (орграфе) из вершины A в вершину B — это последовательность рёбер (соответственно дуг) $AC_1, C_1C_2, \dots, C_{n-1}B$. Количество рёбер (дуг), составляющих путь, называется длиной этого пути. Путь, начало и конец которого совпадают, называется циклом (соответственно ориентированным циклом). Цикл длины 1 называют петлёй. Орграф, изображённый на рисунке 15.1, имеет петлю около вершины A и цикл $ABCA$ длины 3.

Граф (орграф) называется связным (соответственно сильно связным), если для любых двух различных вершин существует путь, ведущий из одной вершины в другую. Орграф на рисунке 15.1 не является сильно связным (ни из какой вершины нельзя попасть в вершину D), но соответствующий ему обычный граф является связным.

Ребро связного графа, после удаления которого он перестаёт быть связным, называется мостом. Вершина связного графа, после удаления которой он перестаёт быть связным, называется точкой сочленения.

Связный граф без циклов называется деревом.

Степенью вершины графа называется количество рёбер, инцидентных данной вершине. Если ребро является петлёй, то при подсчёте степени вершины оно учитывается дважды.

Вопросы и задания

- 1 а) Докажите, что сумма степеней всех вершин любого графа равна удвоенному числу рёбер.
б) Докажите, что в любом графе количество вершин нечётной степени всегда чётно.
- 2 а) В целях борьбы с плохими дорогами правитель некоторого небольшого государства издал указ реконструировать 100 дорог, соединяющих некоторые города этого государства (каждая дорога соединяет два города, все дороги с двусторонним движением). Чтобы все города были в равном положении, мэры этих городов потребовали, чтобы из каждого города выходило ровно по 6 реконструированных дорог. Сможет ли министр транспорта и коммуникаций выполнить указ правителя с учётом требований мэров или ему лучше сразу подать в отставку?
б) Позже правитель разъяснил, что в столице будет реконструировано 10 дорог, а для остальных городов пусть будет выполнено требование мэров (количество облагороженных городов при этом, естественно, уменьшается). Сможет ли теперь министр выполнить указ?
в)* Чтобы избежать отставки, министр решил убедить правителя, что реконструкции 10 дорог для столицы недостаточно. Какое наименьшее число реконструируемых дорог в столице нужно назвать министру, чтобы указ стал выполнимым (конечно, это число должно быть больше 10)? Сколько при этом будет городов, где начнётся реконструкция дорог (считая и столицу)?
- 3 В компьютерном классе 15 компьютеров. Одна фирма предложила объединить компьютеры в сеть, соединив каждый компьютер ровно с тремя другими. Сможет ли фирма выполнить своё обещание?
- 4 Ниже приведены таблицы смежности для четырёх графов.

1)

	A	B	C	D	E	F	G	H
A			1	1		1		
B			1					
C	1	1			1	1	1	
D	1							
E			1					
F	1		1					1
G			1					
H						1		

2)

	A	B	C	D	E	F	G	H
A			1		1		1	1
B				1		1		
C	1							
D		1				1		
E	1							
F		1		1				
G	1							
H	1							

Продолжение

3)

	A	B	C	D	E	F	G	H
A				1			1	
B			1			1		
C		1			1			1
D	1						1	
E			1					
F		1						1
G	1			1				
H			1			1		

4)

	A	B	C	D	E	F	G	H
A						1		
B				1				
C				1	1			
D		1	1			1	1	1
E			1					
F	1			1				
G				1				
H				1				

а) Изобразите эти графы.

б) Укажите, какие из графов являются связными.

в) Имеется ли среди указанных графов хотя бы одно дерево?

5) а) Для каждого связного графа, полученного вами при выполнении задания 4, выясните, можно ли построить несвязный граф с тем же набором степеней вершин.

б) Для каждого несвязного графа, полученного вами при выполнении задания 4, выясните, можно ли построить связный граф с тем же набором степеней вершин

6) Существует ли шестивершинный граф с заданным набором степеней вершин:

а) 0, 1, 2, 2, 3, 4; б) 1, 2, 2, 3, 3, 5; в) 1, 1, 2, 2, 3, 4;

г) 0, 1, 2, 2, 2, 5; д) 1, 2, 2, 3, 4, 4; е) 5, 5, 5, 5, 5, 5;

ж) 1; 1; 1; 1; 1; 1; з) 2; 2; 2; 2; 2; 2?

Если ответ «Да», то нарисуйте подходящий граф, если ответ «Нет» — объясните почему..

7) Для графов, изображённых на рисунке 15.2, укажите мосты и точки сочленения.

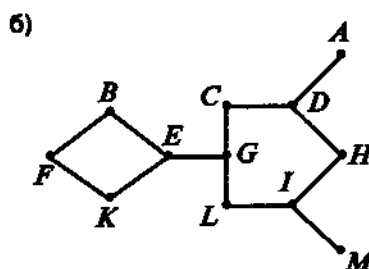
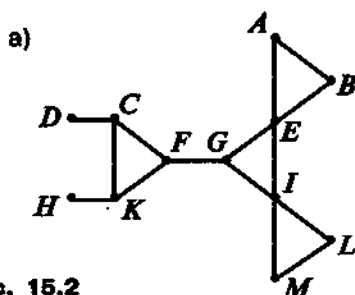


Рис. 15.2

- 8 Дан граф. Составьте алгоритм, позволяющий найти все его петли. Рассмотрите два варианта задания графа — списком рёбер и таблицей смежности.

15.2. Алгоритмы поиска на графе и орграфе

К стандартным задачам на графе относятся в первую очередь задачи обхода всех вершин графа и поиска пути от одной вершины к другой, обладающего тем или иным свойством.

Задача обхода всех вершин графа нередко возникает как задача полного перебора вариантов. Стандартными алгоритмами обхода являются алгоритмы поиска в глубину и поиска в ширину. В каждом случае обход начинается с некоторой вершины v , которая считается помеченной как пройденная. При поиске в глубину выбирается любая смежная с ней вершина. Она тоже считается помеченной и выступает в роли начальной вершины, т. е. разыскивается смежная с ней, но непомеченная вершина, которая снова рассматривается как начальная, и т. д. Если в некоторый момент нет вершин, которые смежны с рассматриваемой и не помечены как пройденные, то возвращаемся к предыдущей вершине и для неё выбирают очередную смежную непомеченную вершину. Процесс заканчивается, когда для исходной вершины v все смежные с ней вершины оказываются помеченными.

При поиске в ширину сначала рассматриваются все вершины, смежные с исходной вершиной v , и они помечаются как пройденные. Затем каждая из этих вершин рассматривается в роли начальной, и для неё разыскиваются все смежные с ней, но не помеченные ещё вершины. Обход заканчивается, когда нет непомеченных вершин, смежных с уже помеченными.

Оба алгоритма могут использоваться для определения того, является ли заданный граф связным (он будет таковым, если после завершения обхода не будет непомеченных вершин исходного графа).

Для поиска кратчайшего пути от одной вершины к другой в ненагруженном графе может использоваться поиск в ширину, при исполнении которого каждая вершина помечается номером шага, на котором она была достигнута. В этом случае исполнение алгоритма прекращается, как только достигнута нужная вершина. Такая модификация поиска в ширину называется волновым алгоритмом. Для нагруженных графов с положительной нагрузкой на каждом ребре для поиска пути с наименьшей суммарной нагрузкой применяют алгоритм Дейкстры. Его описание можно найти в литературе, указанной в конце задачника. Этот алгоритм применим и для орграфов.

Оргграф называется **бесконтурным**, если в нём нет ни одного ориентированного цикла. В бесконтурном графе вершины всегда можно перенумеровать так, что начало любой дуги будет иметь номер, меньший, чем её конец. После этого поиск путей с наименьшей (или наибольшей) суммарной нагрузкой можно выполнить алгоритмом, подобным алгоритму Дейкстры. Нагрузка на дугах бесконтурного графа может быть как положительной, так и отрицательной.

Вопросы и задания

- 1 Для проведения соревнований было проложено несколько лыжных трасс; схема этих трасс с указанием протяжённости между промежуточными пунктами разветвления трасс изображена на рисунке 15.3. Старт (общий для всех трасс) обозначен буквой С, финиш (тоже общий для всех) — буквой Ф, промежуточные пункты обозначены буквами А, Б и В.
 - а) Для ближайшего соревнования требуется разметить трассу длиной 15 км. Укажите маршруты, удовлетворяющие этому требованию.
 - б) Укажите самую длинную трассу, которую здесь можно разметить, если никакой участок не предполагается проходить дважды.
 - в) Укажите самую короткую трассу.
- 2 Напишите на изучаемом вами языке программирования программы, реализующие поиск в глубину для двух способов задания графа, и отладьте их.
- 3 Напишите на изучаемом вами языке программирования программы, реализующие поиск в ширину для двух способов задания графа, и отладьте их.
- 4 а) Составьте алгоритм, позволяющий для связного графа найти все его точки сочленения. Рассмотрите два варианта задания графа — списком рёбер и таблицей смежности.

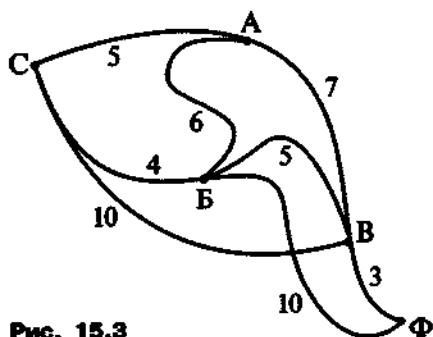


Рис. 15.3

б)  Напишите на изучаемом вами языке программирования программы, реализующие составленные вами алгоритмы (для двух способов задания графа), и отладьте их.

5 а) Дан нагруженный граф и две его вершины. Составьте алгоритм, позволяющий отыскать пути между этими вершинами с наименьшей суммарной нагрузкой (или сообщить, что эти вершины недостижимы друг для друга). Рассмотрите два способа задания графа и орграфа — списком рёбер и таблицей смежности.

б) Выполните такое же задание, как в пункте а для орграфа. Рассмотрите два способа задания графа и орграфа — списком рёбер и таблицей смежности.

в)  Напишите на изучаемом вами языке программирования программы, реализующие составленные вами алгоритмы (для двух способов задания графа или орграфа), и отладьте их.

6*  Пусть данный граф моделирует транспортную сеть между некоторыми городами. На практике каждое ребро AB имеет не только одну характеристику — время в пути, но ещё и стоимость проезда: путь из пункта A в пункт B занимает время t и имеет стоимость c . При этом для каждого ребра таких характеристик может быть несколько: из города в город можно добираться на машине, поезде, самолёте и т. д., а самый быстрый путь зачастую является и самым дорогим. Будем считать, что время в пути и стоимость проезда одинаковы для движения в обоих направлениях.

Граф задан списком рёбер, в котором для каждого ребра указано несколько пар чисел: первое число в паре — время в пути от одной вершины ребра до другой, второе число в паре — стоимость проезда по этому ребру.

а) Найдите самый быстрый путь в заданном графе между двумя заданными вершинами при условии, что на дорогу вы можете потратить не более S рублей. Если таких путей несколько, нужно минимизировать стоимость пути. Составьте алгоритм для нахождения таких путей. Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.

б) Найдите самый дешёвый путь в заданном графе между двумя заданными вершинами, при условии что на дорогу вы можете потратить не более T минут. Если таких наиболее дешёвых путей несколько, нужно минимизировать время пути. Составьте алгоритм для нахождения таких путей. Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.

в) Выгодностью маршрута называется квадратный корень из произведения стоимости маршрута на время, затрачиваемое на движение по этому маршруту. Составьте алгоритм для нахождения самых выгодных путей в заданном графе между

двумя заданными вершинами. Если таких путей несколько, то нужно минимизировать стоимость пути. Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её. (Совет: обратите внимание, что выгодность маршрута не совпадает с суммой выгодностей рёбер этого маршрута.)

- 7 а) В левом нижнем углу шахматного поля стоит конь. Составьте алгоритм, позволяющий найти кратчайший путь перемещения этого коня в другую заданную клетку шахматного поля. (Совет: подумайте, как применить здесь волновой алгоритм.)
б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.

- 8 Имеется прямоугольное клетчатое поле размером $m \times n$. Каждая клетка на этом поле задаётся номером ряда и номером столбца, на пересечении которых она находится. Некоторые клетки поля представляют собой ловушки; они занумерованы числами от 1 до K . Дан двумерный массив $A[1:K; 1:2]$, где $A[i; 1]$ — номер ряда, $A[i; 2]$ — номер столбца, в которых располагается клетка-ловушка с номером i . Фишка может перемещаться по этому полю, переходя из одной клетки в любую соседнюю с ней, если та не является ловушкой. Известно, что клетка, стоящая в левом нижнем углу этого поля, никогда не бывает ловушкой.

а) Составьте алгоритм, или позволяющий найти кратчайший безопасный путь (т. е. не проходящий ни через одну ловушку) из левого нижнего угла данного поля в его правый верхний, или, если такого пути нет, сообщаящий об этом.

б)* Составьте алгоритм, позволяющий определить клетку, наиболее удалённую от левого нижнего угла до клетки, до которой фишка может добраться, не попав ни в одну ловушку. Расстояние между клетками — это длина отрезка, соединяющего центры клеток.

в)  Напишите на изучаемом вами языке программирования программы, реализующие составленные вами алгоритмы, и отладьте их.

- 9 Имеется прямоугольное клетчатое поле размером $m \times n$. Каждая клетка на этом поле выкрашена одним из восьми цветов (будем считать, что цвета занумерованы числами от 1 до 8). Известно, что каждая угловая клетка поля закрашена своим цветом. Тем самым поле разбито на несколько (не менее четырёх) цветных областей — две клетки принадлежат одной цветной области, если они имеют общую сторону и закрашены одним цветом. Каждая клетка задаётся номером ряда и номером столбца, на пересечении которых она находится. Дан двумерный массив $A[1:m; 1:n]$, в котором $A[i; j]$ — номер цвета клетки, стоящей в i -м ряду и j -м столбце.

а) Игрок первоначально владеет цветной областью, которой принадлежит клетка, стоящая в левом нижнем углу. За один ход можно поменять цвет у всех клеток своей области. При этом цветная область, принадлежащая игроку, может увеличиться за счёт присоединения «пограничных» клеток выбранного игроком цвета. Составьте алгоритм, позволяющий определить наименьшее число ходов и порядок смены цветов, позволяющих игроку овладеть всем полем.

б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм, и отладьте её.

в)* Пусть имеется два игрока; одному из них принадлежит клетка, стоящая в левом нижнем углу; другому — клетка, стоящая в правом верхнем. Игроки ходят по очереди, и при каждом ходе игроку запрещается для перекраски своей области выбирать цвет, которым закрашена область, принадлежащая его противнику. Игра заканчивается, когда ни один из игроков не может увеличить свои владения. Выигрывает тот игрок, которому принадлежит область из большего числа клеток.

Разработайте проект, позволяющий определить, какой игрок выигрывает при заданной начальной раскраске поля и какой должна быть его стратегия.

г)*  Напишите на изучаемом вами языке программирования комплекс программ, реализующий разработанный вами проект, и отладьте их.

д)*  В эту игру могут играть 3 или 4 игрока, если каждому выделить в первоначальное владение область с угловой клеткой. Разработайте проект, позволяющий определить, какой игрок выигрывает при заданной начальной раскраске поля и какой должна быть его стратегия. Напишите на изучаемом вами языке программирования комплекс программ, реализующий разработанный вами проект, и отладьте их.

е)* Прямоугольное поле можно заменить конфигурацией из правильных шестиугольников. Тогда в игру смогут играть 6 игроков. Разработайте проект, позволяющий описать такое поле и определять при заданной начальной раскраске поля выигрывающего игрока и его стратегию.

Под игрой понимают процесс, в котором участвуют две или более стороны, ведущие борьбу за реализацию своих интересов. Каждый из игроков выстраивает собственный алгоритм поведения, определяемый правилами выполнения ходов игроками. Результатом игры является набор величин (по количеству игроков), называемых выигрышем

соответствующего игрока. Цель игры для каждого игрока может формулироваться по-разному. В большинстве игр каждый игрок стремится добиться максимального выигрыша. Однако цель игрока может состоять и в том, чтобы обеспечить себе заранее заданную гарантированную величину выигрыша. Наконец, цель может состоять в том, чтобы минимизировать выигрыш остальных игроков. Нередко функция, описывающая выигрыш игрока, принимает только два значения, которые удобно кодировать 0 и 1. При этом значение 1 означает, что данный игрок выиграл, а 0 — что он проиграл. В случае когда возможна ничья, нередко такой результат кодируют числом $\frac{1}{2}$.

Если каждый игрок в любой момент времени полностью информирован о всех предыдущих ходах, сделанных в процессе игры, и располагает информацией о всех допустимых ходах своих противников, то такая игра называется игрой с полной информацией. Игра называется конечной, если она гарантированно заканчивается за конечное число ходов. Конечную игру с полной информацией удобно изображать в виде дерева (или орграфа) с вершинами, соединёнными дугами, — каждая вершина соответствует игровой позиции, а дуга указывает переход от одной позиции к другой при совершении допустимого хода. Около концевых вершин обычно указываются значения тех выигрышей, которые получает каждый из игроков по окончании игры в этой позиции. Обычно значения выигрышей упорядочены по убыванию в соответствии с тем, в каком порядке игроки приходят в заключительную позицию игры или вообще не доходят до «финиша», но бывает и наоборот — выигрыш состоит в том, чтобы избежать попадания в заключительную позицию игры.

Выигрышной стратегией для данного игрока называется алгоритм, следуя которому он добьётся цели игры независимо от того, какие ходы будут совершены другими игроками.

Если в конечной игре с полной информацией участвуют только два игрока, выполняющие ходы последовательно, то все позиции могут быть разбиты на два класса — множество проигрышных позиций и множество выигрышных позиций. Если игрок, сделавший ход в заключительную позицию, считается выигравшим, то все заключительные позиции игры относятся к множеству проигрышных позиций (поскольку второй игрок из этой позиции уже сделать ход не может и автоматически является проигравшим). Если же игрок, сделавший ход в заключительную позицию, считается проигравшим, то все заключительные позиции игры относятся к множеству выигрышных позиций.

Остальные позиции распределяются между этими двумя множествами по следующему рекурсивному правилу: позиция считается проигрышной, если любой ход из этой позиции ведёт в выигрышную позицию; позиция считается выигрышной, если существует ход, ведущий в проигрышную позицию. На орграфе игры каждую вершину, соответствующую выигрышной позиции, надо отметить знаком «+», а каждую вершину, соответствующую проигрышной позиции, надо отметить знаком «-». Если начальная позиция отмечена знаком «+», то выигрывает игрок, делающий первый ход, и его стратегия такова: после очередного хода противника надо делать ход в ту позицию, которая отмечена знаком «-». Отметим, что при наличии такой разметки проигрышных и выигрышных позиций дуги орграфа игры для построения стратегии уже не нужны — их можно и не рисовать.

Вопросы и задания

- 1 Среди ниже перечисленных игр назовите игры с полной информацией:

а) шахматы;	б) футбол;	в) волейбол;
г) теннис;	д) крестики-нолики;	е) домино;
ж) лото;	з) денежно-вещевая лотерея.	
- 2 а) Игры, в которых игроки делают ходы строго по очереди, называются **последовательными**. Игры, в которых игроки могут совершать ходы одновременно, называются **параллельными**. Из игр, перечисленных в задании 1, укажите те, которые являются последовательными.
 б)* Может ли параллельная игра быть игрой с полной информацией?
- 3 Определите, какие из игр, перечисленных в задании 1, являются конечными, и объясните почему.
- 4 Два игрока играют в следующую игру. Перед ними лежат две кучки камней, в одной m камней, в другой n камней. За один ход разрешается увеличить число камней в любой из двух кучек на 5 или добавить в каждую из кучек по 3 камня. Игроки делают ходы по очереди. Выигрывает игрок, после хода которого либо в одной из кучек становится не менее 15 камней, либо общее число камней в обеих кучках становится не менее 20. Постройте дерево игры, если: а) $m = n = 4$; б) $m = 2$, $n = 3$. Для каждой пары значений m и n определите, какой из игроков имеет выигрышную стратегию.
- 5 Два игрока играют в следующую игру. Перед ними лежат две кучки камней, в одной m камней, в другой n камней. За один

ход разрешается увеличить число камней в любой из двух кучек в 3 раза или добавить в каждую из кучек по 2 камня. Игроки делают ходы по очереди. Выигрывает тот, кому удастся своим ходом получить суммарное количество камней в обеих кучках, большее 20. Постройте дерево игры, если: а) $m = n = 4$; б) $m = 2$, $n = 3$. Для каждой пары указанных значений m и n определите, какой из игроков имеет выигрышную стратегию.

- 6 Два игрока играют в следующую игру. Перед ними лежат три кучки камней, в первой из которых 2, во второй — 3, в третьей — 4 камня. У каждого игрока неограниченно много камней. Игроки ходят по очереди. Ход состоит в том, что игрок или удваивает число камней в какой-то кучке, или добавляет по два камня в каждую из кучек. Выигрывает игрок, после хода которого либо в одной из кучек становится не менее 15 камней, либо общее число камней во всех трёх кучках становится не менее 25. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока?
- 7 Из клетчатой бумаги вырезан прямоугольник размером $m \times n$ клеток, где $m > 1$ и $n > 1$. Играют двое. Игроки делают ходы по очереди. Первым ходом прямоугольник разрезается на две части по прямой, идущей по сторонам клеток. За один ход разрешается разрезать любой из имеющихся прямоугольников на две части по прямой, идущей по сторонам клеток. Выигрывает тот, кому удастся своим ходом отрезать квадрат со стороной 1. Постройте дерево игры, если: а) $m = 5$, $n = 7$; б) $m = 8$, $n = 9$. Для каждой пары указанных значений m и n определите, какой из игроков имеет выигрышную стратегию.
- 8 Рассмотрите ещё раз игру, описанную в задании 7.
- а)* Докажите, что если хотя бы одно из чисел m или n больше двух и чётно, то для игрока, делающего первый ход, есть выигрышная стратегия.
- б) Пусть $m = 2$. Исследуйте, при каких n выигрышная стратегия имеется у игрока, делающего ход первым, а при каких — у игрока, делающего ход вторым.
- в)* Пусть числа m и n нечётны. Исследуйте, при каких m и n выигрышная стратегия имеется у игрока, делающего ход первым, а при каких — у игрока, делающего ход вторым.
- 9 Игра Баше. Имеется полоска клетчатой бумаги, на крайней левой клетке которой стоит фишка. Игроки по очереди передвигают эту фишку на любое количество клеток вправо, но не более чем на k клеток. Выходить за пределы полосы запрещено. Проигрывает тот игрок, который не сможет сделать очередной ход.

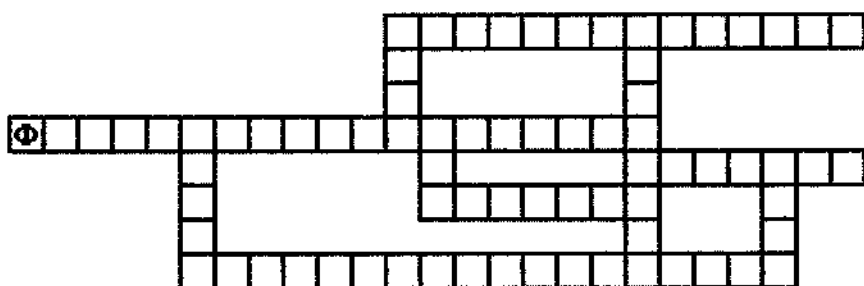


Рис. 16.1

а) Отметьте на полоске проигрышные позиции и определите, какой из игроков — первый или второй — выиграет, если полоска состоит из 15 клеток, а $k = 4$.

б) Определите, кто выиграет в этой игре, если полоска состоит из 40 клеток, а $k = 6$.

в) Сформулируйте правило, по которому можно узнать, кто выиграет в игре Баше, если полоска состоит из m клеток, а k — любое число, меньшее m . Высказанную вами гипотезу обоснуйте.

- 10 На рисунке 16.1 представлено поле, составленное из клеток. На клетке, отмеченной буквой Φ , стоит фишка. Игроки по очереди передвигают эту фишку на любое количество клеток вправо, вверх или вниз, но не более чем на k клеток. Выполняя ход, игрок может изменять направление движения, повернув на 90° (но не на 180°), но выходить за пределы поля ему запрещено. Проигрывает тот игрок, который не сможет сделать очередной ход. Если некоторая позиция повторяется трижды, то объявляется ничья.

а) Отметьте на поле проигрышные позиции и определите, имеется ли у какого-то из двух игроков — игрока, который ходит первым, или игрока, который ходит вторым, — выигрышная стратегия в этой игре при $k = 4$.

б) Выполните такое же задание, как в пункте а, при $k = 6$.

в) Нарисуйте какое-нибудь клетчатое поле и предложите соседу по парте определить, имеется ли у какого-либо из двух игроков выигрышная стратегия в этой игре для заданного вами значения k . Проверьте правильность его решения.

- 11* Два юноши в буфете угощают девушку конфетами. За один ход можно купить 1 или 2 конфеты. Конфета стоит 1 рубль, у буфетчицы 1000 конфет, у каждого юноши по 550 рублей. Проигрывает тот, кто не может сделать очередной ход. У кого из игроков имеется выигрышная стратегия?

- 12 Два игрока играют в следующую игру, делая в ней ходы по очереди. Вначале имеется число 2. За один ход разрешается имеющееся к этому ходу число увеличить на любое натураль-

ное число, меньшее его. Выигрывает игрок, получивший своим ходом число 1000.

- 13 Два игрока играют в следующую игру, делая в ней ходы по очереди. Вначале имеется число 1000. За один ход разрешается вычесть из имеющегося к этому ходу числа натуральное число, являющееся степенью числа 2. Выигрывает игрок, получивший своим ходом число 0.

- 14 Два игрока играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. Ход состоит в том, что игрок перемещает фишку из точки с координатами $(x; y)$ в одну из трёх точек: или в точку с координатами $(x + 3; y)$, или в точку с координатами $(x - 1; y + 4)$, или в точку с координатами $(x + 2; y + 2)$. Выигрывает игрок, после хода которого расстояние от фишки до точки с координатами $(0; 0)$ не меньше 11 единиц. Кто выиграет в этой игре — игрок, который ходит первым, или игрок, который ходит вторым, если начальная позиция фишки:
а) $(0, 4)$; б) $(1, 0)$; в) $(1, 2)$; г) $(1, 4)$; д) $(2, 2)$; е) $(3, -3)$?

- 15 Докажите, что игра, описанная в задании 14, конечна независимо от того, какова начальная позиция фишки.

- 16* Рассмотрим ещё раз игру, описанную в задании 14. Пусть игрок, выигрывающий в этой игре, получает приз, равный сумме координат той точки, в которой он оказался в момент выигрыша. Выигрывающий игрок стремится получить максимальный приз. Проигрывающий игрок стремится, чтобы величина приза у его противника была минимальна. Как он должен играть в каждой из тех позиций, которые указаны в задании 14, чтобы добиться своей цели? Какой приз при этом может гарантировать себе выигрывающий игрок?

Потребность в применении рассматриваемых здесь вычислительных методов нередко возникает при построении математических моделей и проведении вычислительного эксперимента с ними.

17.1. Методы приближённого решения уравнений

Пусть дана некоторая функция $f(x)$ и отрезок $[a; b]$, причём на концах этого отрезка функция принимает значения противоположных знаков. Если функция непрерывна, т. е. её график — непрерывная линия, то ясно, что график функции пересекает ось абсцисс в некоторой точке c отрезка $[a; b]$. Иными словами, $f(c) = 0$, т. е. c — корень уравнения $f(x) = 0$ (рис. 17.1).

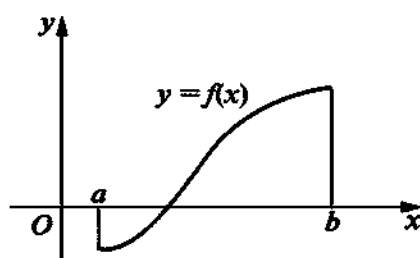


Рис. 17.1

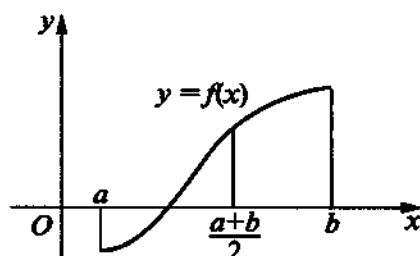


Рис. 17.2

Метод деления пополам.

Чтобы найти корень уравнения $f(x) = 0$, делим отрезок $[a; b]$ пополам, т. е. берём середину отрезка $(a + b)/2$ (рис. 17.2). В этой точке вычисляем значение функции $f(x)$. Если это значение 0, то корень найден; если нет, то оно имеет тот же знак, что и значение на одном из концов отрезка $[a; b]$. Тогда этот конец заменяем точкой $(a + b)/2$. Новый отрезок тоже содержит корень уравнения $f(x) = 0$, поскольку на его концах функция $f(x)$ снова имеет разные знаки. Этот отрезок в 2 раза короче предыдущего. И самое главное: с ним можно поступить точно так же. Со следующим отрезком ещё раз сделать то же самое и т. д.

Поскольку длина отрезка каждый раз уменьшается вдвое, можно получить отрезок сколь угодно малой длины, внутри которого содержится корень уравнения $f(x) = 0$. Концы отрезка дают приближённое значение корня с точностью, равной длине отрезка: левый конец отрезка — приближённое значение корня с недостатком; правый конец — приближённое значение корня с избытком.

Метод хорд. Этот метод решения уравнения $f(x) = 0$ состоит в следующем. Соединим концы графика функ-

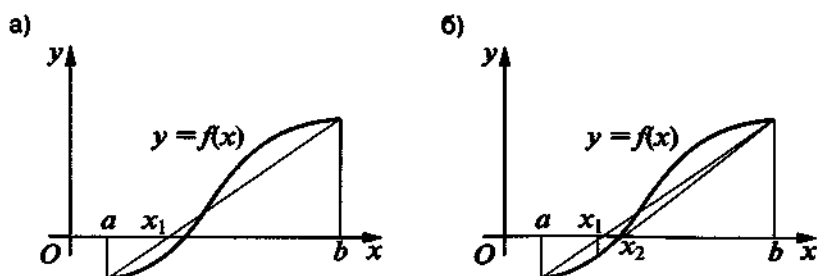


Рис. 17.3. Поиск корня методом хорд

ции $f(x)$, который построен на отрезке $[a; b]$, отрезком прямой (рис. 17.3, а). Этот отрезок пересекает ось абсцисс в некоторой точке. Возьмем её за первое приближение к корню. Затем вычисляем значение в полученной точке и снова проводим хорду; точка пересечения с осью абсцисс даёт нам следующее приближение к корню (рис. 17.3, б) и т. д.

Метод простой итерации. Для решения методом простой итерации уравнение записывают в виде $x = \varphi(x)$, после чего последовательность приближений к корню строится по формуле $x_n = \varphi(x_{n-1})$. Гарантированное приближение этой последовательности на заданном отрезке $[a; b]$ к корню обеспечивает условие $|\varphi'(x)| < 1$ для всех точек интервала $(a; b)$, где через $\varphi'(x)$ обозначена производная функции $\varphi(x)$. Уравнение $f(x) = 0$ преобразуется к виду, нужному для применения метода простой итерации, обычно следующим образом: $x = x - \alpha f(x)$, где числовой параметр α выбирается так, чтобы выполнялось указанное ограничение на производную.

Вопросы и задания

- 1 Пусть функция на концах отрезка $[1,5; 4,5]$ принимает разные знаки. За сколько шагов методом деления пополам можно получить точность 0,001?
- 2 Уравнение $x^3 - 5x + 3 = 0$ на отрезке $[0; 1]$ имеет корень (попытайтесь объяснить почему). Составьте алгоритмы нахождения этого корня:
 - а) методом деления пополам;
 - б) методом хорд;
 - в) методом простой итерации.
 Напишите на изучаемом вами языке программирования программы, реализующие составленные вами алгоритмы, и отладьте их. Найдите корень с точностью до 0,0001. Каким из алгоритмов это удалось сделать за меньшее число шагов? (Совет: для этого предусмотрите в программе счётчик.)
- 3 а) Составьте алгоритмы вычисления всех корней уравнения $x^3 - 5x + 3 = 0$ методом деления пополам и методом хорд с точностью 0,0001. (Совет: докажите, что у этого уравнения 3 корня и для каждого корня найдите такой отрезок $[a; b]$, на котором располагается только этот 1 корень.)
 - б)  Напишите на изучаемом вами языке программирования программы, реализующие составленные вами алгоритмы, и отладьте их. Найдите корни с указанной точностью.
- 4 а)  Для уравнения $\cos x = x$ найдите отрезок, содержащий его корень. Составьте алгоритмы вычисления этого корня с точностью 0,0001 методом деления пополам, методом хорд и методом простой итерации.

Напишите на изучаемом вами языке программирования программы, реализующие составленные алгоритмы, и отладьте их. Найдите корни с указанной точностью.

б) Выполните такое же задание, как в пункте а, для уравнения $2^x = 3x$.

5* а) Составьте алгоритм решения системы уравнений

$$\begin{cases} x^2 = y + 2 \cos^2 y \\ y^2 = x + 2 \cos^2 x \end{cases}$$

с точностью 0,0001. (Совет: докажите сначала, что для любых x и y , удовлетворяющих данной системе, выполнены неравенства $-1 < x < 2$ и $-1 < y < 2$.)

б)  Напишите на изучаемом вами языке программирования программу, реализующую составленный алгоритм, и отладьте её. Найдите все решения системы с указанной точностью.

17.2. Датчики случайных чисел.

Метод Монте-Карло

Пусть дана последовательность чисел, принадлежащих некоторому интервалу $(a; b)$. Эту последовательность называют равномерно распределённой в данном интервале, если для любого интервала $(x; y)$, содержащегося в $(a; b)$, частота, с которой члены последовательности попадают в этот интервал, зависит только от длины этого интервала и не зависит от того, где на $(a; b)$ этот интервал располагается. Для построения таких последовательностей используются датчики случайных чисел.

Вычислительные методы, использующие датчик случайных чисел, получили название методов Монте-Карло (по названию города, где расположена знаменитая рулетка, которую можно рассматривать как «генератор» случайных чисел). Одно из приложений метода Монте-Карло относится к приближённому вычислению площадей фигур и объёмов тел. Ниже приведено изложение указанного метода применительно к вычислению площади плоской фигуры.

Пусть дана фигура F . Поместим её в квадрат, одна вершина которого совпадает с началом координат и две его стороны располагаются на осях координат (рис. 17.4). Пусть сторона получившегося при этом квадрата равна a . Тогда его площадь равна a^2 .

Если N — общее число точек, брошенных в данный квадрат, а M — число тех точек, которые попали в фигуру F , то справедливо следующее приближённое равенство:

$$\frac{S_F}{a^2} \approx \frac{M}{N}.$$

Отсюда получаем

$$S_F = \frac{M}{N} a^2.$$

Если требуется вычислить объём пространственного тела, то оно помещается в подходящий куб.

В алгоритмах получение случайного числа из последовательности, равномерно распределённой на интервале $(0; a)$, будем записывать командой

$$x := \text{ДСЧ}(a).$$

В программе обычно требуется записать оператор инициализации датчика.

Для получения случайной точки, принадлежащей квадрату со стороной a на координатной плоскости (рис. 17.4), надо дважды применить датчик случайных чисел; для получения точки, принадлежащей кубу со стороной a , датчик надо применить трижды.

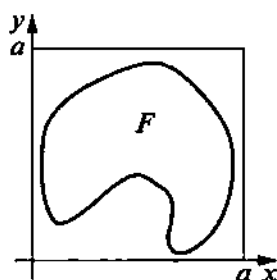


Рис. 17.4

Вопросы и задания

- 1 Рассмотрите следующий алгоритм:

Алгоритм Площадь

вещ: a, x, y ; **цел:** N, M ; **вещ:** S ;

{ **Сообщить** "Введите величину стороны квадрата";

Запросить a ;

Сообщить "Введите количество точек, бросаемых в квадрат";

Запросить N ;

$M := 0$;

Повторить N **раз**

{ $x := a * \text{ДСЧ}(1)$; $y := a * \text{ДСЧ}(1)$;

Если $(x*x + y*y \leq a*a)$ **то** { $M := M + 1$; }

}

$S := (M/N) * a * a$;

Сообщить "Площадь фигуры F приближённо равна";

Сообщить S ;

}

а) Площадь какой фигуры F подсчитывается с помощью этого алгоритма?

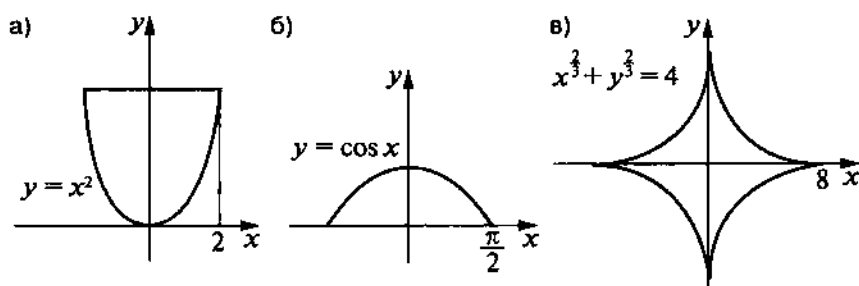


Рис. 17.5. а) сегмент параболы; б) сегмент синусоиды; в) астроида

б) На изучаемом вами языке программирования напишите программу, соответствующую этому алгоритму, и отладьте её. Проведите вычислительный эксперимент, беря $a = 2$ и меняя N от 10 000 до 1 000 000 с шагом 5000. Сравните полученные вами результаты со значением

$\pi \approx 3,141592653589793238462643383279502884197169399375$.

в) Продолжите вычислительный эксперимент с тем же значением a , меняя N от 1 000 000 до 10 000 000 с шагом 100 000. Сравните полученные вами результаты со значением π .

2 а) Составьте алгоритмы, позволяющие методом Монте-Карло найти площади фигур, изображённых на рисунках 17.5, а—в.

б) На изучаемом вами языке программирования напишите программы, соответствующие вашим алгоритмам, и отладьте их.

Найдите площади указанных фигур. Сколько случайных точек оказалось достаточно «бросить», чтобы получить значение площади с точностью до 0,001?

Замечание: точные значения площадей этих фигур:

а) $10\frac{2}{3}$; б) 2; в) 6π .

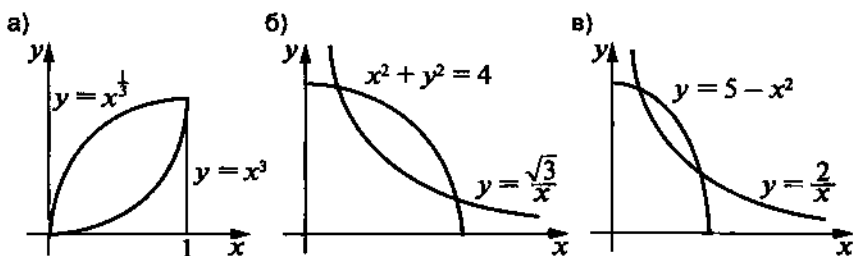


Рис. 17.6

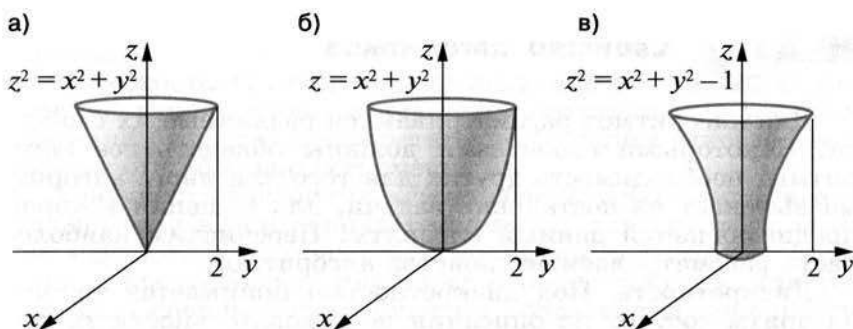


Рис. 17.7. а) конус; б) параболоид; в) гиперболоид

- 3 а) Составьте алгоритмы, позволяющие методом Монте-Карло найти площади «лунок», изображённых на рисунке 17.6, а—в.
 б) На изучаемом вами языке программирования напишите программы, соответствующие вашим алгоритмам, и отладьте их. Найдите площади указанных фигур. Сколько случайных точек оказалось достаточно «бросить», чтобы получить значение площади с точностью до 0,001? Замечание: точные значения площадей: а) 0,5; б) $\frac{\pi}{3} - 0,5 \sqrt{3} \ln 3 \approx 0,0957714003$;
 в) $14 \frac{2}{3} - \frac{20}{3} \sqrt{2} - 2 \ln 2 (\sqrt{2} + 1) \approx 2,089534716$.
- 4 а) Составьте алгоритмы, позволяющие методом Монте-Карло найти объёмы тел, изображённых на рисунке 17.7, а—в.
 б) На изучаемом вами языке программирования напишите программы, соответствующие вашим алгоритмам, и отладьте их. Найдите объёмы указанных тел. Сколько случайных точек оказалось достаточно «бросить», чтобы получить значение объёма с точностью до 0,001? Замечание: точные значения объёмов: а) $\frac{8\pi}{3}$; б) 8π ; в) $2\pi\sqrt{3}$.
- 5 Пусть имеется вспомогательный алгоритм-функция, определяющий, лежит ли точка с координатами $(x; y)$ внутри фигуры F . Вот заголовок этого алгоритма:

Алгоритм Inside (**arg** **вещ:** x, y): **лог**

Если точка с координатами $(x; y)$ лежит внутри фигуры F , то значение функции Inside — Истина, в противном случае её значение Ложь.

Составьте алгоритм нахождения площади фигуры F методом Монте-Карло с использованием указанного алгоритма-функции.

Для алгоритмов рассматриваются различные их свойства. Некоторыми свойствами должны обладать все алгоритмы, необходимость других для того или иного алгоритма вытекает из постановки задачи, для решения которой предназначается данный алгоритм. Перечислим наиболее часто рассматриваемые свойства алгоритмов.

Дискретность. Под дискретностью понимается то, что алгоритм состоит из описания последовательности тактов обработки, организованной таким образом, что в начальный момент задаётся исходная ситуация, а в каждый следующий момент ситуация преобразуется на основе данных, полученных в предшествующие такты обработки. Дискретность алгоритма означает, что он исполняется по шагам: каждое действие, предусмотренное алгоритмом, исполняется только после того, как закончилось исполнение предыдущего.

Детерминированность. Это свойство означает, что на каждом шаге исполнения алгоритма (см. свойство дискретности) однозначно определено преобразование объектов среды исполнителя, полученных на предшествующих шагах алгоритма. В некоторых учебниках это свойство алгоритма называют точностью.

Результативность. Это свойство подразумевает, что каждое действие в алгоритме после своего завершения создаёт ситуацию, в которой все имеющиеся объекты однозначно определены. Если это по каким-либо причинам невозможно, то алгоритм должен сообщить, что решения задачи не существует. Это свойство является обязательным для любого алгоритма.

Нередко говорят о результативности в целом, имея в виду, что за конечное число шагов исполнения алгоритма будет получен требуемый результат или сообщено, что такой результат получить невозможно. Поэтому результативность алгоритма в целом подразумевает и его конечность, т. е. завершение исполнения алгоритма за конечное число шагов (при этом количество шагов может быть заранее неизвестным и различным для разных начальных ситуаций).

Недетерминированный алгоритм, результат исполнения которого полностью определяется начальной ситуацией, называется неоднозначным.

То свойство алгоритма, что в нём могут использоваться только допустимые действия исполнителя, нередко называют понятностью алгоритма (в одних учебниках — эффективностью, в других — элементарностью действий).

Понятность является обязательным свойством любого алгоритма.

Массовость. С точки зрения практической ценности алгоритма важно, чтобы множество допустимых начальных ситуаций было для него достаточно большим (иногда говорят — потенциально бесконечным).

Конечность алгоритма нужно обосновывать лишь в том случае, если в нём используется цикл, для которого заранее неизвестно количество исполнений тела цикла (т. е. цикл с предусловием или цикл с постусловием), или рекурсия. Основным инструментом обоснования конечности алгоритма в этом случае служит лимитирующая функция. Так называют переменную величину, которая каждый раз меняет своё значение при очередном исполнении тела цикла или рекурсивном обращении к вспомогательному алгоритму и обладает следующими двумя свойствами:

- она принимает лишь конечное число значений (быть может, зависящее от исходных данных);
- в ходе исполнения алгоритма она принимает каждое своё допустимое значение не более одного раза.

Обоснование результативности алгоритма нередко подразумевает выявление (и доказательство) того, что именно является результатом исполнения алгоритма. Одним из приёмов исследования алгоритма на результативность является обнаружение подходящего инварианта, т. е. такой характеристики объекта, обрабатываемого с помощью данного алгоритма или его фрагмента (например, тела цикла), которая не изменяется.

Вопросы и задания

- 1 Рассмотрите следующий алгоритм:

Алгоритм Поиск

- ```
{ Задумайте однозначное число;
 Умножьте это число на 9;
 Найдите сумму цифр этого произведения;
 Из результата вычтите 4;
 Определите букву русского алфавита, номер которой
 совпадает с получившимся числом;
 Найдите с помощью Интернета названия всех госу-
 дарств, начинающихся на найденную букву;
 Упорядочьте найденные названия в алфавитном по-
 рядке;
 В первом из названий полученного списка найдите
 третью букву от начала слова;
```

Найдите с помощью Интернета всех млекопитающих, названия которых начинаются с найденной буквы и вес взрослой особи которых не менее 100 кг;

**Если** (среди найденных животных есть такие, которые живут в природных условиях найденной ранее страны) **то**

{ **Сообщить** название этого животного; }

**иначе** { **Сообщить** "Такого животного нет"; }

}

- а) Является ли этот алгоритм детерминированным?
- б) Является ли этот алгоритм однозначным?
- в) Исполните этот алгоритм.

2

В распоряжении учёного имеются марсианские амёбы трёх видов, каждая из которых находится в персональной пробирке. Он обнаружил, что при помещении двух амёб разного вида в одну пробирку они через некоторое время сливаются, образуя одну амёбу третьего вида. В целях исследования этого явления учёный исполняет следующий алгоритм:

**Алгоритм Слияние**

{ **Делать пока** (есть две амёбы разного типа)

{ Поместить двух разных амёб в одну пробирку;

Дождаться появления в этой пробирке новой амёбы;

}

}

- а) Будет ли этот алгоритм конечным?
- б) Объясните, почему этот алгоритм не является детерминированным.
- в) Приведите пример таких не нулевых начальных количеств амёб и такого исполнения алгоритма, чтобы в распоряжении учёного осталась ровно одна амёба.
- г) Всегда ли в результате исполнения алгоритма в руках учёного будет оставаться одна амёба?
- д)\* Пусть первоначально у учёного имеется 20 амёб одного вида, 21 амёба другого вида и 22 амёбы третьего вида. Может ли учёный так исполнить алгоритм, чтобы в результате осталась ровно одна амёба?

Если да, то зависит ли тип этой последней амёбы от того, как учёный выбирал амёб при исполнении тела цикла? Ответ обоснуйте.

- е)\* В результате исполнения алгоритма в распоряжении учёного осталась одна амёба. Зависит ли вид этой последней амёбы от того, как учёный выбирал амёб при исполнении тела цикла, или он предопределён начальной ситуацией? Ответ «Да» подтвердите, приведя пример начального количества амёб разных видов, для которых при исполнении алгоритма

для разных вариантов выбора амёб, помещаемых в одну пробирку, будет оставаться одна амёба, но различного вида в зависимости от варианта выбора. Ответ «Нет» обоснуйте подходящим рассуждением.

ж)\* Пусть первоначально у учёного имеется 21 амёба одного вида, 23 амёбы другого вида и 25 амёб третьего вида. Может ли учёный так исполнить алгоритм, чтобы в результате осталась ровно одна амёба? Ответ обоснуйте.

- 3) Ниже приведены алгоритмы. Укажите, какие из них обладают свойством массовости, а какие нет. Для тех алгоритмов, которые являются массовыми, постарайтесь описать множество допустимых начальных ситуаций.

а) **Алгоритм Сумма**

цел:  $K, N$ ;

вещ:  $S$ ;

{ Запросить  $N$ ;

$S := N$ ;

Делать от  $K := 1$  до  $N - 1$

{  $S := K + S$ ;

}

Сообщить  $S$ ;

}

б) **Алгоритм Преобразование в вопрос**

сим:  $A, B$ ;

{  $A := \text{"Nick will go to school next year."}$ ;

$B := \text{Часть}(A, 6, 5) + \text{Часть}(A, 1, 5) +$   
 $+ \text{Часть}(A, 11, 22) + \text{"?"}$ ;

Сообщить  $B$ ;

}

в) **Алгоритм Значение функции**

вещ:  $x, y$ ;

{ Запросить  $x$ ;

$y := \sqrt{2 \cos x - 1,7^x - 1,7^{-x}}$ ;

Сообщить  $y$ ;

}

г)\* **Алгоритм**

вещ:  $A, X$ ;

цел:  $N$ ;

{ Запросить  $A$ ;

$X := 1$ ; (\*начальное приближение\*)

$N := 0$ ; (\* $N$  — счётчик числа выполнений тела цикла\*)

```

Делать пока ($|A - X * X| > 0.0001$)
{
 $X := 0.5 * (X + A/X)$;
 $N := N + 1$;
 Сообщить N ;
 Сообщить X ; (* N -е приближение*)
}

```

- 4 На полке с книгами стоят  $n$  томов сочинений одного автора, расположенные, увы, не по порядку. Библиотекарь решил расположить их в порядке возрастания номеров, используя следующий алгоритм:

**Делать пока** (есть том, номер которого больше номера места, на котором он сейчас стоит)

```

{
 Переставить этот том на последнее место;
}

```

а) Исполните этот алгоритм для 3—4 различных расположений четырёх томов.

б) При любом ли  $n$  будет достигнута цель, если исполнение этого алгоритма когда-нибудь закончится?

в)\* Верно ли, что это исполнение этого алгоритма заканчивается за конечное число шагов при любом начальном значении  $n$ ?

- 5\* Рассмотрите алгоритм, обрабатывающий натуральное число  $n$ :

**Алгоритм Превращение**

**цел:**  $n, i$ ;

{ **Запросить**  $n$ ;

$i := 1$ ;

**Делать пока** ( $n > 1$ )

{ **Если** ( $n \bmod 3 \neq 0$ ) **то** {  $n := n + 4$ ; }

**иначе** {  $n := n/3$ ; }

$i := i + 1$ ;

}

**Сообщить**  $i$ ;

}

Закончится ли исполнение данного алгоритма за конечное число шагов? Ответ обоснуйте.

- 6\* Рассмотрите алгоритм, обрабатывающий натуральное число  $n$ :

**Алгоритм Превращение\_1**

**цел:**  $n, i$ ;

{ **Запросить**  $n$ ;

$i := 1$ ;

**Делать пока** ( $n > 1$ )

{ **Если** ( $n \bmod 3 \neq 0$ ) **то** {  $n := 2 * n + 1$ ; }

```

 иначе { $n := n/3$; }
 $i := i + 1$;
 }
 Сообщить i ;
}

```

Укажите, для каких значений  $n$ , которые вводятся пользователем по запросу компьютера, исполнение данного алгоритма заканчивается за конечное число шагов. Ответ обоснуйте.

- 7 Рассмотрите алгоритм, приведённый в задании 4 из п. 13.3. Объясните, почему он будет конечным для любого натурального  $x$ .

- 8 Элементами двумерного массива  $A$  [1:35, 1:20] являются числа 1, -1 и 0. Рассмотрите следующий алгоритм по преобразованию элементов этого массива:

#### **Алгоритм**

**цел:**  $i, j, k, m, A$  [1:35, 1:20]; **лог**  $q$ ;

{ **Запросить**  $A$  [1:35, 1:20];

$q := 1$ ;

**Делать пока** ( $q = \text{Истина}$ )

{  $q := \text{Ложь}$ ;

**Делать от**  $i := 1$ , **до** 35

{  $k := 0$ ;

$M := 0$ ;

**Делать от**  $i := 1$  **до** 20

{ **Если** ( $A[i, j] = 1$ ) **то** {  $k := k + 1$ ; }

**Если** ( $A[i, j] = -1$ ) **то** {  $m := m + 1$ ; }

} (\*конец цикла\*)

**Если** ( $k > m$ ) **то**

{ **Делать от**  $j := 1$  **до** 20

{  $A[i, j] := -A[i, j]$ ;

}

$q := \text{Истина}$ ;

}

}

**Делать от**  $j := 1$  **до** 20

{  $k := 0$ ;

$m := 0$ ;

**Делать от**  $i := 1$  **до** 35

{ **Если** ( $A[i, j] = 1$ ) **то** {  $k := k + 1$ ; }

**Если** ( $A[i, j] = -1$ ) **то** {  $m := m + 1$ ; }

}

**Если** ( $k > m$ ) **то**

{ **Делать от**  $i := 1$  **до** 35

```

 { $A[i, j] := -A[i, j];$
 }
 $q := \text{Истина};$
 }
}

```

(\*конец цикла\*)  
 (\*конец ветвления\*)  
 (\*конец цикла\*)  
 (\*конец цикла\*)

Верно ли, что при любом начальном заполнении массива  $A$  алгоритм завершает свою работу за конечное число шагов? (Совет: чтобы легче было понять, как преобразуется массив, напишите программу и поэкспериментируйте с ней.)

- 9\* Рассмотрите алгоритм, приведённый в задании 3г. Объясните, почему он будет конечным для любого положительного  $A$ .
- 10\* Дан алгоритм, применяемый к натуральным значениям переменной  $x$ :

**Алгоритм** Особое суммирование

цел:  $x, y, z$ ;

{ **Сообщить** "Введите натуральное число";

**Запросить**  $x$ ;

$y := x + S(x)$ ;

**Делать пока**  $(x \bmod 9 \neq y \bmod 9)$

{  $z := y + S(y)$ ;

$x := y$ ;

$y := z$ ;

}

**Сообщить**  $x$ ;

}

Через  $S$  в этом алгоритме обозначен следующий алгоритм-функция:

**Алгоритм**  $S$  (арг цел:  $x$ ): цел

{ **Если**  $(x < 8)$  **то** {  $S := x$ ; }

**иначе** {  $S := (x \bmod 8) - S(x \operatorname{div} 8)$ ; }

}

а) Сколько существует различных начальных значений переменной  $x$ , не превосходящих 100, для которых алгоритм «Особое суммирование» заканчивает работу за конечное число шагов?

б) Сколько раз в этом алгоритме выполняется тело цикла для тех начальных значений переменной  $x$ , для которых алгоритм «Особое суммирование» заканчивает работу за конечное число шагов?

- 11 Дан фрагмент схемы алгоритма (рис. 18.1).

а) Закончится ли исполнение этого алгоритма за конечное число шагов, если ввести  $a = 3$  и  $b = 7$ ? А если  $a = 7$  и  $b = 3$ ?

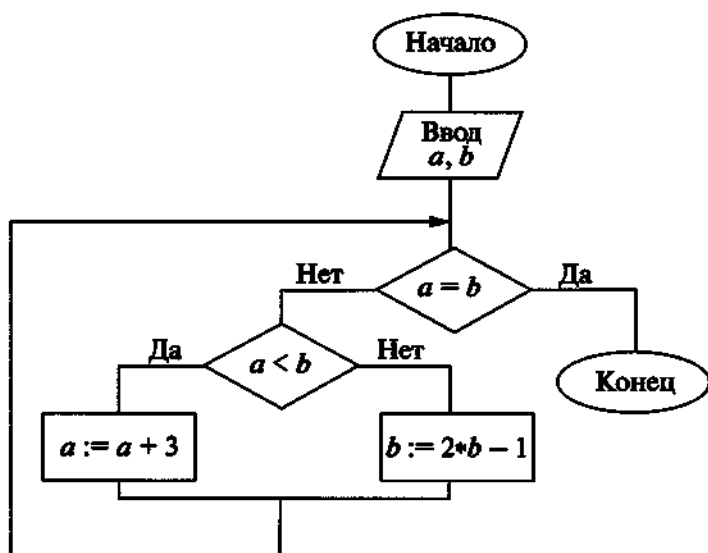


Рис. 18.1

б) Приведите пример таких неравных между собой начальных значений  $a$  и  $b$ , для которых исполнение этого алгоритма заканчивается за конечное число шагов.

в)\* Сколько существует различных пар начальных значений переменных  $a$  и  $b$ , удовлетворяющих неравенствам  $0 \leq a \leq 100$ ,  $0 \leq b \leq 100$  и для которых этот алгоритм заканчивает работу за конечное число шагов?

12) Рассмотрите следующий алгоритм:

**Алгоритм** Генератор

**цел:**  $k, n, m, x$ ; **лог:**  $q$ ;

{  $x := 1$ ;

**Сообщить** "Введите натуральное число  $n$ ";

**Запросить**  $n$ ;

$q := \text{Истина}$ ;

**Делать пока** ( $q = \text{Истина}$ )

{  $m := x * (x + 1) + n$ ;

$k := 2$ ;

**Делать пока** ( $k * k \leq m$ ) **и** ( $q = \text{Истина}$ )

{ **Если** ( $m \bmod k = 0$ ) **то** {  $q := \text{Ложь}$ ; }

$k := k + 1$ ;

}

**Если** ( $q = \text{Истина}$ ) **то** {  $x := x + 1$ ; }

}

**Сообщить**  $x$ ;

}

а) Исследуйте, для каждого ли натурального  $n$  этот алгоритм завершает работу за конечное число шагов. Ответ обоснуйте.

б)  Напишите на изучаемом вами языке программирования программу, реализующую данный алгоритм, и отладьте её.

Проведите вычислительный эксперимент, задавая значения  $n$  от 1 до 50. Сколько раз и для каких  $n$  будет сообщено значение  $x$ , равное  $n - 2$ ?

в)\* Верно ли следующее утверждение: «Если после ввода числа  $n$  сообщено число  $n - 2$ , то число  $n$  простое»? Верно ли обратное утверждение?

- 13 Пусть  $N$  — натуральное число. Составьте алгоритм, позволяющий найти наименьшее число, записанное с использованием только цифры 1, которое делится на  $N$ , или сообщающий, что такого числа не существует.

б)\* Докажите, что составленный вами алгоритм заканчивает работу за конечное число шагов, сообщив при этом правильный результат.

в)\*  Напишите на изучаемом вами языке программирования программу, реализующую составленный вами алгоритм. Проведите вычислительный эксперимент для  $N = 3; 11; 17; 31$ .

- 14 Рассмотрите следующий алгоритм:

**Алгоритм** Квадратичный\_феномен

**вещ:**  $d, p, q$ ; **цел:**  $N$ ;

{ **Сообщить** "Введите число  $p$ ";

**Запросить**  $p$ ;

**Сообщить** "Введите число  $q$ ";

**Запросить**  $q$ ;

$N := 0$ ;

$d := p * p - 4 * q$ ;

**Делать пока** ( $d > 0$ )

    {  $q := (-p + \text{sqrt}(d))/2$ ;

$p := -p - q$ ;

$d := p * p - 4 * q$ ;

$N := N + 1$ ;

    }

**Сообщить**  $N$ ;

}

а) Исполните этот алгоритм для  $p = 6$  и  $q = 5$ .

б)\* Верно ли, что этот алгоритм завершает работу за конечное число шагов при любых начальных значениях  $p$  и  $q$ ?

Среда, в которой работает машина Тьюринга, представляет собой бесконечную (в обе стороны) ленту, разбитую на одинаковые секции. Во время работы машина движется вдоль ленты, смещаясь каждый раз точно на одну секцию вправо или влево. При этом машина может находиться в различных состояниях. Этих состояний конечное число, и для их обозначения мы будем использовать символы  $q_0, q_1, \dots, q_n$ . Набор символов, кодирующих состояния машины, называют внутренним алфавитом машины Тьюринга. Кроме того, имеется внешний алфавит, символы которого могут вписываться в секции ленты и распознаваться. Пусть это будут  $a_0, a_1, \dots, a_m$ . В любой момент работы машины в каждой секции хранится ровно один знак внешнего алфавита. К началу работы машины на ленту записывается исходная информация (т. е. последовательность символов внешнего алфавита). В ходе работы машины эта последовательность изменяется. Если после выполнения конечного числа действий машина прекращает работу (будем говорить — останавливается), то на ленте оказывается записанной новая последовательность символов, которую и считаем результатом обработки исходной информации.

Допустимые действия машины Тьюринга таковы:

- записать какой-либо символ внешнего алфавита в секцию ленты (символ, бывший там до того, затирается);
- сместиться в соседнюю секцию;
- сменить состояние на одно из обозначенных символом внутреннего алфавита.

Конечно, в этом перечислении в каждой строке указано не одно действие, а группа действий одного вида — действий «записать символ внешнего алфавита» столько, сколько этих символов, сместиться в соседнюю секцию можно вправо, а можно влево, действий по изменению состояния столько, сколько этих состояний (т. е. сколько символов внутреннего алфавита). Как видно из этого описания, машина Тьюринга «бегает» вдоль ленты и заменяет одни символы на другие.

Каждая команда машины имеет вид:

указание  
о записи сим-  
вола

указание  
о перемещении  
машины

указание  
о смене со-  
стояния

Фактически команды выглядят так:

$a_i \rightarrow q_j$  — в обозреваемую секцию записать  $a_i$ , сместиться вправо (к следующей секции) и перейти в состояние  $q_j$ ;

$a_i \leftarrow q_i$  — в обозреваемую секцию записать  $a_i$ , сместиться влево (к следующей секции) и перейти в состояние  $q_j$ ;  
 $a_i Sq_j$  — в обозреваемую секцию записать  $a_i$ , остановиться и перейти в состояние  $q_j$ .

Для осуществления действий машина Тьюринга имеет операционно-логический блок. У этого блока имеется два входа: через один из них поступает информация о том, какой символ стоит в обозреваемой ячейке, через другой — символ, обозначающий состояние, в котором находится машина на данном шаге своей работы. Эта информация однозначно определяет, какую именно команду следует выполнить машине. Поскольку команда может содержать указание на выполнение трёх действий — записи символа на ленту, смещения и смены состояния, операционно-логический блок имеет три выхода: запись символа  $A$ , смещение  $M$  и смена состояния  $Q$  (рис. 19.1).

Рис. 19.1. Операционно-логический блок машины Тьюринга

Поскольку у машины Тьюринга два входа, её реакцию на подаваемые символы удобно представлять в виде прямоугольной таблицы, в которой строки и столбцы помечены символами внешнего и внутреннего алфавитов соответственно (см. таблицу). В клетки таблицы записываются команды машины. Если машина находится напротив секции, где написано  $a_i$ , а её состояние при этом есть  $q_j$ , то выполняется команда, стоящая на пересечении строки, отмеченной символом  $a_i$ , и столбца, отмеченного символом  $q_j$ . Если же в клетке таблицы ничего не написано, то машина прекращает работу.

Функциональная схема машины Тьюринга

| Внешний алфавит $A$ | Состояния машины $Q$ |       |     |                       |     |       |
|---------------------|----------------------|-------|-----|-----------------------|-----|-------|
|                     | $q_0$                | $q_1$ | ... | $q_j$                 | ... | $q_n$ |
| $a_0$               |                      |       |     |                       |     |       |
| $a_1$               |                      |       |     |                       |     |       |
| ...                 |                      |       |     |                       |     |       |
| $a_i$               |                      |       |     | $a_i \rightarrow q_i$ |     |       |
| ...                 |                      |       |     |                       |     |       |
| $a_m$               |                      |       |     |                       |     |       |

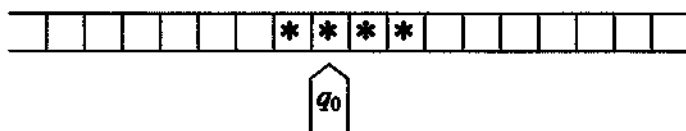


Рис. 19.2. Машина Тьюринга перед началом работы

Эту таблицу называют функциональной схемой данной машины; она-то и играет роль инструкции (программы) для машины Тьюринга.

Договоримся, что в начале работы машина всегда находится в состоянии  $q_0$  и может располагаться напротив любой секции ленты. На рисунке 19.2 изображена машина Тьюринга перед началом работы при некотором заполнении ленты (слева и справа от символов «\*» вся лента расписана символами «пробел», который мы никак изображать не будем). Эта договорённость означает, в частности, что «пробел» обязательно входит в число символов внешнего алфавита. Для краткости мы будем обозначать его во всех функциональных схемах как  $a_0$ . Кроме того, положение машины Тьюринга мы в дальнейшем будем указывать просто ↑.

### Вопросы и задания

- На ленте записана последовательность из  $n$  символов «\*» ( $n$  — натуральное число).
  - Составьте функциональную схему машины Тьюринга, с помощью которой на ленте вместо исходной последовательности будет записана последовательность, состоящая из  $2n$  подряд идущих звёздочек.
  - Составьте функциональную схему машины Тьюринга, с помощью которой на ленте вместо исходной последовательности будет записана последовательность из  $\left[ \frac{n+1}{2} \right]$  подряд идущих звёздочек. (Квадратные скобки обозначают целую часть числа.)
- Пусть на информационной ленте дана следующая запись:

|  |  |  |  |  |   |   |   |   |   |   |   |  |  |  |  |  |  |
|--|--|--|--|--|---|---|---|---|---|---|---|--|--|--|--|--|--|
|  |  |  |  |  | 8 | 7 | 4 | + | 6 | 2 | 9 |  |  |  |  |  |  |
|--|--|--|--|--|---|---|---|---|---|---|---|--|--|--|--|--|--|

Её требуется преобразовать в запись:

|  |  |  |  |  |   |   |   |   |   |   |   |   |   |   |   |   |  |
|--|--|--|--|--|---|---|---|---|---|---|---|---|---|---|---|---|--|
|  |  |  |  |  | 8 | 7 | 4 | + | 6 | 2 | 9 | = | 1 | 5 | 0 | 3 |  |
|--|--|--|--|--|---|---|---|---|---|---|---|---|---|---|---|---|--|

а) Какой внешний алфавит требуется для решения этой задачи?

б) Составьте функциональную схему, следуя которой машина Тьюринга решит данную задачу. Сколько символов внутреннего алфавита пришлось вам использовать?

в) Годится ли составленная вами схема, чтобы вычислить сумму любых двух натуральных чисел. Если нет, составьте схему, которая бы решала такую задачу.

3 Пусть внешний алфавит машины Тьюринга содержит символ  $a_0$ , все цифры, знаки  $-$  и  $=$ . Составьте функциональную схему, следуя которой машина Тьюринга выполнит вычитание двух заданных натуральных чисел (результат может оказаться отрицательным числом).

4\* Пусть внешний алфавит машины Тьюринга содержит символ  $a_0$ , все цифры, знаки  $\times$  и  $=$ . Составьте функциональную схему, следуя которой машина Тьюринга выполнит умножение натурального числа на однозначное натуральное число.

5 Внешний алфавит состоит из символа  $a_0$  и цифр  $0, 1, 2, \dots, 9$ . На ленту записано натуральное число. Составьте функциональную схему машины Тьюринга, с помощью которой на ленте будет записана сумма цифр этого числа. Ответ требуется записать правее исходного числа через пробел.

6 Внешний алфавит состоит из символа  $a_0$  и цифр  $0, 1, 2, \dots, 9$ . На ленту записано натуральное число в двоичной системе счисления. Составьте функциональную схему машины Тьюринга, с помощью которой на ленте будет то же число в десятичной системе счисления. Ответ требуется записать правее исходного числа через пробел.

7 Пусть внешний алфавит состоит из символа  $a_0$ , символа « $\cdot$ » и цифр « $0$ » и « $1$ ». На ленту записывается целое число в двоичной системе или конечная двоичная дробь.

а) Придумайте машину Тьюринга и составьте для неё функциональную схему, согласно которой данное число будет увеличено вдвое. Если после удвоения получается целое число, то в его записи не должно быть символа « $\cdot$ ». Например, запись на информационной ленте

|  |  |   |   |   |  |  |  |
|--|--|---|---|---|--|--|--|
|  |  | 1 | 0 | 1 |  |  |  |
|--|--|---|---|---|--|--|--|

должна быть преобразована в запись

|  |  |   |   |   |   |  |  |
|--|--|---|---|---|---|--|--|
|  |  | 1 | 0 | 1 | 0 |  |  |
|--|--|---|---|---|---|--|--|

а запись

|  |  |   |   |   |   |  |  |
|--|--|---|---|---|---|--|--|
|  |  | 1 | 0 | , | 1 |  |  |
|--|--|---|---|---|---|--|--|

должна быть преобразована в запись

|  |   |   |   |   |  |  |
|--|---|---|---|---|--|--|
|  | 1 | 0 | 0 | 1 |  |  |
|--|---|---|---|---|--|--|

б) Придумайте машину Тьюринга и составьте для неё функциональную схему, согласно которой данное число будет уменьшено вдвое. Если после этого снова получается целое число, то в его записи не должно быть символа «,».

8\* Пусть внешний алфавит машины Тьюринга содержит символ  $a_0$  и все гласные буквы русского алфавита.

а) Составьте функциональную схему, следуя которой машина Тьюринга расположит в алфавитном порядке заданную на информационной ленте последовательность гласных букв (последовательность не содержит одинаковых букв). Например, запись на информационной ленте

|  |  |  |  |  |   |   |   |   |   |  |  |  |  |  |  |  |  |
|--|--|--|--|--|---|---|---|---|---|--|--|--|--|--|--|--|--|
|  |  |  |  |  | у | е | ы | а | о |  |  |  |  |  |  |  |  |
|--|--|--|--|--|---|---|---|---|---|--|--|--|--|--|--|--|--|

должна быть преобразована в запись

|  |  |  |  |  |  |  |   |   |   |   |   |  |  |  |  |  |  |
|--|--|--|--|--|--|--|---|---|---|---|---|--|--|--|--|--|--|
|  |  |  |  |  |  |  | а | е | о | у | ы |  |  |  |  |  |  |
|--|--|--|--|--|--|--|---|---|---|---|---|--|--|--|--|--|--|

Совет: в условии не требуется, чтобы результат располагался в точности на тех же местах, где стояла исходная последовательность; подумайте, как этим можно воспользоваться для облегчения решения задачи.

б) Выполните такое же задание, как в пункте а, но допуская теперь, что буквы в последовательности могут повторяться.

## 18 20 II Языки программирования

В этом параграфе представлено несколько заданий, которые предназначены для начального освоения языка (в том числе, его синтаксиса) и некоторых приёмов программирования. Поэтому все задания выполняются без компьютера. По тем же причинам в этом параграфе нет заданий на разработку программ (тем более что таковые в достаточном, на наш взгляд, количестве и ассортименте присутствуют в других параграфах этой главы).

Все программы приводятся на трёх языках: Бейсик, Паскаль и Си. Внутри одного задания эти программы идентичны — при выполнении задания нужно только выбрать ту, которая записана на изучаемом вами языке.

# **Вопросы и задания**

- 1 Дан фрагмент программы, в которой переменные  $a$  и  $b$  имеют вещественный тип:

| Бейсик          | Паскаль            | Си               |
|-----------------|--------------------|------------------|
| $a = 5$         | $a := 5;$          | $a = 5;$         |
| $b = 5 + 7*a$   | $b := 5 + 7*a;$    | $b = 5 + 7*a;$   |
| $b = a + b/2*a$ | $b := a + b/2 *a;$ | $b = a + b/2*a;$ |

Какое значение получит переменная  $b$  после исполнения этого фрагмента?

- 2 Дан фрагмент программы, в которой переменные  $a$ ,  $b$  и  $k$  имеют целый тип:

|         |                                                                                                                                                                                                                            |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Бейсик  | $a = 3$<br>$b = 2$<br><b>FOR</b> $k = 1$ <b>TO</b> 4<br><b>IF</b> $((b > k) \text{ OR } (a < i))$ <b>THEN</b> $a = b - k$<br><b>ELSE</b> $b = a + k$<br><b>NEXT</b> $k$<br><b>PRINT</b> $a, b$                             |
| Паскаль | $a := 3;$<br>$b := 2;$<br><b>for</b> $k := 1$ <b>to</b> 4 <b>do</b><br><b>if</b> $(b > k)$ <b>or</b> $(a < k)$ <b>then</b> $a := b - k$<br><b>else</b> $b := a + k;$<br><b>writeln</b> ( $a$ );<br><b>writeln</b> ( $b$ ); |
| Си      | $a = 3;$<br>$b = 2;$<br><b>for</b> ( $k = 1; k \leq 4; k++$ )<br><b>if</b> $(b > k)$ <b>or</b> $(a < k)$ <b>then</b> $a := b - k;$<br><b>else</b> $b := a + k;$<br><b>printf</b> ( $a$ );<br><b>printf</b> ( $b$ );        |

Какое значение получат переменные  $a$  и  $b$  после исполнения этого фрагмента?

- 3 В программе описан одномерный целочисленный массив  $A$  с индексами от 0 до 10,  $i$  и  $s$  — переменные целого типа. Ниже представлен фрагмент этой программы:

| Бейсик                                                                                                                               | Паскаль                                                                                                                             | Си                                                                                                                                                  |
|--------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>FOR i = 0 TO 10 A (i) = i + 1 NEXT i FOR i = 0 TO 10 A (i) = A (10 - i) NEXT i s = 0 FOR i = 0 TO 10 s = s + A (i) NEXT i</pre> | <pre>for i := 0 to 10 do A [i] := i + 1; for i := 0 to 10 do A [i] := A [10 - i]; s := 0; for i := 0 to 10 do s := s + A [i];</pre> | <pre>for (i = 0; i &lt;= 10; i++) A [i] = i + 1; for (i = 0; i &lt;= 10; i++) A [i] = A [10 - i]; for (i = 0; i &lt;= 10; i++) s = s + A [i];</pre> |

Какое значение получит переменная  $s$  после исполнения этого фрагмента?

- 4 В программе описан одномерный целочисленный массив  $A$  с индексами от 0 до 10,  $i$  и  $s$  — переменные целого типа. Ниже представлен фрагмент этой программы:

| Бейсик                                                                                                                                      | Паскаль                                                                                                                               | Си                                                                                                                                                |
|---------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>FOR i = 0 TO 10 A (i) = i - 1 NEXT i FOR i = 9 TO 0 STEP - 1 A (i) = A (i + 1) NEXT i s = 0 FOR i = 0 TO 10 s = A (i) - s NEXT i</pre> | <pre>for i := 0 to 10 do A [i] := i - 1; for i := 9 downto 0 do A [i] := A [i + 1]; s := 0; for i := 0 to 10 do s := A [i] - s;</pre> | <pre>for (i = 0; i &lt;= 10; i++) A [i] = i + 1; for (i = 9; i &gt;= 0; i--) A [i] = A [i + 1]; for (i = 0; i &lt;= 10; i++) s = A [i] - s;</pre> |

Какое значение получит переменная  $s$  после исполнения этого фрагмента?

- 5 В программе описан двумерный целочисленный массив  $A$ , оба индекса которого меняются от 0 до 7,  $i$  и  $j$  — переменные целого типа. Ниже представлен фрагмент этой программы, в котором заполняется массив:

| Бейсик                                                                               | Паскаль                                                                                       | Си                                                                                                      |
|--------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------|
| <pre>FOR i = 0 TO 7 FOR j = 0 TO 7 A (i, j) = j - i A (j, i) = i NEXT j NEXT i</pre> | <pre>for i := 0 to 7 do for j := 0 to 7 do begin A [i, j] := j - i; A [j, i] := i; end;</pre> | <pre>for (i = 0; i &lt;= 7; i++) for (j = 0; j &lt;= 10; j++) { A [i, j] = j - i; A [j, i] = i; }</pre> |

Какое число в этом массиве встречается наибольшее число раз?

- 6 Требовалось написать программу, которая вводит с клавиатуры координаты точки на плоскости ( $x, y$  — действительные числа) и определяет принадлежность точки заштрихованной области, включая её границы (рис. 20.1). Программист торопился и запрограммировал неверный алгоритм.

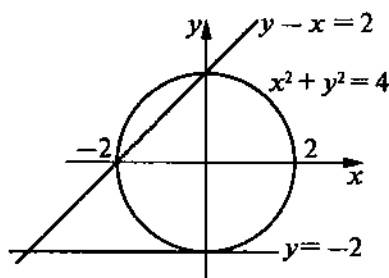


Рис. 20.1

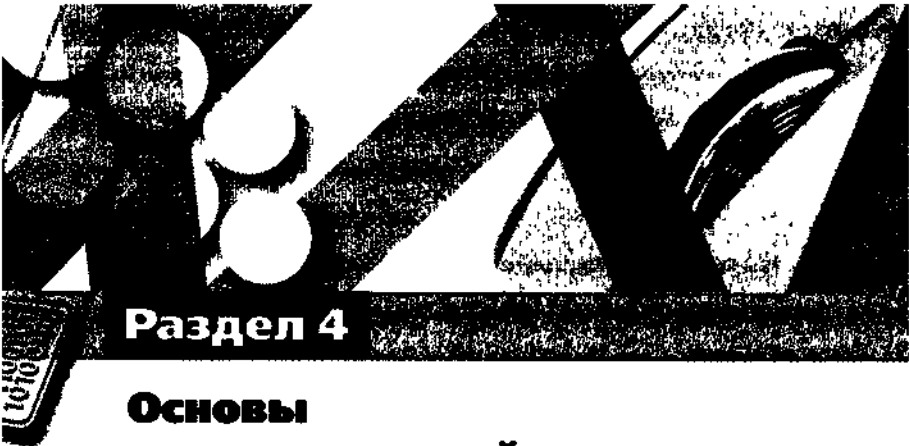
1) Приведите пример таких чисел  $x, y$ , при которых программа неверно решает поставленную задачу.

2) Доработайте эту программу так, чтобы не было случаев её неправильной работы. (Такая доработка подразумевает включение в программу ещё нескольких операторов, позволяющих устранить ошибки алгоритмизации.)

|        |                                                                                                                                                |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------|
| Бейсик | <pre>INPUT x, y IF x*x + y*y &gt;= 4 THEN IF y &gt;= -2 THEN IF y - x &lt;= 2 THEN PRINT "принадлежит" ELSE PRINT "не принадлежит" ENDIF</pre> |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------|

*Продолжение*

|         |                                                                                                                                                                                                                                         |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|         | <b>ENDIF</b><br><b>ENDIF</b><br><b>END</b>                                                                                                                                                                                              |
| Паскаль | <pre> var x, y: real; begin   readln (x, y);   if x*x + y*y &gt;= 4 then     if y &gt;= -2 then       if y - x &lt;= 2 then         write ('принадлежит')       else         write ('не принадлежит')       end.     end.   end. </pre> |
| Си      | <pre> void main(void) { float x, y;   scanf ("%f %f",&amp;x,&amp;y);   if (x*x + y*y &gt;= 4)     if (y &gt;= -2)       if (y - x &lt;= 2)         printf ("принадлежит");       else         printf ("не принадлежит");     } } </pre> |



## Раздел 4

### Основы вычислительной техники

В компьютере вся информация представляется в двоичном коде. Это положение называют принципом двоичного кодирования — одним из трёх так называемых принципов фон Неймана, лежащих в основе вычислительной техники. Соответственно обработка информации в компьютере — это преобразование исходной последовательности над двухсимвольным алфавитом в результирующую последовательность. Символы алфавита договоримся обозначать «0» и «1». Технически двухсимвольный алфавит может быть реализован по-разному: низкий и высокий потенциал напряжения, отсутствие или наличие намагниченности, низкая или высокая интенсивность светового потока. Для осуществления такого двоичного кодирования напряжение на все логические элементы компьютера подаётся дискретно — импульсами — с определённой тактовой частотой. Появление импульса на входе логического элемента кодирует подачу символа «1» на этот вход. Отсутствие импульса на входе кодирует подачу символа «0».

Средством для математического описания работы логических элементов является язык математической логики. Поэтому параграфы, посвящённые различным аспектам основ вычислительной техники, предваряются параграфами, относящимися к математической логике. В них вместо 1 и 0 используются традиционные для математической логики символы И и Л. Первый из них означает истинность высказывания, второй — его ложность. Элементы математической логики представляют самостоятельный интерес, а многие предлагаемые задания близки к тем, которые применяются в ЕГЭ по информатике.

## § 21

## Элементы математической логики

Одну из простейших моделей рассуждения представляет математическая логика, основным понятием которой является «высказывание». Над высказываниями можно выполнять операции, о которых кратко рассказывается в п. 21.1.

## 21.1. Алгебра логики

**Высказывание** — повествовательное предложение, о котором можно сказать, истинно оно или ложно. Договоримся высказывания обозначать большими латинскими буквами. Если  $A$  и  $B$  — некоторые высказывания, то высказывание « $A$  и  $B$ » называется конъюнкцией указанных высказываний и обозначается  $A \wedge B$ , высказывание « $A$  или  $B$ » называется дизъюнкцией указанных высказываний и обозначается  $A \vee B$ , высказывание «не  $A$ » называется отрицанием высказывания  $A$  и обозначается  $\neg A$ , высказывание «если  $A$ , то  $B$ » называется импликацией высказываний  $A$  и  $B$  и обозначается  $A \rightarrow B$ , высказывание « $A$  тогда и только тогда, когда  $B$ » называется эквивалентностью и обозначается  $A \leftrightarrow B$ . Символы  $\wedge$ ,  $\vee$ ,  $\neg$ ,  $\rightarrow$ ,  $\leftrightarrow$  называются логическими связками.

Зависимость значений истинности полученных с помощью связок высказываний от значений истинности высказываний  $A$  и  $B$  определяется таблицей истинности связок.

Таблица истинности связок

| $A$ | $B$ | $A \wedge B$ | $A \vee B$ | $\neg A$ | $A \rightarrow B$ | $A \leftrightarrow B$ |
|-----|-----|--------------|------------|----------|-------------------|-----------------------|
| И   | И   | И            | И          | Л        | И                 | И                     |
| И   | Л   | Л            | И          | Л        | Л                 | Л                     |
| Л   | И   | Л            | И          | И        | И                 | Л                     |
| Л   | Л   | Л            | Л          | И        | И                 | И                     |

Если  $A_1, A_2, \dots, A_n$  — обозначения некоторых высказываний, то выражение, полученное из них с помощью связок, называется формулой. В формуле могут использоваться скобки для указания порядка применения связок. Чтобы писать меньше скобок, договариваются, что связка  $\neg$  старше связки  $\wedge$ , а связка  $\wedge$  старше связки  $\vee$ , которые, в свою очередь, старше связок  $\rightarrow$  и  $\leftrightarrow$ . Если символам придавать значение И или Л, то с помощью таблиц истинности для связок можно вычислить соответствующее значение форму-

лы. Формула называется тождественно истинной, если при любых значениях входящих в неё символов она принимает значение И. Две формулы от одного и того же набора символов называются равносильными, если у них одинаковые таблицы истинности. Равносильность формул будем обозначать знаком  $\Leftrightarrow$ .

### Вопросы и задания

- 1 Из приведённых ниже предложений выберите высказывания и обоснуйте свой выбор:

а) «В тот год осенняя погода стояла долго на дворе» (А. Пушкин);  
 б) «Никто не может объять необъятное» (К. Прутков);  
 в) «Давайте восклицать, друг другом восхищаться» (Б. Окуджава);  
 г) «Не жалею, не зову, не плачу» (С. Есенин);  
 д) «Если у вас нет собаки, её не отравит сосед» (А. Аронов);  
 е) «Пусть всегда будет солнце!» (Л. Ошанин);  
 ж) «Я спросил у ясеня: «Где моя любимая?»» (В. Киршон);  
 з) «Скажи-ка, дядя, ведь недаром Москва, спалённая пожаром, французам отдана?» (М. Лермонтов);  
 и) «Никого не будет в доме, кроме сумерек» (Б. Пастернак);  
 к) «А где мне взять такую песню и о любви, и о судьбе?» (М. Агашина);  
 л) «Пусть бегут неуклюже пешеходы по лужам, а вода — по асфальту рекой» (А. Тимофеевский).

- 2 Составьте таблицы истинности для следующих формул:

а)  $X \wedge \neg Y \rightarrow Y \wedge Z$ ; б)  $(\neg X \wedge Y \rightarrow Z) \rightarrow ((Z \rightarrow \neg Y) \rightarrow X)$ .

- 3 Проверьте равносильность формул:

а)  $(\neg A \wedge B) \vee (\neg A \wedge \neg B \wedge C) \vee (B \wedge C)$  и  $(\neg A \vee B) \wedge C$ ;  
 б)  $\neg((\neg A \wedge B) \wedge (\neg B \vee C)) \vee \neg(C \wedge \neg A)$  и  $(A \vee B) \wedge (A \vee \neg B) \vee C$ .

- 4 а) Известно, что высказывание

$$(\neg A \vee B) \wedge (\neg B \vee A \rightarrow C) \wedge (\neg C \vee \neg A)$$

истинно. Какое наименьшее число ложных высказываний может быть среди высказываний  $A$ ,  $B$  и  $C$ ?

б) Известно, что высказывание

$$(\neg A \wedge \neg B) \vee (B \rightarrow \neg A \wedge C) \vee (\neg C \wedge A)$$

ложно. Какое наименьшее число истинных высказываний может быть среди высказываний  $A$ ,  $B$  и  $C$ ?

- 5 а) Укажите, какие из высказываний  $A$ ,  $B$  и  $C$  истинны, а какие ложны, если составленное из них высказывание

$$(\neg A \vee \neg B) \wedge (B \vee A \vee C) \rightarrow (\neg C \vee \neg A)$$

ложно.

б) Выполните задание а, если для высказываний  $A$ ,  $B$  и  $C$  истинно высказывание

$$(A \vee B) \wedge (\neg B \vee \neg A \vee C) \rightarrow (\neg C \vee \neg A) \leftrightarrow C.$$

- 6\* а) Укажите, какие из высказываний  $A$ ,  $B$ ,  $C$  и  $D$  истинны, а какие ложны, если известно, что истинно высказывание

$$B \wedge (D \vee (C \rightarrow A)) \leftrightarrow A,$$

а высказывание

$$(D \rightarrow C) \vee (A \wedge B)$$

ложно.

б) Выполните задание а, если для высказываний  $A$ ,  $B$ ,  $C$  и  $D$  истинны высказывания  $((D \vee C) \rightarrow \neg B) \wedge (D \vee B)$  и  $A \rightarrow (C \vee \neg B) \leftrightarrow B$ .

- 7 Постройте формулу  $F$ , имеющую таблицу истинности:

а)

| $X$ | $Y$ | $Z$ | $F$ |
|-----|-----|-----|-----|
| И   | И   | И   | И   |
| И   | И   | Л   | И   |
| И   | Л   | И   | Л   |
| И   | Л   | Л   | И   |
| Л   | И   | И   | Л   |
| Л   | И   | Л   | Л   |
| Л   | Л   | И   | И   |
| Л   | Л   | Л   | Л   |

б)

| $X$ | $Y$ | $Z$ | $F$ |
|-----|-----|-----|-----|
| И   | И   | И   | Л   |
| И   | И   | Л   | И   |
| И   | Л   | И   | И   |
| И   | Л   | Л   | Л   |
| Л   | И   | И   | Л   |
| Л   | И   | Л   | И   |
| Л   | Л   | И   | Л   |
| Л   | Л   | Л   | И   |

в)

| <i>A</i> | <i>B</i> | <i>C</i> | <i>D</i> | <i>F</i> |
|----------|----------|----------|----------|----------|
| И        | И        | И        | И        | И        |
| И        | И        | И        | Л        | Л        |
| И        | И        | Л        | И        | И        |
| И        | И        | Л        | Л        | Л        |
| И        | Л        | И        | И        | Л        |
| И        | Л        | И        | Л        | И        |
| И        | Л        | Л        | И        | Л        |
| И        | Л        | Л        | Л        | И        |
| Л        | И        | И        | И        | И        |
| Л        | И        | И        | Л        | И        |
| Л        | И        | Л        | И        | И        |
| Л        | И        | Л        | Л        | Л        |
| Л        | Л        | И        | И        | Л        |
| Л        | Л        | И        | Л        | И        |
| Л        | Л        | Л        | И        | Л        |
| Л        | Л        | Л        | Л        | И        |

- 8 **П** Последовательность высказываний  $A_n$  определена следующим образом:

$$A_n = (A_{n-1} \wedge A_{n-2}) \vee (A_{n-3} \rightarrow A_{n-2}).$$

Известно, что  $A_1$  — истинно,  $A_2$  — ложно,  $A_3$  — истинно. Какое значение имеет  $A_{2000}$ ?

- 9\* **П** Определите, существует ли формула  $F$ , такая, что ей тождественно истинна формула  $G$ :

а)  $G = (F \wedge Y \rightarrow \neg Z) \rightarrow (Z \rightarrow \neg Y)$ ;

б)  $G = (F \wedge Y \rightarrow \neg Z) \rightarrow ((Z \rightarrow \neg Y) \rightarrow F)$ .

## 1.2. Высказывательные формы (предикаты)

**Высказывательная форма (предикат)** — утверждение, содержащее одну переменную или более и становящееся высказыванием, когда переменным приданы конкретные значения. Предикаты, как и высказывания, можно соединять логическими связками. Предикаты мы будем обозна-

чать большими латинскими буквами, а фигурирующие в них переменные — малыми латинскими буквами (с индексами или без них).

Над предикатом может выполняться операция связывания переменной. Если  $P(x_1, x_2, \dots, x_n)$  — предикат, зависящий от переменных  $x_1, x_2, \dots, x_n$ , то утверждение «для любого  $x_1$   $P(x_1, x_2, \dots, x_n)$ » является предикатом от переменных  $x_2, \dots, x_n$ . Такой способ получения нового предиката называется связыванием переменной  $x_1$  квантором всеобщности. Аналогично о предикате «существует  $x_1$   $P(x_1, x_2, \dots, x_n)$ » говорят, что он получен из предиката  $P(x_1, x_2, \dots, x_n)$  связыванием переменной  $x_1$  квантором существования. В частности, если исходный предикат зависел только от одной переменной, то после её связывания получается высказывание. Переменные, не связанные никаким квантором, называются свободными.

Для обозначения кванторов применяют символы « $\forall$ » (квантор всеобщности) и « $\exists$ » (квантор существования).

### Вопросы и задания

- 1 Среди приведённых ниже предложений укажите те, которые являются предикатами. Для каждого из этих предикатов назовите свободные переменные, которые он имеет.
  - а) Окружность вписана в треугольник.
  - б) Ученик пришёл в школу.
  - в) Кому на Руси жить хорошо?
  - г) Книга будет напечатана в этом году.
  - д) Впишите треугольник в окружность.
  - е) Проект завершён.
  - ж) Число 0 делится на любое число.
  - з) Верно ли, что число делится на 5?
  - и) Для каждого натурального числа есть натуральное число, большее его.
  - к) Младшие уважают старших.
  - л) Переходить перекрёсток нужно только на зелёный свет светофора.
  - м) Не каждый ответит на вопрос: «Кому на Руси жить хорошо?»
- 2 Для каждого из приведённых ниже предикатов укажите свободные и связанные переменные, называя тип связывающего квантора:
  - а)  $\forall x \exists y (x + y = z \wedge z > 0 \vee (xz < y))$ ;
  - б)  $\forall x (x + y = z \wedge z > 0 \vee \exists y (xz < y))$ ;
  - в)  $\forall x (\exists z (x + y = z \wedge z > 0) \vee \exists y (xz < y))$ .

3. Пусть область допустимых значений переменных  $x, y$  и  $z$  — множество целых чисел. Вычислите значение предиката

$$\forall x (\exists y (x + y = z \wedge z > 0) \vee \exists z (xz < y \wedge y < 0)):$$

- а) при  $y = 1$  и  $z = 1$ ;      б) при  $y = 1$  и  $z = -1$ ;  
в) при  $y = -1$  и  $z = 2$ ;    г) при  $y = -1$  и  $z = -1$ .

4. Пусть область допустимых значений переменных  $x, y$  и  $z$  — множество целых чисел. Найдите все значения переменной  $z$ , для которых предикат  $\forall x \exists y (x + y = z \wedge xz > y)$  принимает значение Истина.

5. Рассмотрим предикат  $D(x, y)$ : «число  $x$  делится нацело на число  $y$ ».

а) Пусть этот предикат рассматривается на множестве всех натуральных чисел (т. е. всех целых положительных чисел). Запишите каждое из высказываний  $\forall x \forall y (D(x, y))$ ;  $\forall x \exists y (D(x, y))$ ;  $\exists x \forall y (D(x, y))$ ;  $\exists x \exists y (D(x, y))$  предложением русского языка. Определите, какие из этих высказываний истинны, а какие ложны.

б) Выполните задание пункта а, если предикат  $D(x, y)$  рассматривается на множестве всех целых чисел.

в) Выполните задание пункта а, если предикат  $D(x, y)$  рассматривается на множестве всех положительных рациональных чисел.

6. Пусть  $P(x, y, z)$  — предикат от трёх переменных  $x, y, z$ , означающий, что точка  $y$  лежит между точками  $x$  и  $z$ . Существует восемь вариантов связывания переменных  $x, y, z$  (в указанном порядке) кванторами всеобщности и существования. Вот три из этих вариантов:  $\exists x \forall y \forall z P(x, y, z)$ ;  $\forall x \exists y \forall z P(x, y, z)$ ;  $\forall x \forall y \exists z P(x, y, z)$ .

а) Для приведённых выше вариантов укажите, какие из них являются истинными высказываниями.

б) Запишите остальные пять возможных вариантов связывания переменных и для каждого из них определите, истинное или ложное высказывание получается.

в) Если порядок связывания переменных изменять, то можно получить ещё 40 записанных по-разному высказываний. Вот пять из них:  $\exists y \forall x \forall z P(x, y, z)$ ;  $\forall x \exists z \forall y P(x, y, z)$ ;  $\exists y \forall x \exists z P(x, y, z)$ ;  $\exists y \exists x \forall z P(x, y, z)$ ;  $\forall y \exists x \exists z P(x, y, z)$ . Укажите, какие из этих высказываний истинны.

7. Пусть  $P(x, y)$  — произвольный предикат от свободных переменных  $x$  и  $y$ . Верно ли, что для любого предиката  $P(x, y)$  следующие два высказывания одновременно истинны или одновременно ложны:

- а)  $\forall x \forall y P(x, y)$  и  $\forall y \forall x P(x, y)$ ;  
б)  $\exists x \exists y P(x, y)$  и  $\exists y \exists x P(x, y)$ ;  
в)  $\exists x \forall y P(x, y)$  и  $\forall y \exists x P(x, y)$ ?

- 8 Рассмотрите предикаты, заданные на множестве натуральных чисел:
- $x$  — нечётное число, и для любого нечётного числа  $y$  выполнено неравенство  $x \leq y$ ;
  - $x$  — простое число, и для любого простого числа  $y$  выполнено неравенство  $x \leq y$ .
- Для каждого из этих предикатов укажите все те значения аргумента  $x$ , для которого данный предикат истинен.
- 9 П На множестве натуральных чисел рассматривается предикат  $\exists x \forall y (y > zx^2 - 3x)$ . Укажите все значения переменной  $z$ , для которых данный предикат истинен.
- 10 П На множестве целых неотрицательных чисел задан предикат  $S(x, y, z)$  — число  $z$  равно сумме чисел  $x$  и  $y$ . Используя этот предикат, запишите подходящей формулой каждое из следующих утверждений:
- число  $x$  равно 0;
  - число  $x$  чётно;
  - число  $x$  не больше числа  $y$ ;
  - числа  $x$  и  $y$  равны;
  - число  $x$  равно 1;
  - число  $x$  кратно 3.
- 11 П На множестве натуральных чисел задан предикат  $S(x, y, z)$  — число  $z$  равно сумме чисел  $x$  и  $y$ . Используя этот предикат, запишите подходящей формулой каждое из следующих утверждений:
- число  $x$  чётно;
  - число  $x$  кратно 3;
  - число  $x$  меньше числа  $y$ ;
  - числа  $x$  и  $y$  равны;
  - число  $x$  равно 1;
  - число  $y$  равно  $2x + 1$ .
- 12 П На множестве натуральных чисел задан предикат  $S(x, y, z)$  — число  $z$  равно сумме чисел  $x$  и  $y$ . Используя этот предикат, запишем формулу, содержащую некоторый предикат  $P(x, y)$ :
- $$P(x, y) \leftrightarrow (\forall u \forall w \neg S(u, w, x)) \& (\forall u \forall w \neg S(u, w, y)) \vee (\exists u \exists w (S(w, 1, x) \wedge P(w, u) \wedge S(x, u, y))).$$
- Будем считать, что эта формула тождественно истинна.
- Найдите значения  $P(1, 1)$ ;  $P(2, 2)$ ;  $P(2, 3)$ ;  $P(4, 3)$ ;  $P(5, 15)$ .
  - Пусть  $x = 3$ . Укажите те значения переменной  $y$ , для которых предикат  $P(x, y)$  имеет значение Истина.
  - Пусть  $y = 15$ . Укажите те значения переменной  $x$ , для которых предикат  $P(x, y)$  имеет значение Истина.
  - Пусть  $y = 13$ . Найдётся ли такое значение переменной  $x$ , для которого предикат  $P(x, y)$  имеет значение Истина?
  - \* Докажите, что значение  $P(x, y)$  действительно однозначно определено при любых заданных значениях переменных  $x$  и  $y$ .
  - \* Докажите, что предикат  $P(x, y)$  имеет значение Истина тогда и только тогда, когда  $y = \frac{x(x+1)}{2}$ .

- 13 П На множестве натуральных чисел задан предикат  $S(x, y, z)$  — число  $z$  равно сумме чисел  $x$  и  $y$ . Выполнив задание 11, вы с помощью этого предиката записали предикаты «число  $x$  равно 1» и «число  $y$  равно  $2x + 1$ ». Обозначим эти предикаты для краткости как  $One(x)$  и  $L(x, y)$  соответственно. Используя эти предикаты, запишем формулу, содержащую некоторый предикат  $P(x, y)$ :

$$P(x, y) \leftrightarrow (One(x) \wedge One(y)) \vee (\exists u \exists w \exists z (S(w, 1, x) \wedge L(w, z) \wedge P(w, u) \wedge S(u, w, y))).$$

Будем считать, что эта формула тождественно истинна.

а) Найдите значения  $P(1, 1)$ ;  $P(2, 4)$ ;  $P(3, 6)$ ;  $P(4, 2)$ ;  $P(5, 25)$ .

б) Пусть  $x = 3$ . Укажите те значения переменной  $y$ , для которых предикат  $P(x, y)$  имеет значение Истина.

в) Пусть  $y = 16$ . Укажите те значения переменной  $x$ , для которых предикат  $P(x, y)$  имеет значение Истина.

г) Пусть  $y = 13$ . Найдётся ли такое значение переменной  $x$ , для которого предикат  $P(x, y)$  имеет значение Истина?

д)\* Докажите, что значение  $P(x, y)$  действительно однозначно определено при любых заданных значениях переменных  $x$  и  $y$ .

е)\* Докажите, что предикат  $P(x, y)$  имеет значение Истина тогда и только тогда, когда  $y = x^2$ .

- 14 П На множестве натуральных чисел задан предикат  $S(x, y, z)$  — число  $z$  равно сумме чисел  $x$  и  $y$ . Выполнив задание 11, вы с помощью этого предиката записали предикат «числа  $x$  и  $y$  равны». Для краткости обозначим его через  $R(x, y)$ . Используя предикаты  $S$  и  $R$ , запишем формулу, содержащую некоторый предикат  $U(x, y, z)$ :

$$U(x, y, z) \leftrightarrow (R(x, z) \wedge R(y, 1)) \vee (\exists v \exists w (S(v, 1, y) \wedge U(x, v, w) \wedge S(x, w, z))).$$

Будем считать, что эта формула тождественно истинна.

а) Найдите значения  $U(1, 1, 1)$ ;  $U(2, 2, 2)$ ;  $U(2, 3, 6)$ .

б) Пусть  $x = 3$ ,  $y = 5$ . Укажите те значения переменной  $z$ , для которых предикат  $U(x, y, z)$  имеет значение Истина.

в) Пусть  $x = 5$ ,  $z = 30$ . Укажите те значения переменной  $y$ , для которых предикат  $U(x, y, z)$  имеет значение Истина.

г)\* Докажите, что значение  $U(x, y, z)$  однозначно определено при любых заданных значениях переменных  $x$ ,  $y$  и  $z$ .

д)\* Докажите, что предикат  $U(x, y, z)$  имеет значение Истина тогда и только тогда, когда  $z = xy$ .

- 15 П На множестве натуральных чисел задан предикат  $S(x, y, z)$  — число  $z$  равно сумме чисел  $x$  и  $y$ . Выполнив задание 11, вы с помощью этого предиката записали предикат «числа  $x$  и  $y$  равны». Обозначим этот предикат для краткости как  $R(x, y)$ . Используя предикаты  $S(x, y, z)$  и

$R(x, y)$ , запишем формулу, содержащую некоторый предикат  $P(x, y, z)$ :

$$P(x, y, z) \leftrightarrow (R(x, y) \wedge R(x, z)) \vee (\exists u (S(z, u, x) \wedge P(u, y, z))) \vee (\exists u (S(x, u, y) \wedge P(x, u, z))).$$

Будем считать, что эта формула тождественно истинна.

а) Найдите значения  $P(3, 3, 3)$ ;  $P(5, 2, 6)$ ;  $P(4, 4, 8)$ ;  $P(6, 9, 3)$ ;  $P(6, 9, 2)$ .

б) Пусть  $x = 18$ ,  $y = 12$ . Укажите те значения переменной  $z$ , для которых предикат  $P(x, y, z)$  имеет значение Истина.

в) Пусть  $x = 15$ ,  $z = 5$ . Укажите все значения переменной  $y$ , для которых предикат  $P(x, y, z)$  имеет значение Истина.

г)\* Докажите, что значение  $P(x, y, z)$  однозначно определено при любых заданных значениях переменных  $x$ ,  $y$  и  $z$ .

д)\* Докажите, что при любых заданных значениях переменных  $x$  и  $y$  существует, и притом единственное, значение переменной  $z$ , для которого предикат  $P(x, y, z)$  имеет значение Истина.

- 16 **П** На множестве натуральных чисел задан предикат  $S(x, y, z)$  — число  $z$  равно сумме чисел  $x$  и  $y$ . Выполнив задание 11, вы с помощью этого предиката записали предикат «числа  $x$  и  $y$  равны». Обозначим этот предикат для краткости как  $R(x, y)$ . Используя эти предикаты, запишем формулу, содержащую некоторый предикат  $P(x, y)$ :

$$P(x, y) \leftrightarrow ((R(x, 1) \vee (R(x, 2) \wedge R(x, y))) \vee (\exists u \exists w \exists t \exists s (S(t, 1, x) \wedge S(s, 2, x) \wedge P(s, u) \wedge P(t, w) \wedge S(u, w, y))).$$

Будем считать, что эта формула тождественно истинна.

а) Найдите значения предиката  $P$  при  $x = y = 2$ ; при  $x = y = 3$ ; при  $x = 3$  и  $y = 5$ ;

б) Пусть  $x = 4$ . Укажите те значения переменной  $y$ , для которых предикат  $P(x, y)$  имеет значение Истина.

в) Пусть  $y = 13$ . Укажите те значения переменной  $x$ , для которых предикат  $P(x, y)$  имеет значение Истина.

г) Пусть  $y = 15$ . Найдётся ли такое значение переменной  $x$ , для которого предикат  $P(x, y)$  имеет значение Истина?

д)\* Докажите, что значение  $P(x, y)$  действительно однозначно определено при любых заданных значениях переменных  $x$  и  $y$ .

е)\* Докажите, что при любом заданном значении переменной  $x$  существует, и притом единственное, значение переменной  $y$ , для которого предикат  $P(x, y)$  имеет значение Истина.

Релейно-контактная схема — это электрическая схема, содержащая один или несколько переключателей, позволяющих управлять прохождением тока через эту схему. Два переключателя можно соединить либо последовательно (рис. 22.1, а), либо параллельно (рис. 22.1, б).

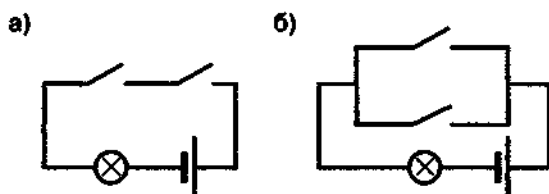


Рис. 22.1

Для математического описания релейно-контактных схем и изучения их свойств используются методы математической логики. Каждый переключатель обозначается как переменная, принимающая значение 0 (Ложь) или 1 (Истина), параллельное соединение двух переключателей соответствует операции ИЛИ, обозначаемой символом  $\vee$ , последовательное соединение — операции И, обозначаемой символом  $\wedge$ . Один и тот же переключатель может фигурировать в разных частях электронной схемы, в этом случае он обозначается одной и той же буквой. Если некий контакт разомкнут тогда и только тогда, когда контакт  $X$  замкнут, то такой контакт обозначается как  $\neg X$ . Тем самым каждой схеме можно сопоставить логическое выражение, которое принимает значение 1 при заданных значениях переменных тогда и только тогда, когда при соответствующем положении переключателей через эту схему идёт электрический ток. Верно и обратное утверждение: по логическому выражению можно построить последовательно-параллельную схему, для которой выполнено указанное свойство.

### Вопросы и задания

- 1 Почему в электронной вычислительной технике обычно используется двухсимвольное кодирование?
- 2 Для каждой из схем, изображённых на рисунке 22.2, запишите соответствующее ей логическое выражение.

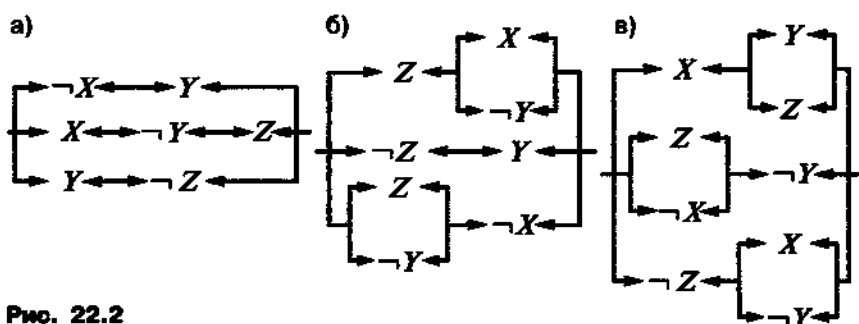


Рис. 22.2

- 3 Для каждого из приведённых ниже логических выражений нарисуйте соответствующую ему последовательно-параллельную схему:

- $(X \vee Z) \wedge (Y \vee \neg X) \wedge (\neg Y \vee Z)$ ;
- $\neg Y \vee (X \wedge \neg Z) \vee (\neg X \wedge Y \wedge Z)$ ;
- $((X \vee \neg Y) \wedge Z) \vee ((Y \vee \neg Z) \wedge X) \vee ((Z \vee \neg X) \wedge Y)$ ;
- $\neg(X \wedge Z) \wedge (\neg Y \vee X) \wedge (Y \vee \neg Z)$ ;
- $(X \wedge \neg Y \vee Z) \wedge \neg((Y \vee \neg X) \wedge Z)$ ;
- $((Z \vee X) \wedge \neg Y) \vee \neg X \vee \neg(Y \vee Z)$ .

- Напишите какое-либо логическое выражение, содержащее три переменные, и предложите соседу по парте нарисовать соответствующую этому выражению последовательно-параллельную схему. Проверьте, правильно ли он выполнил задание.
- Для схем, изображённых на рисунке 22.3, постройте эквивалентные им более простые схемы.
- Для каждой мостовой схемы, изображённой на рисунке 22.4, постройте эквивалентную ей последовательно-параллельную схему.
- а) Известно, что высказывание  $F$ , зависящее от трёх вы-

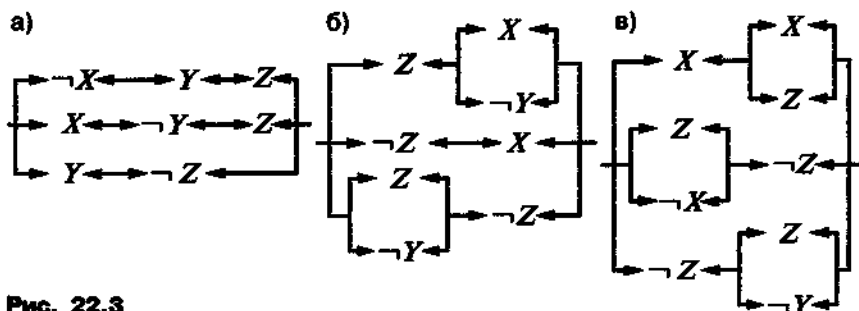


Рис. 22.3

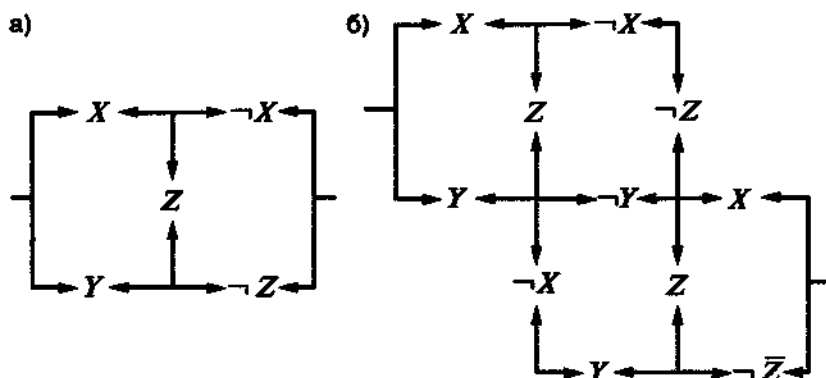


Рис. 22.4

сказываний  $A$ ,  $B$  и  $C$ , принимает то значение для каждого данного набора значений  $A$ ,  $B$  и  $C$ , которое принимает большинство из этих трёх высказываний. Функцию  $F$  поэтому называют **функцией голосования**. Составьте для  $F$  логическую формулу.

б) Составьте релейно-контактную схему, реализующую функцию голосования для трёх участников.

19

23

## II Логические элементы. Вентили

**Логическим элементом или вентиляем** называют устройство с  $n$  входами и одним выходом, которое преобразует входные двоично закодированные сигналы в двоичный сигнал на выходе. Каждый вентиль является, по существу, некоторой релейно-контактной схемой; входы вентиля — это электронно управляемые переключатели такой схемы.

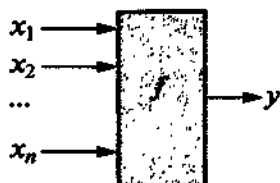


Рис. 23.1

Условно логический элемент  $f$  можно изобразить так, как показано на рисунке 23.1. Поскольку внутреннее устройство вентиля нас не интересует, можно рассматривать его как некоторый чёрный ящик.

Различных логических элементов с  $n$  входами существует  $2^{2^n}$ . В таблице указано, как обрабатывает входной сигнал каждый из четырёх логических элементов с одним входом.

## Логические элементы с одним входом

| Входной<br>сигнал $x$ | Сигнал на выходе |     |          |     |
|-----------------------|------------------|-----|----------|-----|
|                       | $\lambda$        | $x$ | $\neg x$ | $0$ |
| 0                     | 1                | 0   | 1        | 0   |
| 1                     | 1                | 1   | 0        | 0   |

Первый из этих элементов называется тождественной единицей, второй — просто тождественным, третий — отрицанием, или вентилем НЕ, четвёртый — тождественным нулём.

В таблице указано, как обрабатывает пару входных сигналов каждый из 16 логических элементов с двумя входами.

## Логические элементы с двумя входами

| Вход-<br>ной<br>сигнал<br>$x_1 x_2$ | Сигнал на выходе |                |                    |                    |                |       |       |                  |
|-------------------------------------|------------------|----------------|--------------------|--------------------|----------------|-------|-------|------------------|
|                                     | $\lambda$        | $x_1 \vee x_2$ | $x_1 \uparrow x_2$ | $x_2 \uparrow x_1$ | $x_1 \mid x_2$ | $x_1$ | $x_2$ | $x_1 \oplus x_2$ |
| 00                                  | 1                | 0              | 1                  | 1                  | 1              | 0     | 0     | 0                |
| 01                                  | 1                | 1              | 0                  | 1                  | 1              | 0     | 1     | 1                |
| 10                                  | 1                | 1              | 1                  | 0                  | 1              | 1     | 0     | 1                |
| 11                                  | 1                | 1              | 1                  | 1                  | 0              | 1     | 1     | 0                |

| Вход-<br>ной<br>сигнал<br>$x_1 x_2$ | Сигнал на выходе |            |            |              |                |                    |                    |     |
|-------------------------------------|------------------|------------|------------|--------------|----------------|--------------------|--------------------|-----|
|                                     | $x_1 - x_2$      | $\neg x_2$ | $\neg x_1$ | $x_1 \& x_2$ | $x_1 \neq x_2$ | $x_2 \uparrow x_1$ | $x_1 \uparrow x_2$ | $0$ |
| 00                                  | 1                | 1          | 1          | 0            | 0              | 0                  | 1                  | 0   |
| 01                                  | 0                | 0          | 1          | 0            | 0              | 1                  | 0                  | 0   |
| 10                                  | 0                | 1          | 0          | 0            | 1              | 0                  | 0                  | 0   |
| 11                                  | 1                | 0          | 0          | 1            | 0              | 0                  | 0                  | 0   |

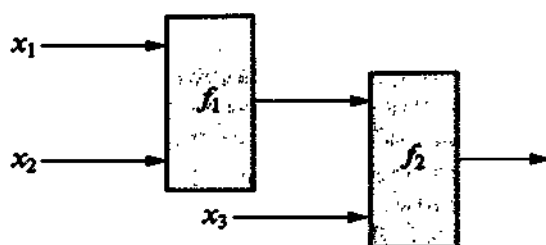


Рис. 23.2. Комбинация  $f_2(f_1(x_1, x_2), x_3)$  элементов  $f_1$  и  $f_2$

Приведём общепринятые названия некоторых из этих логических элементов (не повторяя тех, которые названы ранее).

Элемент  $\&$  называется конъюнкцией. Другое название для него — вентиль И.

Элемент  $\vee$  называется дизъюнкцией. Другое название для него — вентиль ИЛИ.

Элемент  $\rightarrow$  называется импликацией.

Элемент  $\sim$  называется эквиваленцией.

Элемент  $\oplus$  называется сложением по модулю 2.

Элемент  $|$  называется операцией Шеффера.

Элемент  $\uparrow$  называется операцией Пирса.

Логические элементы можно комбинировать друг с дру-

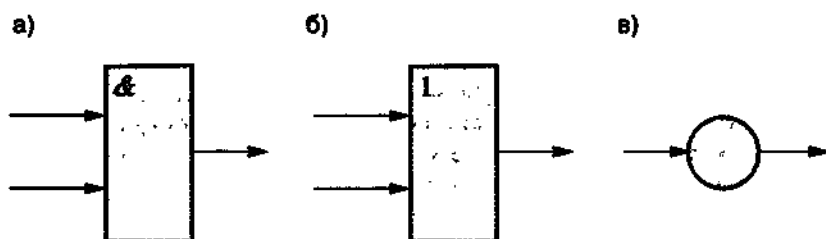


Рис. 23.3. Обозначения вентилях И, ИЛИ, НЕ

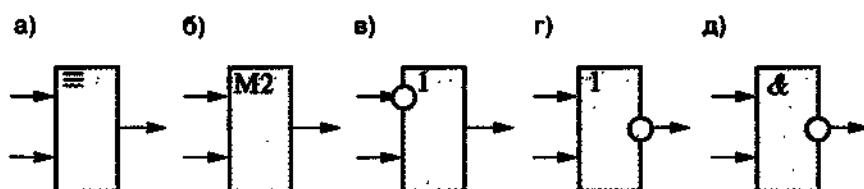


Рис. 23.4. Стандартные изображения вентилях: а) эквиваленция; б) сравнение по модулю 2; в) импликация; г) операция Пирса; д) операция Шеффера

гом, подавая на входы одного элемента то, что получилось на выходах других элементов. Пример такой комбинации приведён на рисунке 23.2.

Получившееся устройство естественно рассматривать как логический элемент с тремя входами  $x_1, x_2, x_3$ .

Оказывается, что, комбинируя вентили НЕ, И, ИЛИ, можно собрать схему, равносильную любому логическому элементу от любого числа переменных. Для удобства эти вентили в схемах обозначают так, как показано на рисунке 23.3. Обозначения других наиболее употребительных вентилей с двумя входами приведены на рисунке 23.4.

### Вопросы и задания

- 1 а) Составьте формулу, описывающую схему, изображённую на рисунке 23.5.
- б) Составьте таблицу значений логической функции, реализованной схемой, изображённой на рисунке 23.5.
- в) Какая функция из перечисленных в таблице на с. 144 эквивалентна логической функции, реализованной данной схемой?

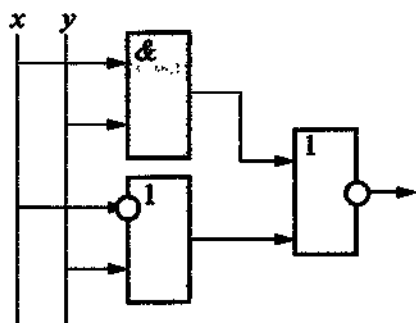


Рис. 23.5

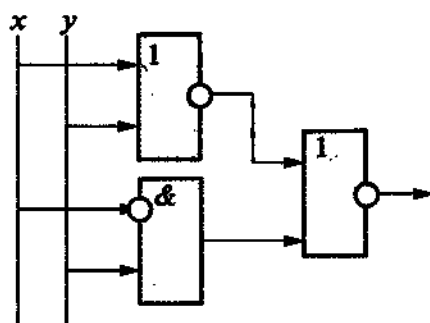


Рис. 23.6

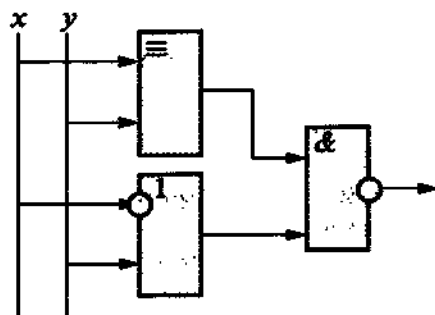


Рис. 23.7

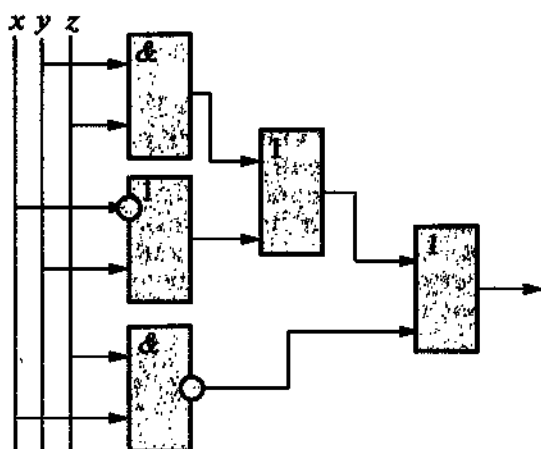


Рис. 23.8

- 2 Выполните задание 1 для схемы, изображённой на рисунке 23.6.
- 3 Выполните задание 1 для схемы, изображённой на рисунке 23.7.
- 4 а) Составьте формулу, описывающую схему, изображённую на рисунке 23.8.  
 б) Упростите получившуюся формулу.  
 в) Составьте вентильную схему для получившейся формулы.

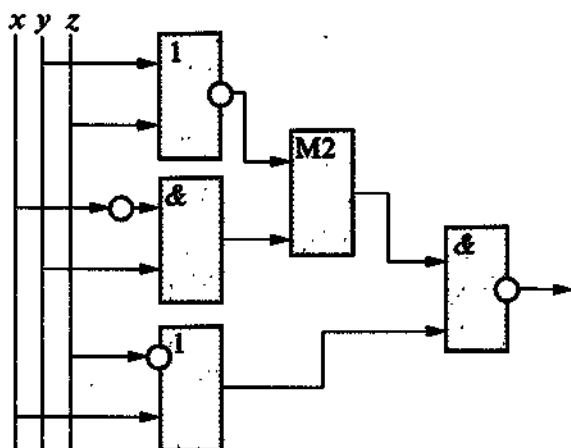


Рис. 23.9

- 5 Выполните задание 4 для схемы, изображённой на рисунке 23.9.
- 6 Объясните, почему из вентилях И, ИЛИ, НЕ можно собрать схему, эквивалентную любому логическому элементу. (Совет: воспользуйтесь сведениями из п. 21.1.)
- 7 а) Можно ли из вентилях И и НЕ построить схему, эквивалентную вентилю ИЛИ? Ответ «да» подтвердите, указав такую схему, ответ «нет» обоснуйте.  
б)\* Можно ли из вентилях И и ИЛИ построить схему, эквивалентную вентилю НЕ? Ответ «да» подтвердите, указав такую схему, ответ «нет» обоснуйте.
- 8 а) Составьте для логической функции  $f = (x \rightarrow y) \rightarrow (y \rightarrow x)$  таблицу значений. Какая функция из перечисленных в таблице на с. 144 эквивалентна функции  $f$ ?  
б) Выполните задание а для логической функции  $f = (x \mid y) \mid (y \mid x)$ .  
в) Выполните задание а для логической функции  $f = (x \uparrow y) \uparrow (y \uparrow x)$ .
- 9 а) Известно, что, используя только операцию Шеффера, можно построить схему, эквивалентную вентилю И. Нарисуйте схему, подтверждающую это утверждение.  
б) Выполните задание, аналогичное заданию а, для вентилях ИЛИ и НЕ.  
Вместе с утверждением, сформулированным в задании б, полученные вами результаты показывают, что *при конструировании логических элементов можно обойтись только операцией Шеффера.*
- 10 Выполните для операции Пирса такие же задания, как 9а и 9б. Какой вывод можно сделать?
- 11\* Исследуйте, существуют ли такие логические элементы с двумя входами, отличные от операции Шеффера и операции Пирса, которые, подобно этим элементам, «в одиночку» могли бы реализовать любой логический элемент.
- 12\* а) Постройте схему сумматора, вычисляющего сумму трёх однозначных чисел в двоичной системе. (Совет: составьте сначала таблицу значений младшего и старшего разрядов для суммы трёх однозначных чисел.)  
б) При сложении многозначных чисел в старших разрядах приходится учитывать появление так называемой единицы переноса. Это означает, что в этих разрядах складываются не два однозначных числа, а три. Используя результат пункта а, построьте схему сумматора для сложения двух двузначных чисел в двоичной системе счисления.

## Ответы и результаты

§ 1. 5. б), д), е).

**§ 2. 4. 8.**

§ 3.1. 7. 6) 2832. 11. в) 7. 17. а)  $x = 6$ ,  $y = 5$ . 20. 6) 14.

**§ 3.2. 5. 120. 11. а) 4096 и 32767.**

§ 4. 8. г) коричневато-оранжевый (кирпичный). 9. г) сиреневый (или фиолетовый).

**§ 5. 1. 6) 3.**

**§ 6.1. 15. a) 533. 6)**

[illegible]

(для большей наглядности нули заменены точками).

**19. 7 c.**

**§ 6.2. 10. 6)  $\approx 7,88, 13, 00$ .**

**§ 8. 6. в) 0.**

**§ 9. 4. а) Население < 1000 И Блок = ЕЭС** (если в ответе на этот запрос будет указано хотя бы одно государство, то ответ на вопрос, сформулированный в пункте а, отрицательный).

**§ 10. 9. а) влево; вниз; вправо. 10. б) 9. 12. в) 10.**

**§ 11. 5. 15; F8.**

§ 12.1. 4. а)  $12,5\pi + 24$ . 7. а)  $X_2 = -7$ ;  $X_4 = 27$ ;  $X_7 = -1207$ .

|        |                  |            |             |               |
|--------|------------------|------------|-------------|---------------|
| 11. 6) | <b>n</b>         | <b>637</b> | <b>4937</b> | <b>51 983</b> |
|        | <b>Результат</b> | <b>7</b>   | <b>4937</b> | <b>1091</b>   |

|        |           |    |      |        |
|--------|-----------|----|------|--------|
| 12. 6) | <i>n</i>  | 20 | 300  | 7000   |
|        | Результат | 71 | 1987 | 70 657 |

16. 6)

|                  |        |        |
|------------------|--------|--------|
| $a$              | 6      | 0      |
| $b$              | 10     | 5      |
| Координаты точки | (5; 3) | (0; 0) |

17. 6) максимальное значение 420.

§ 12.2. 4. а) 6; в) 12. 5. 6)  $8 + 18k$ , где  $k$  — любое целое неотрицательное число; в) таких  $n$  не существует. 6. а) 2.

§ 12.3. 1. в) 2, 4, 6; г) -8, -7, -6, -5, -4, -2, -3, -2, -1, 2, 3.

§ 13.2. 1. в) 143.

§ 13.3. 2. а) 9. 3.

|     |   |   |   |    |   |    |    |          |
|-----|---|---|---|----|---|----|----|----------|
| $m$ | 1 | 2 | 3 | 4  | 5 | 6  | 7  | $\geq 8$ |
| $s$ | 2 | 7 | 9 | 26 | 8 | 18 | 26 | 36       |

2. а) 2. 3. 0 при  $m < 5$ , 6 при  $m \geq 5$ . То же самое можно записать одной формулой  $3 + 3 |m - 4,5| / (m - 4,5)$ .

§ 13.4. 4. Третье число Кармайкла: 1729.

§ 14. 1. а) 6; б) 10; в) 24. 3. а) 2, 5, 7, 16, 33, 65. 3. а) -1, 0, 3, -12, 1, 5.

§ 15.1. 6. ж)

|          |          |          |
|----------|----------|----------|
| $A$      | $C$      | $E$      |
| $\vdots$ | $\vdots$ | $\vdots$ |
| $B$      | $D$      | $F$      |

§ 15.2. 1. в) СВВФ.

§ 16. 10. 6) Выигрышной стратегии нет ни у одного из игроков. Лучший результат, которого может добиться каждый из игроков, — ничья.

§ 18. 2. д) Может. Вид последней амёбы всегда второй, независимо от того, как будут выбираться амёбы при исполнении тела цикла. 3. а) массовый,  $N$  — любое целое число; б) не массовый; в) не массовый, единственное допустимое значение  $x$  — число 0, для остальных значений  $x$  произойдёт остановка с ошибкой «извлечение корня из отрицательного числа»; г) массовый,  $A$  — любое неотрицательное число. 10. а) 11; б) ни разу.

§ 21.1. 4. а)  $A = И$ ,  $B = Л$ ,  $C = И$ ; б)  $A = Л$ ,  $B = И$ ,  $C = И$ .

§ 21.2. 10. 6)  $\exists y S(y, y, x)$ ; в)  $\exists z S(x, z, y)$ .

11. 6)  $\exists y \exists z S(y, y, z) \& S(y, z, x)$ ; д)  $\forall y \forall z \neg S(y, z, x)$ .

15. а)  $P(3, 3, 3) = И$ ;  $P(5, 2, 6) = Л$ ;  $P(4, 4, 8) = Л$ ;  
 $P(6, 9, 3) = И$ ;  $P(6, 9, 2) = Л$ ; б)  $z = 6$ ; е)  $z = \text{НОД}(x, y)$ .

§ 22. 2. а)  $(\neg X \& Y) \vee (X \& \neg Y \& Z) \vee (Y \& \neg Z)$ .

3. а)

|              |              |              |
|--------------|--------------|--------------|
| $X$          | $Y$          | $\neg Y$     |
| $\leftarrow$ | $\leftarrow$ | $\leftarrow$ |
| $Z$          | $\neg X$     | $Z$          |

§ 23. 1. а)  $(X \& Y) \uparrow (X \rightarrow Y)$ ; в)  $X \nrightarrow Y$ .

## Основная литература

Гейн А. Г. Информатика и ИКТ: учеб. для 10 кл. общеобразоват. учреждений: базовый и профил. уровни / А. Г. Гейн, А. Б. Ливчак, А. И. Сенокосов, Н. А. Юерман. — М.: Просвещение, 2008. — 272 с.

Гейн А. Г. Информатика и ИКТ: учеб. для 10 кл. общеобразоват. учреждений: базовый и профил. уровни / А. Г. Гейн, А. И. Сенокосов. — М.: Просвещение, 2009. — 336 с.

Информатика. Задачник-практикум. В 2 т. / под ред. И. Г. Семакина, Е. К. Хеннера. Т. 1. — М.: ЛБЗ, 1999.

## Дополнительная литература

Алексеев А. В. Олимпиады школьников по информатике / А. В. Алексеев. — Красноярск: Кн. изд-во, 1995. — 224 с.

Андреева Е. В. Математические основы информатики. Элективный курс: учеб. пособие / Е. В. Андреева, Л. Л. Босова, И. Н. Фалина. — М.: БИНОМ. Лаборатория знаний, 2005. — 328 с.

Андреева Е. В. Математические основы информатики. Элективный курс: метод. пособие / Е. В. Андреева, Л. Л. Босова, И. Н. Фалина. — М.: БИНОМ. Лаборатория знаний, 2007. — 312 с.

Андреева Е. В. Системы счисления и компьютерная арифметика / Е. В. Андреева, И. Н. Фалина. — М.: ЛБЗ, 2000. — 248 с.

Гейн А. Г. Информатика и информационные технологии: кн. для учителя: метод. рекомендации к учеб. 10 кл. / А. Г. Гейн. — М.: Просвещение, 2008. — 160 с.

Гейн А. Г. Информатика и ИКТ: кн. для учителя: 11 класс / А. Г. Гейн, Н. А. Юерман, А. А. Гейн. — М.: Просвещение, 2009. — 240 с.

Ефимова О. А. Практикум по компьютерной технологии: метод. пособие / О. А. Ефимова, М. В. Моисеева, Ю. А. Шафрин. — М.: АБФ, 1997. — 560 с.

Касаткин В. Н. Информация, алгоритмы, ЭВМ: пособие для учителя / В. Н. Касаткин. — М.: Просвещение, 1991. — 192 с.

Кириухин В. М. Информатика. Всероссийские олимпиады (Серия «Пять колец») / В. М. Кириухин. — М.: Просвещение, 2009. — 182 с.

Кольман Э. Занимательная логика / Э. Кольман, О. Зих. — М.: Наука, 1966. — 127 с.

Матвеева Т. А. Информационная культура: учеб. пособие. Ч. 1 / Т. А. Матвеева, А. Г. Гейн, В. В. Мачульский и др. — Смоленск: Ассоциация XXI век, 2006. — 392 с.

Матвеева Т. А. Информационная культура: учеб. пособие. Ч. 2 / Т. А. Матвеева, А. Г. Гейн, В. В. Мачульский и др. — Смоленск: Ассоциация XXI век, 2007. — 416 с.

Окулов С. М. Задачи по программированию / С. М. Окулов, Т. В. Ашихмин, Н. А. Бушмелёва и др.: под ред. С. М. Окулова. — М.: БИНОМ. Лаборатория знаний, 2006. — 820 с.

Шауцукова Л. З. Информатика: 10—11 кл.: учеб. пособие для 10—11 кл. / Л. З. Шауцукова. — М.: Просвещение, 2004. — 143 с.

Шафрин Ю. А. Информационные технологии / Ю. А. Шафрин. — М.: Лаборатория Базовых Знаний, 1998. — 704 с.

# Приложения

Таблица II.1

## Основная часть ASCII

| Двоичный код | Десятичный код | Символ | Двоичный код | Десятичный код | Символ |
|--------------|----------------|--------|--------------|----------------|--------|
| 00100000     | 32             | Пробел | 00110111     | 55             | 7      |
| 00100001     | 33             | !      | 00111000     | 56             | 8      |
| 00100010     | 34             | #      | 00111001     | 57             | 9      |
| 00100011     | 35             | \$     | 00111010     | 58             | :      |
| 00100100     | 36             | *      | 00111011     | 59             | ;      |
| 00100101     | 37             | %      | 00111100     | 60             | <      |
| 00100110     | 38             | &      | 00111101     | 61             | =      |
| 00100111     | 39             | '      | 00111110     | 62             | >      |
| 00101000     | 40             | (      | 00111111     | 63             | ?      |
| 00101001     | 41             | )      | 01000000     | 64             | @      |
| 00101010     | 42             | *      | 01000001     | 65             | A      |
| 00101011     | 43             | +      | 01000010     | 66             | B      |
| 00101100     | 44             | ,      | 01000011     | 67             | C      |
| 00101101     | 45             | -      | 01000100     | 68             | D      |
| 00101110     | 46             | .      | 01000101     | 69             | E      |
| 00101111     | 47             | /      | 01000110     | 70             | F      |
| 00110000     | 48             | 0      | 01000111     | 71             | G      |
| 00110001     | 49             | 1      | 01001000     | 72             | H      |
| 00110010     | 50             | 2      | 01001001     | 73             | I      |
| 00110011     | 51             | 3      | 01001010     | 74             | J      |
| 00110100     | 52             | 4      | 01001011     | 75             | K      |
| 00110101     | 53             | 5      | 01001100     | 76             | L      |
| 00110110     | 54             | 6      | 01001101     | 77             | M      |

## Продолжение

| Двоичный код | Десятичный код | Символ | Двоичный код | Десятичный код | Символ |
|--------------|----------------|--------|--------------|----------------|--------|
| 01001110     | 78             | N      | 01100111     | 103            | g      |
| 01001111     | 79             | O      | 01101000     | 104            | h      |
| 01010000     | 80             | P      | 01101001     | 105            | i      |
| 01010001     | 81             | Q      | 01101010     | 106            | j      |
| 01010010     | 82             | R      | 01101011     | 107            | k      |
| 01010011     | 83             | S      | 01101100     | 108            | l      |
| 01010100     | 84             | T      | 01101101     | 109            | m      |
| 01010101     | 85             | U      | 01101110     | 110            | n      |
| 01010110     | 86             | V      | 01101111     | 111            | o      |
| 01010111     | 87             | W      | 01110000     | 112            | p      |
| 01011000     | 88             | X      | 01110001     | 113            | q      |
| 01011001     | 89             | Y      | 01110010     | 114            | r      |
| 01011010     | 90             | Z      | 01110011     | 115            | s      |
| 01011011     | 91             | ]      | 01110100     | 116            | t      |
| 01011100     | 92             | \      | 01110101     | 117            | u      |
| 01011101     | 93             | ]      | 01110110     | 118            | v      |
| 01011110     | 94             | ^      | 01110111     | 119            | w      |
| 01011111     | 95             | _      | 01111000     | 120            | x      |
| 01100000     | 96             | `      | 01111001     | 121            | y      |
| 01100001     | 97             | a      | 01111010     | 122            | z      |
| 01100010     | 98             | b      | 01111011     | 123            | {      |
| 01100011     | 99             | c      | 01111100     | 124            |        |
| 01100100     | 100            | d      | 01111101     | 125            | }      |
| 01100101     | 101            | e      | 01111110     | 126            | ~      |
| 01100110     | 102            | f      | 01111111     | 127            |        |

Таблица П.2

## Расширение CP-866

|     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| А   | В   | В   | Г   | Д   | Е   | Ж   | З   | И   | Й   | К   | Л   | М   | Н   | О   | П   |
| 128 | 129 | 130 | 131 | 132 | 133 | 134 | 135 | 136 | 137 | 138 | 139 | 140 | 141 | 142 | 143 |
| Р   | С   | Т   | У   | Ф   | Х   | Ц   | Ч   | Ш   | Щ   | Ъ   | Ы   | Ь   | Э   | Ю   | Я   |
| 144 | 145 | 146 | 147 | 148 | 149 | 150 | 151 | 152 | 153 | 154 | 155 | 156 | 157 | 158 | 159 |
| а   | б   | в   | г   | д   | е   | ж   | з   | и   | й   | к   | л   | м   | н   | о   | п   |
| 160 | 161 | 162 | 163 | 164 | 165 | 166 | 167 | 168 | 169 | 170 | 171 | 172 | 173 | 174 | 175 |
| ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   |
| 176 | 177 | 178 | 179 | 180 | 181 | 182 | 183 | 184 | 185 | 186 | 187 | 188 | 189 | 190 | 191 |
| ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   |
| 192 | 193 | 194 | 195 | 196 | 197 | 198 | 199 | 200 | 201 | 202 | 203 | 204 | 205 | 206 | 207 |
| ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   | ⌘   |
| 208 | 209 | 210 | 211 | 212 | 213 | 214 | 215 | 216 | 217 | 218 | 219 | 220 | 221 | 222 | 223 |
| р   | с   | т   | у   | ф   | х   | ц   | ч   | ш   | щ   | ъ   | ы   | ь   | э   | ю   | я   |
| 224 | 225 | 226 | 227 | 228 | 229 | 230 | 231 | 232 | 233 | 234 | 235 | 236 | 237 | 238 | 239 |
| ≡   | ±   | ≥   | ≤   | ┌   | ┐   | ÷   | ≈   | °   | •   | •   | √   | •   | •   | •   | •   |
| 240 | 241 | 242 | 243 | 244 | 245 | 246 | 247 | 248 | 249 | 250 | 251 | 252 | 253 | 254 | 255 |

Таблица П.3

## Расширение CP-1251

|     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Ъ   | Ѓ   | Ѕ   | Ї   | Ї   | Ї   | Ї   | Ї   | Ї   | Ї   | Ї   | Ї   | Ї   | Ї   | Ї   | Ї   |
| 128 | 129 | 130 | 131 | 132 | 133 | 134 | 135 | 136 | 137 | 138 | 139 | 140 | 141 | 142 | 143 |
| ђ   | ‘   | ’   | “   | ”   | •   | —   | —   | □   | ™   | Љ   | Ў   | ѐ   | ћ   | ц   |     |
| 144 | 145 | 146 | 147 | 148 | 149 | 150 | 151 | 152 | 153 | 154 | 155 | 156 | 157 | 158 | 159 |
| џ   | џ   | џ   | џ   | џ   | џ   | џ   | џ   | џ   | џ   | џ   | џ   | џ   | џ   | џ   | џ   |
| 160 | 161 | 162 | 163 | 164 | 165 | 166 | 167 | 168 | 169 | 170 | 171 | 172 | 173 | 174 | 175 |
| °   | ±   | І   | і   | І   | і   | І   | і   | І   | і   | І   | і   | І   | і   | І   | і   |
| 176 | 177 | 178 | 179 | 180 | 181 | 182 | 183 | 184 | 185 | 186 | 187 | 188 | 189 | 190 | 191 |
| А   | Б   | В   | Г   | Д   | Е   | Ж   | З   | И   | Й   | К   | Л   | М   | Н   | О   | П   |
| 192 | 193 | 194 | 195 | 196 | 197 | 198 | 199 | 200 | 201 | 202 | 203 | 204 | 205 | 206 | 207 |
| Р   | С   | Т   | У   | Ф   | Х   | Ц   | Ч   | Ш   | Щ   | Ъ   | Ы   | Ь   | Э   | Ю   | Я   |
| 208 | 209 | 210 | 211 | 212 | 213 | 214 | 215 | 216 | 217 | 218 | 219 | 220 | 221 | 222 | 223 |
| а   | б   | в   | г   | д   | е   | ж   | з   | и   | й   | к   | л   | м   | н   | о   | п   |
| 224 | 225 | 226 | 227 | 228 | 229 | 230 | 231 | 232 | 233 | 234 | 235 | 236 | 237 | 238 | 239 |
| р   | с   | т   | у   | ф   | х   | ц   | ч   | ш   | щ   | ъ   | ы   | ь   | э   | ю   | я   |
| 240 | 241 | 242 | 243 | 244 | 245 | 246 | 247 | 248 | 249 | 250 | 251 | 252 | 253 | 254 | 255 |

Таблица П.4

## Расширение КОИ-8 (koi-8r)

|     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| —   |     | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   |
| 128 | 129 | 130 | 131 | 132 | 133 | 134 | 135 | 136 | 137 | 138 | 139 | 140 | 141 | 142 | 143 |
| ⌘   | ⌘   | ⌘   | Г   | ·   | ·   | √   | ≈   | ≤   | ≥   |     | Г   | °   | °   | °   | ÷   |
| 144 | 145 | 146 | 147 | 148 | 149 | 150 | 151 | 152 | 153 | 154 | 155 | 156 | 157 | 158 | 159 |
| =   |     | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   |
| 160 | 161 | 162 | 163 | 164 | 165 | 166 | 167 | 168 | 169 | 170 | 171 | 172 | 173 | 174 | 175 |
|     |     | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   | Г   |
| 176 | 177 | 178 | 179 | 180 | 181 | 182 | 183 | 184 | 185 | 186 | 187 | 188 | 189 | 190 | 191 |
| ю   | а   | б   | ц   | д   | е   | ф   | г   | х   | и   | й   | к   | л   | м   | н   | о   |
| 192 | 193 | 194 | 195 | 196 | 197 | 198 | 199 | 200 | 201 | 202 | 203 | 204 | 205 | 206 | 207 |
| п   | я   | р   | с   | т   | у   | ж   | в   | ь   | ы   | э   | ш   | э   | ш   | ч   | ъ   |
| 208 | 209 | 210 | 211 | 212 | 213 | 214 | 215 | 216 | 217 | 218 | 219 | 220 | 221 | 222 | 223 |
| Ю   | А   | В   | Ц   | Д   | Е   | Ф   | Г   | Х   | И   | Й   | К   | Л   | М   | Н   | О   |
| 224 | 225 | 226 | 227 | 228 | 229 | 230 | 231 | 232 | 233 | 234 | 235 | 236 | 237 | 238 | 239 |
| П   | Я   | Р   | С   | Т   | У   | Ж   | В   | Ь   | Ы   | Э   | Ш   | Э   | Ш   | Ч   | Ъ   |
| 240 | 241 | 242 | 243 | 244 | 245 | 246 | 247 | 248 | 249 | 250 | 251 | 252 | 253 | 254 | 255 |

# Содержание

|                                                                                         |           |
|-----------------------------------------------------------------------------------------|-----------|
| Предисловие . . . . .                                                                   | 3         |
| <b>Раздел 1. Информация, виды информации и способы её представления . . . . .</b>       | <b>5</b>  |
| § 1. Информация и информационные процессы . . . . .                                     | —         |
| § 2. Кодирование символьной информации . . . . .                                        | 11        |
| § 3. Кодирование числовой информации . . . . .                                          | 14        |
| 3.1. Позиционные системы счисления с произвольным основанием . . . . .                  | —         |
| 3.2. Системы счисления, используемые в программировании . . . . .                       | 17        |
| 3.3. <b>П</b> Уравновешенные и другие системы счисления                                 |           |
| § 4. Кодирование видеоинформации . . . . .                                              | 22        |
| § 5. <b>П</b> Коды, обнаруживающие и исправляющие ошибки                                | 26        |
| § 6. Измерение количества информации . . . . .                                          | 27        |
| 6.1. Информационный объём сообщения . . . . .                                           | 28        |
| 6.2. <b>П</b> Экономное кодирование. Алгоритмы сжатия информации . . . . .              | 32        |
| <b>Раздел 2. Основные информационные технологии . . . . .</b>                           | <b>38</b> |
| § 7. Обработка текстовой информации . . . . .                                           | —         |
| § 8. Обработка числовой информации с помощью электронной таблицы . . . . .              | 40        |
| § 9. Базы данных и информационно-поисковые системы . . . . .                            | 43        |
| <b>Раздел 3. Алгоритмизация, структуры данных и элементы программирования . . . . .</b> | <b>48</b> |
| § 10. Понятие алгоритма и исполнителя. Линейные алгоритмы . . . . .                     | —         |
| § 11. Алгоритмические конструкции . . . . .                                             | 52        |
| § 12. Переменные в алгоритмах . . . . .                                                 | 62        |
| 12.1. Переменные числового типа . . . . .                                               | —         |
| 12.2. Символьные и строковые переменные . . . . .                                       | 68        |
| 12.3. Переменные логического типа . . . . .                                             | 73        |
| § 13. Вспомогательные алгоритмы и подпрограммы . . . . .                                | 75        |

|                                                            |            |
|------------------------------------------------------------|------------|
| 13.1. Вспомогательный алгоритм-процедура . . . . .         | 75         |
| 13.2. Вспомогательный алгоритм-функция . . . . .           | 77         |
| 13.3. Рекурсия . . . . .                                   | 80         |
| 13.4. Нисходящее и восходящее программирование . . . . .   | 82         |
| § 14. Массивы . . . . .                                    | 85         |
| § 15. <b>П</b> Графы и алгоритмы на графах . . . . .       | 91         |
| 15.1. Свойства графов . . . . .                            | 93         |
| 15.2. Алгоритмы поиска на графе и орграфе . . . . .        | 96         |
| § 16. <b>П</b> Игры и стратегии . . . . .                  | 100        |
| § 17. <b>П</b> Основные вычислительные методы . . . . .    | 105        |
| 17.1. Методы приближённого решения уравнений . . . . .     | —          |
| 17.2. Датчики случайных чисел. Метод Монте-Карло . . . . . | 108        |
| § 18. Свойства алгоритмов . . . . .                        | 112        |
| § 19. <b>П</b> Машина Тьюринга . . . . .                   | 121        |
| § 20. <b>П</b> Языки программирования . . . . .            | 125        |
| <b>Раздел 4. Основы вычислительной техники . . . . .</b>   | <b>130</b> |
| § 21. Элементы математической логики . . . . .             | 131        |
| 21.1. Алгебра логики . . . . .                             | —          |
| 21.2. Высказывательные формы (предикаты) . . . . .         | 134        |
| § 22. Релейно-контактные схемы . . . . .                   | 140        |
| § 23. <b>П</b> Логические элементы. Вентили . . . . .      | 142        |
| Ответы и результаты вычислительных экспериментов . . . . . | 148        |
| Основная литература . . . . .                              | 150        |
| Дополнительная литература . . . . .                        | —          |
| Приложения . . . . .                                       | 152        |

**Учебное издание**  
**Гейн Александр Георгиевич**  
**Информатика и ИКТ**  
**Задачник-практикум**  
**10—11 классы**

**Базовый и профильный уровни**

**Зав. редакцией Т. А. Бурмистрова**

**Редактор О. В. Платонова**

**Художник О. П. Богомолова**

**Художественный редактор О. П. Богомолова**

**Компьютерная графика А. Г. Вьюникова, О. Ю. Тупикина**

**Технический редактор и верстальщик Е. В. Саватеева**

**Корректоры Л. С. Александрова, Ю. В. Григорьева**

**Налоговая льгота — Общероссийский классификатор продукции ОК 005-93—953000. Изд. лиц. Серия ИД № 05824 от 12.09.01. Подписано в печать с оригинал-макета 16.08.10. Формат 60 × 90<sup>1/16</sup>. Бумага офсетная. Гарнитура Школьная. Печать офсетная. Уч.-изд. л. 9,69. Тираж 5000 экз. Заказ № 26354 (к-л).**

**Открытое акционерное общество «Издательство «Просвещение».**  
**127521, Москва, 3-й проезд Марьиной рощи, 41.**

**Открытое акционерное общество «Смоленский полиграфический комбинат».**  
**214020, Смоленск, ул. Смольянинова, 1.**

**УДК 373.167.1:004**  
**ББК 32.81я72**  
**Г29**

**Гейн А. Г.**  
**Г29 Информатика и ИКТ. Задачник-практикум. 10—11 клас-**  
**сы: базовый и профил. уровни / А. Г. Гейн. — М. : Про-**  
**свещение, 2010. — 157 с. : ил. — ISBN 978-5-09-019446-4.**  
**УДК 373.167.1:004**  
**ББК 32.81я72**

**ISBN 978-5-09-019446-4**

© Издательство «Просвещение», 2010  
© Художественное оформление.  
Издательство «Просвещение», 2010  
Все права защищены



# ИНФОРМАТИКА и ИКТ

## ЗАДАЧНИК-ПРАКТИКУМ

Учебно-методический комплект  
по информатике и ИКТ включает:

А. Г. Гейн, А. Б. Ливчак,  
А. И. Сенокосов,  
Н. А. Юнерман  
**Учебник для 10 класса**

А. Г. Гейн, А. И. Сенокосов  
**Учебник для 11 класса**

А. Г. Гейн  
**Книга для учителя**  
**Методические рекомендации**  
**к учебнику для 10 класса**

А. Г. Гейн, Н. А. Юнерман, А. А. Гейн  
**Книга для учителя**  
**Методические рекомендации**  
**к учебнику для 11 класса**

А. Г. Гейн, Н. А. Юнерман  
**Тематические тесты для 10 класса**

А. Г. Гейн, Н. А. Юнерман  
**Тематические тесты для 11 класса**

А. Г. Гейн  
**Задачник-практикум**  
**для 10 – 11 классов**

ISBN 978-5-09-019446-4



9 785090 194464



**ПРОСВЕЩЕНИЕ**  
ИЗДАТЕЛЬСТВО