

ФГОС

8



Л. Л. Босова  
А. Ю. Босова

ИНФОРМАТИКА



ИЗДАТЕЛЬСТВО

БИНОМ

**ФГОС**

**Л. Л. Босова, А. Ю. Босова**

# **ИНФОРМАТИКА**

**Учебник  
для 8 класса**

**2-е издание, исправленное**

Рекомендовано  
Министерством образования и науки  
Российской Федерации  
к использованию в образовательном процессе  
в имеющих государственную аккредитацию  
и реализующих образовательные программы  
общего образования образовательных учреждениях

**爱  
谢谢**



Москва  
БИНОМ. Лаборатория знаний  
2014

# Введение

## *Уважаемые восьмиклассники!*

Мы живём во время стремительных перемен, когда для человека важна способность к постоянному развитию, готовность к освоению новых, в том числе информационных, технологий. Необходимость подготовки к быстро наступающим переменам в окружающем мире требует от человека развитого мышления, умения организации собственной учебной деятельности, ориентации на деятельностную жизненную позицию. Формирование таких качеств личности невозможно без фундаментального базового образования.

В курсе информатики 8 класса большое внимание уделяется фундаментальным (теоретическим) вопросам информатики. Вы будете изучать математические основы информатики, освоите базовые алгоритмические конструкции, познакомитесь с языком программирования Паскаль.

Мы надеемся, что знания и умения, полученные на уроках информатики, вы сможете применять в дальнейшем при решении разнообразных жизненных задач, предполагающих такие этапы, как:

- **целеполагание** — постановка задачи на основе соотнесения того, что уже известно, и того, что требуется установить;
- **планирование** — определение последовательности промежуточных целей с учётом конечного результата, разбиение задачи на подзадачи, разработка последовательности и структуры действий, необходимых для достижения цели с помощью фиксированного набора средств;
- **прогнозирование** — предвосхищение результата;
- **контроль** — интерпретация полученного результата, его соотнесение с имеющимися данными с целью установления соответствия или несоответствия (обнаружения ошибки);
- **коррекция** — внесение необходимых дополнений и исправлений в план действий в случае обнаружения ошибки;
- **оценка** — осознание человеком того, насколько качественно им решена задача.











Как и ранее, в этом учебнике кроме основной информации содержатся многочисленные ссылки на образовательные ресурсы сети Интернет, в том числе на такие порталы, как:

- 1) Единая коллекция цифровых образовательных ресурсов (<http://sc.edu.ru/>);
- 2) Федеральный центр информационных образовательных ресурсов (<http://fcior.edu.ru/>);
- 3) сайт методической службы издательства (<http://metodist.Lbz.ru/>).

На страницах учебника подробно рассмотрены решения типовых задач по каждой изучаемой теме. В конце каждой главы учебника приведены тестовые задания, которые помогут вам оценить, хорошо ли вы освоили теоретический материал и можете ли применять свои знания для решения возникающих проблем.

Изучая теоретический материал, работая с дополнительными материалами, отвечая на вопросы, решая задачи и выполняя практические задания на компьютере, вы сможете полностью подготовиться к сдаче выпускного экзамена по курсу информатики и в форме государственной итоговой аттестации (ГИА), требования к которой размещены на сайте <http://fipi.ru>.

В работе с учебником вам помогут навигационные значки:

-  — важное утверждение или определение;
-  — интересная информация;
-  — пример решения задачи;
-  — информация, полезная для решения практических задач;
-  — ссылка на ресурс в Интернете;
-  — дополнительный материал к параграфу, содержащийся в электронном приложении к учебнику (<http://metodist.Lbz.ru/>);
-  — вопросы в тексте параграфа, вопросы и задания для самоконтроля;
-  — задания для подготовки к итоговой аттестации;
-  — домашний проект или исследование;
-  — задания для практических работ на компьютере.

Желаем успехов в изучении информатики!

---

# Глава 1

## МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ

### § 1.1

#### Системы счисления

**Ключевые слова:**

- система счисления
- цифра
- алфавит
- позиционная система счисления
- основание
- развёрнутая форма записи числа
- свёрнутая форма записи числа
- двоичная система счисления
- восьмеричная система счисления
- шестнадцатеричная система счисления

#### 1.1.1. Общие сведения о системах счисления

---

**Система счисления** — это знаковая система, в которой приняты определённые правила записи чисел. Знаки, с помощью которых записываются числа (рис. 1.1), называются **цифрами**, а их совокупность — **алфавитом** системы счисления.

---

В любой системе счисления цифры служат для обозначения чисел, называемых *узловыми*; остальные числа (*алгоритмические*) получаются в результате каких-либо операций из узловых чисел.

**Пример 1.** У вавилонян узловыми являлись числа 1, 10, 60; в римской системе счисления узловые числа — это 1, 5, 10, 50, 100, 500 и 1000, обозначаемые соответственно I, V, X, L, C, D, M.



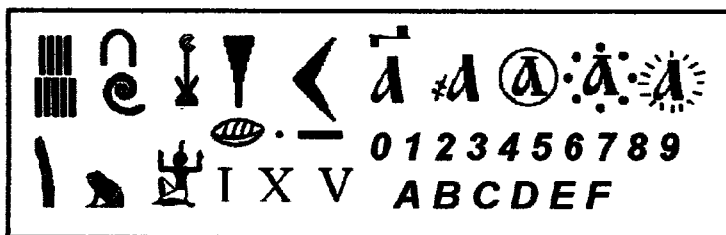


Рис. 1.1. Знаки, используемые для записи чисел в различных системах счисления

Системы счисления различаются выбором узловых чисел и способами образования алгоритмических чисел. Можно выделить следующие виды систем счисления:

- 1) унарная система;
- 2) непозиционные системы;
- 3) позиционные системы.

Простейшая и самая древняя система — так называемая **унарная система счисления**. В ней для записи любых чисел используется всего один символ — палочка, узелок, зарубка, камушек. Длина записи числа при таком кодировании прямо связана с его величиной, что роднит этот способ с геометрическим представлением чисел в виде отрезков. Именно унарная система лежит в фундаменте арифметики, и именно она до сих пор вводит первоклассников в мир счёта. Унарную систему ещё называют системой бирок.



Система счисления называется **непозиционной**, если количественный эквивалент (количественное значение) цифры в числе не зависит от её положения в записи числа.

В большинстве непозиционных систем счисления числа образуются путём сложения узловых чисел.

**Пример 2.** В древнеегипетской системе счисления числа 1, 2, 3, 4, 10, 13, 40 обозначались соответственно следующим образом:

|, ||, |||, ||||, ∪, ∪∪, ∪∪∪

Те же числа в римской системе счисления обозначаются так: I, II, III, IV, X, XIII, XL. Здесь алгоритмические числа получаются путём сложения и вычитания узловых чисел с учётом следующего правила: каждый меньший знак, поставленный справа от большего, прибавляется к его значению, а каждый меньший знак, поставленный слева от большего, вычитается из него.

Система счисления называется **позиционной**, если количественный эквивалент цифры зависит от её положения (позиции) в записи числа.

**Основание** позиционной системы счисления равно количеству цифр, составляющих её алфавит.

Десятичная система записи чисел, которой мы привыкли пользоваться в повседневной жизни, с которой мы знакомы с детства, в которой производим все наши вычисления, — пример позиционной системы счисления. Алфавит десятичной системы составляют цифры 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Алгоритмические числа образуются в ней следующим образом: значения цифр умножаются на «веса» соответствующих разрядов, и все полученные значения складываются. Это отчётливо прослеживается в числительных русского языка, например: «три-ста пять-десят семь».

Основанием позиционной системы счисления может служить любое натуральное число  $q > 1$ . Алфавитом произвольной позиционной системы счисления с основанием  $q$  служат числа 0, 1, ...,  $q-1$ , каждое из которых может быть записано с помощью одного уникального символа; младшей цифрой всегда является 0.

Основные достоинства любой позиционной системы счисления — простота выполнения арифметических операций и ограниченное количество символов, необходимых для записи любых чисел.

В позиционной системе счисления с основанием  $q$  любое число может быть представлено в виде:

$$A_q = \pm (a_{n-1} \cdot q^{n-1} + a_{n-2} \cdot q^{n-2} + \dots + a_0 \cdot q^0 + a_{-1} \cdot q^{-1} + \dots + a_{-m} \cdot q^{-m}). \quad (1)$$

Здесь:

$A$  — число;

$q$  — основание системы счисления;

$a_i$  — цифры, принадлежащие алфавиту данной системы счисления;

$n$  — количество целых разрядов числа;

$m$  — количество дробных разрядов числа;

$q^i$  — «вес»  $i$ -го разряда.

Запись числа по формуле (1) называется **развёрнутой формой записи**. **Свёрнутой формой** записи числа называется его представление в виде<sup>1</sup>  $\pm a_{n-1}a_{n-2}\dots a_1a_0,a_{-1}\dots a_{-m}$ .

<sup>1</sup> Далее будут рассматриваться только положительные целые числа.



**Пример 3.** Рассмотрим десятичное число 14351,1. Его свёрнутая форма записи настолько привычна, что мы не замечаем, как в уме переходим к развёрнутой записи, умножая цифры числа на «веса» разрядов и складывая полученные произведения:

$$1 \cdot 10^4 + 4 \cdot 10^3 + 3 \cdot 10^2 + 5 \cdot 10^1 + 1 \cdot 10^0 + 1 \cdot 10^{-1}.$$

## 1.1.2. Двоичная система счисления

**Двоичной системой счисления** называется позиционная система счисления с основанием 2. Для записи чисел в двоичной системе счисления используются только две цифры: 0 и 1.

На основании формулы (1) для целых двоичных чисел можно записать:

$$a_{n-1}a_{n-2}\dots a_1a_0 = a_{n-1} \cdot 2^{n-1} + a_{n-2} \cdot 2^{n-2} + \dots + a_0 \cdot 2^0. \quad (1')$$

Например:

$$10011_2 = 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 2^4 + 2^1 + 2^0 = 19_{10}.$$



Такая форма записи «подсказывает» правило перевода натуральных двоичных чисел в десятичную систему счисления: *необходимо вычислить сумму степеней двойки, соответствующих единицам в свёрнутой форме записи двоичного числа.*



Получим правило перевода целых десятичных чисел в двоичную систему счисления из формулы (1').

Разделим  $a_{n-1} \cdot 2^{n-1} + a_{n-2} \cdot 2^{n-2} + \dots + a_0 \cdot 2^0$  на 2. Частное будет равно  $a_{n-1} \cdot 2^{n-2} + \dots + a_1$ , а остаток будет равен  $a_0$ .

Полученное частное опять разделим на 2, остаток от деления будет равен  $a_1$ .

Если продолжить этот процесс деления, то на  $n$ -м шаге получим набор цифр:

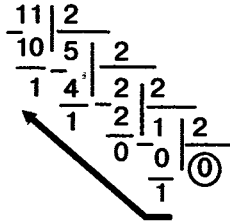
$$a_0, a_1, a_2, \dots, a_{n-1},$$

которые входят в двоичное представление исходного числа и совпадают с остатками при его последовательном делении на 2.

Таким образом, для перевода целого десятичного числа в двоичную систему счисления нужно последовательно выполнять деление данного числа и получаемых целых частных на 2 до тех пор, пока не получим частное, равное нулю. Исходное число в двоичной системе счисления составляется последовательной записью полученных остатков, начиная с последнего.



**Пример 4.** Переведём десятичное число 11 в двоичную систему счисления. Рассмотренную выше последовательность действий (алгоритм перевода) можно изобразить так:



Выписывая остатки от деления в направлении, указанном стрелкой, получим:  $11_{10} = 1011_2$ .

**Пример 5.** Если десятичное число достаточно большое, то более удобен следующий способ записи рассмотренного выше алгоритма:

363	181	90	45	22	11	5	2	1
1	1	0	1	0	1	1	0	1

←

$$363_{10} = 101101011_2$$

### 1.1.3. Восьмеричная система счисления

Восьмеричной системой счисления называется позиционная система счисления с основанием 8. Для записи чисел в восьмеричной системе счисления используются цифры: 0, 1, 2, 3, 4, 5, 6, 7.

На основании формулы (1) для целого восьмеричного числа можно записать:

$$a_{n-1}a_{n-2}\dots a_1a_0 = a_{n-1} \cdot 8^{n-1} + a_{n-2} \cdot 8^{n-2} + \dots + a_0 \cdot 8^0. \quad (1'')$$

Например:  $1063_8 = 1 \cdot 8^3 + 0 \cdot 8^2 + 6 \cdot 8^1 + 3 \cdot 8^0 = 563_{10}$ .

Таким образом, для перевода целого восьмеричного числа в десятичную систему счисления следует перейти к его развёрнутой записи и вычислить значение получившегося выражения.

Для перевода целого десятичного числа в восьмеричную систему счисления следует последовательно выполнять деление данного числа и получаемых целых частных на 8 до тех пор, пока не получим частное, равное нулю. Исходное число в новой системе счисления составляется последовательной записью полученных остатков, начиная с последнего.

**Пример 6.** Переведём десятичное число 103 в восьмеричную систему счисления.

$$\begin{array}{r|l}
 103 & 8 \\
 \hline
 8 & 12 \\
 \hline
 23 & 8 \\
 \hline
 16 & 4 \\
 \hline
 7 & 0 \\
 \hline
 & 1
 \end{array}
 \begin{array}{l}
 \\
 \\
 \\
 \\
 \\
 \textcircled{0}
 \end{array}$$

$$103_{10} = 147_8$$

## 1.1.4. Шестнадцатеричная система счисления

Основание:  $q = 16$ .

Алфавит: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

Здесь только десять цифр из шестнадцати имеют общепринятое обозначение 0, ..., 9. Для записи цифр с десятичными количественными эквивалентами 10, 11, 12, 13, 14, 15 обычно используются первые пять букв латинского алфавита.

Таким образом, запись  $3AF_{16}$  означает:

$$3AF_{16} = 3 \cdot 16^2 + 10 \cdot 16^1 + 15 \cdot 16^0 = 768 + 160 + 15 = 943_{10}.$$

**Пример 7.** Переведём десятичное число 154 в шестнадцатеричную систему счисления.

$$\begin{array}{r|l}
 154 & 16 \\
 \hline
 144 & 9 \\
 \hline
 10 & 0 \\
 \hline
 (A) & 9 \\
 \hline
 & \textcircled{0}
 \end{array}$$

$$154_{10} = 9A_{16}$$

## 1.1.5. Правило перевода целых десятичных чисел в систему счисления с основанием $q$

Для перевода целого десятичного числа в систему счисления с основанием  $q$  следует:

- 1) последовательно выполнять деление данного числа и получаемых целых частных на основание новой системы счисления до тех пор, пока не получим частное, равное нулю;
- 2) полученные остатки, являющиеся цифрами числа в новой системе счисления, привести в соответствие с алфавитом новой системы счисления;

3) составить число в новой системе счисления, записывая его, начиная с последнего полученного остатка.

Представим таблицу соответствия десятичных, двоичных, восьмеричных и шестнадцатеричных чисел от 0 до  $20_{10}$ .

Десятичная система	Двоичная система	Восьмеричная система	Шестнадцатеричная система
0	0	0	0
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
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14



В Единой коллекции цифровых образовательных ресурсов (<http://sc.edu.ru/>) размещена интерактивная анимация «Преобразование десятичного числа в другую систему счисления» (135050). С её помощью можно понаблюдать за переводом произвольного целого числа от 0 до 512 в позиционную систему счисления, основание которой не превышает 16.

В размещённой там же виртуальной лаборатории «Цифровые весы» (135009) вы сможете освоить ещё один способ перевода целых десятичных чисел в другие системы счисления — метод разностей.

## 1.1.6. Двоичная арифметика

Арифметика двоичной системы счисления основывается на использовании следующих таблиц сложения и умножения:

+	0	1
0	0	1
1	1	10

×	0	1
0	0	0
1	0	1



**Пример 8.** Таблица двоичного сложения предельно проста. Так как  $1 + 1 = 10$ , то 0 остаётся в младшем разряде, а 1 переносится в старший разряд.

+	1	0	0	1
	1	0	1	0
	1	0	1	1

+	1	1	1	1
				1
	1	0	0	0



**Пример 9.** Операция умножения двоичных чисел выполняется по обычной схеме, применяемой в десятичной системе счисления, с последовательным умножением множимого на очередную цифру множителя.

			1	0	1	1
				1	0	1
				1	0	1
				1	0	1
1	0		1	1		
1	1	0	1	1	1	1

Таким образом, в двоичной системе счисления умножение сводится к сдвигам множимого и сложениям.

### 1.1.7. «Компьютерные» системы счисления

В компьютерной технике используется двоичная система счисления, обеспечивающая ряд преимуществ по сравнению с другими системами счисления:

- двоичные числа представляются в компьютере с помощью достаточно простых технических элементов с двумя устойчивыми состояниями;
- представление информации посредством только двух состояний надёжно и помехоустойчиво;
- двоичная арифметика наиболее проста;
- существует математический аппарат, обеспечивающий логические преобразования двоичных данных.

Обмен информацией между компьютерными устройствами осуществляется путём передачи двоичных кодов. Пользоваться такими кодами из-за их большой длины и зрительной однородности человеку неудобно. Поэтому специалисты (программисты, инженеры) на некоторых этапах разработки, создания, настройки вычислительных систем заменяют двоичные коды на эквивалентные им величины в восьмеричной или шестнадцатеричной системах счисления. В результате длина исходного слова сокращается в три, четыре раза соответственно. Это делает информацию более удобной для рассмотрения и анализа.

С помощью ресурса «Интерактивный задачник, раздел «Системы счисления»» (128659), размещённого в Единой коллекции цифровых образовательных ресурсов, можно проверить, насколько прочно вы усвоили изученный в этом параграфе материал.



## САМОЕ ГЛАВНОЕ

Система счисления — это знаковая система, в которой приняты определённые правила записи чисел. Знаки, с помощью которых записываются числа, называются цифрами, а их совокупность — алфавитом системы счисления.

Система счисления называется позиционной, если количественный эквивалент цифры зависит от её положения (позиции) в записи числа. Основание позиционной системы счисления равно количеству цифр, составляющих её алфавит.

Основанием позиционной системы счисления может служить любое натуральное число  $q > 1$ .



В позиционной системе счисления с основанием  $q$  любое число может быть представлено в виде:

$$A_q = \pm (a_{n-1} \cdot q^{n-1} + a_{n-2} \cdot q^{n-2} + \dots + a_0 \cdot q^0 + a_{-1} \cdot q^{-1} + \dots + a_{-m} \cdot q^{-m}).$$

Здесь:

$A$  — число;

$q$  — основание системы счисления;

$a_i$  — цифры, принадлежащие алфавиту данной системы счисления;

$n$  — количество целых разрядов числа;

$m$  — количество дробных разрядов числа;

$q^i$  — «вес»  $i$ -го разряда.








## Вопросы и задания



1. Ознакомьтесь с материалами презентации к параграфу, содержащейся в электронном приложении к учебнику. Что вы можете сказать о формах представления информации в презентации и в учебнике? Какими слайдами вы могли бы дополнить презентацию?
2. Найдите дополнительную информацию об унарной, позиционных и непозиционных системах счисления. Чем они различаются? Приведите примеры.
3. Цифры каких систем счисления приведены на рис. 1.1?
4. Объясните, почему позиционные системы счисления с основаниями 5, 10, 12 и 20 называют системами счисления анатомического происхождения.
5. Как от свёрнутой формы записи десятичного числа перейти к его развёрнутой форме?
6. Запишите в развёрнутой форме числа:
  - а)  $143,511_{10}$ ;
  - б)  $143511_8$ ;
  - в)  $143511_{16}$ ;
  - г)  $1435,11_8$ .
7. Вычислите десятичные эквиваленты следующих чисел:
  - а)  $172_8$ ;
  - б)  $2EA_{16}$ ;
  - в)  $101010_2$ ;
  - г)  $10,1_2$ ;
  - д)  $243_6$ .

8. Укажите, какое из чисел  $110011_2$ ,  $111_4$ ,  $35_8$  и  $1B_{16}$  является:
- наибольшим;
  - наименьшим.
9. Какое минимальное основание имеет система счисления, если в ней записаны числа 123, 222, 111, 241? Определите десятичный эквивалент данных чисел в найденной системе счисления.
10. Верны ли следующие равенства?
- $33_4 = 21_7$ ;
  - $33_8 = 21_4$ .
11. Найдите основание  $x$  системы счисления, если:
- $14_x = 9_{10}$ ;
  - $2002_x = 130_{10}$ .
12. Переведите целые числа из десятичной системы счисления в двоичную:
- 89;
  - 600;
  - 2010.
13. Переведите целые числа из десятичной системы счисления в восьмеричную:
- 513;
  - 600;
  - 2010.
14. Переведите целые числа из десятичной системы счисления в шестнадцатеричную:
- 513;
  - 600;
  - 2010.
15. Заполните таблицу, в каждой строке которой одно и то же число должно быть записано в системах счисления с основаниями 2, 8, 10 и 16.

Основание 2	Основание 8	Основание 10	Основание 16
101010			
	127		
		321	
			2A

-  16. Выполните операцию сложения над двоичными числами:
- а)  $101010 + 1101$ ;
  - б)  $1010 + 1010$ ;
  - в)  $10101 + 111$ .
-  17. Выполните операцию умножения над двоичными числами:
- а)  $1010 \cdot 11$ ;
  - б)  $111 \cdot 101$ ;
  - в)  $1010 \cdot 111$ .
18. Расставьте знаки арифметических операций так, чтобы были верны следующие равенства в двоичной системе:
- а)  $1100 ? 11 ? 100 = 100000$ ;
  - б)  $1100 ? 10 ? 10 = 100$ ;
  - в)  $1100 ? 11 ? 100 = 0$ .
-  19. Вычислите выражения:
- а)  $(1111101_2 + AF_{16}) : 36_8$ ;
  - б)  $125_8 + 101_2 \cdot 2A_{16} - 141_8$ .
- Ответ дайте в десятичной системе счисления.
20. Какими преимуществами и недостатками обладает двоичная система счисления по сравнению с десятичной?
21. Разработайте таблицы сложения и умножения для восьмеричной системы счисления.
22. Постройте граф, отражающий разновидности систем счисления.
-  23. Подготовьте небольшое сообщение об одной из систем счисления (когда и где применялась, какие символы использовались и т. д.). Можете воспользоваться материалами электронного приложения к учебнику.
- 



## Представление чисел в компьютере

### Ключевые слова:

- разряд
- беззнаковое представление целых чисел
- представление целых чисел со знаком
- представление вещественных чисел

### 1.2.1. Представление целых чисел

Оперативная память компьютера состоит из ячеек, каждая из которых представляет собой физическую систему, состоящую из некоторого числа однородных элементов. Эти элементы обладают двумя устойчивыми состояниями, одно из которых соответствует нулю, а другое — единице. Каждый такой элемент служит для хранения одного из битов — разряда двоичного числа. Именно поэтому каждый элемент ячейки называют битом или разрядом (рис. 1.2).

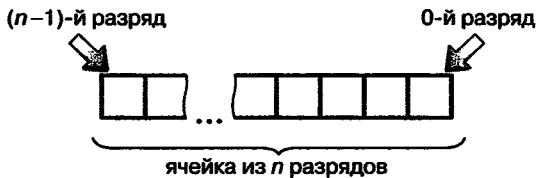


Рис. 1.2. Ячейка памяти

Для компьютерного представления целых чисел используется несколько различных способов, отличающихся друг от друга количеством разрядов (под целые числа обычно отводится 8, 16, 32 или 64 разряда) и наличием или отсутствием знакового разряда. Беззна-

ковое представление можно использовать только для неотрицательных целых чисел, отрицательные числа представляются только в знаковом виде.

Беззнаковое представление используется для таких объектов, как адреса ячеек, всевозможные счётчики (например, число символов в тексте), а также числа, обозначающие дату и время, размеры графических изображений в пикселях и т. д.

Максимальное значение целого неотрицательного числа достигается в случае, когда во всех разрядах ячейки хранятся единицы. Для  $n$ -разрядного представления оно будет равно  $2^n - 1$ . Минимальное число соответствует  $n$  нулям, хранящимся в  $n$  разрядах памяти, и равно нулю.

Ниже приведены максимальные значения для беззнаковых целых  $n$ -разрядных чисел:

Количество битов	Минимальное значение	Максимальное значение
8	0	255 ( $2^8 - 1$ )
16	0	65 535 ( $2^{16} - 1$ )
32	0	4 294 967 295 ( $2^{32} - 1$ )
64	0	18 446 744 073 709 551 615 ( $2^{64} - 1$ )

Для получения компьютерного представления беззнакового целого числа достаточно перевести число в двоичную систему счисления и дополнить полученный результат слева нулями до стандартной разрядности.



**Пример 1.** Число  $53_{10} = 110101_2$  в восьмиразрядном представлении имеет вид:

0	0	1	1	0	1	0	1
---	---	---	---	---	---	---	---

Это же число 53 в шестнадцати разрядах будет записано следующим образом:

0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

При представлении со знаком самый старший (левый) разряд отводится под знак числа, остальные разряды — под само число. Если число положительное, то в знаковый разряд помещается 0, если число отрицательное — 1. Такое представление чисел называется прямым кодом. В компьютере прямые коды используются для хранения положительных чисел в запоминающих устройствах, для выполнения операций с положительными числами.

На сайте Федерального центра информационно-образовательных ресурсов (<http://fcior.edu.ru/>) размещён информационный модуль «Число и его компьютерный код». С помощью этого ресурса вы можете получить дополнительную информацию по изучаемой теме.



Для выполнения операций с отрицательными числами используется **дополнительный код**, позволяющий заменить операцию вычитания сложением. Узнать алгоритм образования дополнительного кода вы можете с помощью информационного модуля «Дополнительный код», размещённого на сайте Федерального центра информационно-образовательных ресурсов (<http://fcior.edu.ru/>).



## 1.2.2. Представление вещественных чисел

Любое вещественное число  $A$  может быть записано в экспоненциальной форме:

$$A = \pm m \cdot q^p,$$

где:

$m$  — мантисса числа;

$q$  — основание системы счисления;

$p$  — порядок числа.

Например, число 472 000 000 может быть представлено так:  $4,72 \cdot 10^8$ ,  $47,2 \cdot 10^7$ ,  $472,0 \cdot 10^6$  и т. д.

С экспоненциальной формой записи чисел вы могли встречаться при выполнении вычислений с помощью калькулятора, когда в качестве ответа получали записи следующего вида:  $4.72E+8$ .



Здесь знак «Е» обозначает основание десятичной системы счисления и читается как «умножить на десять в степени».

Из приведённого выше примера видно, что положение запятой в записи числа может изменяться.

Для единообразия мантиссу обычно записывают как правильную дробь, имеющую после запятой цифру, отличную от нуля. В этом случае число 472 000 000 будет представлено как  $0,472 \cdot 10^9$ .



При хранении в компьютере вещественных чисел выделяются разряды на хранение знака порядка числа, самого порядка, знака мантиссы и мантиссы. При этом любое число записывается так:

$$A = \pm m \cdot q^p,$$

где:  $m$  — мантисса числа;

$q$  — основание системы счисления;

$p$  — порядок числа.

### Вопросы и задания

1. Ознакомьтесь с материалами презентации к параграфу, содержащейся в электронном приложении к учебнику. Используйте эти материалы при подготовке ответов на вопросы и выполнении заданий.
2. Как в памяти компьютера представляются целые положительные и отрицательные числа?
3. Любое целое число можно рассматривать как вещественное, но с нулевой дробной частью. Обоснуйте целесообразность наличия особых способов компьютерного представления целых чисел.
4. Представьте число  $63_{10}$  в беззнаковом 8-разрядном формате.
5. Найдите десятичные эквиваленты чисел по их прямым кодам, записанным в 8-разрядном формате со знаком:
  - а) 01001100;
  - б) 00010101.
6. Какие из чисел  $443_8$ ,  $101010_2$ ,  $256_{10}$  можно сохранить в 8-разрядном формате?
7. Запишите следующие числа в естественной форме:
  - а)  $0,3800456 \cdot 10^2$ ;
  - б)  $0,245 \cdot 10^{-3}$ ;
  - а)  $1,256900E+5$ ;
  - а)  $9,569120E-3$ .
8. Запишите число  $2010,0102_{10}$  пятью различными способами в экспоненциальной форме.
9. Запишите следующие числа в экспоненциальной форме с нормализованной мантиссой — правильной дробью, имеющей после запятой цифру, отличную от нуля:
  - а)  $217,934_{10}$ ;
  - б)  $75321_{10}$ ;
  - в)  $0,00101_{10}$ .
10. Изобразите схему, связывающую основные понятия, рассмотренные в данном параграфе.



## § 1.3

### Элементы алгебры логики

#### **Ключевые слова:**

- алгебра логики
- высказывание
- логическая операция
- конъюнкция
- дизъюнкция
- отрицание
- логическое выражение
- таблица истинности
- законы логики

#### **1.3.1. Высказывание**

Алгебра в широком смысле этого слова — наука об общих операциях, аналогичных сложению и умножению, которые могут выполняться над разнообразными математическими объектами. Многие математические объекты (целые и рациональные числа, многочлены, векторы, множества) вы изучаете в школьном курсе алгебры, где знакомитесь с такими разделами математики, как алгебра чисел, алгебра многочленов, алгебра множеств и т. д.

Для информатики важен раздел математики, называемый алгеброй логики; объектами алгебры логики являются высказывания.



---

**Высказывание** — это предложение на любом языке, содержание которого можно однозначно определить как истинное или ложное.

---

Например, относительно предложений *«Великий русский учёный М. В. Ломоносов родился в 1711 году»* и *«Two plus six is eight»* мож-

но однозначно сказать, что они истинны. Предложение «*Зимой воробьи впадают в спячку*» ложно. Следовательно, эти предложения являются высказываниями.

В русском языке высказывания выражаются повествовательными предложениями. Но не всякое повествовательное предложение является высказыванием.



Например, предложение «*Это предложение является ложным*» не является высказыванием, так как относительно него нельзя сказать, истинно оно или ложно, без того чтобы не получить противоречие. Действительно, если принять, что предложение истинно, то это противоречит сказанному. Если же принять, что предложение ложно, то отсюда следует, что оно истинно.

Относительно предложения «*Компьютерная графика — самая интересная тема в курсе школьной информатики*» также нельзя однозначно сказать, истинно оно или ложно. Подумайте сами почему.



Побудительные и вопросительные предложения высказываниями не являются.



Например, не являются высказываниями такие предложения, как: «*Запишите домашнее задание*», «*Как пройти в библиотеку?*», «*Кто к нам пришёл?*».

Высказывания могут строиться с использованием знаков различных формальных языков — математики, физики, химии и т. п.



Примерами высказываний могут служить:

- 1) «*Na — металл*» (истинное высказывание);
- 2) «*Второй закон Ньютона выражается формулой  $F = m \cdot a$* » (истинное высказывание);
- 3) «*Периметр прямоугольника с длинами сторон  $a$  и  $b$  равен  $a \cdot b$* » (ложное высказывание).

Не являются высказываниями числовые выражения, но из двух числовых выражений можно составить высказывание, соединив их знаками равенства или неравенства. Например:

- 1) « $3 + 5 = 2 \cdot 4$ » (истинное высказывание);
- 2) « $\text{II} + \text{VI} > \text{VIII}$ » (ложное высказывание).

Не являются высказываниями и равенства или неравенства, содержащие переменные. Например, предложение « $X < 12$ » становится высказыванием только при замене переменной каким-либо конкретным значением: « $5 < 12$ » — истинное высказывание; « $12 < 12$ » — ложное высказывание.

Обоснование истинности или ложности высказываний решается теми науками, к сфере которых они относятся. Алгебра логики отвлекается от смысловой содержательности высказываний. Её интересует только то, истинно или ложно данное высказывание. В алгебре логики высказывания обозначают буквами и называют логическими переменными. При этом если высказывание истинно, то значение соответствующей ему логической переменной обозначают единицей ( $A = 1$ ), а если ложно — нулём ( $B = 0$ ). 0 и 1, обозначающие значения логических переменных, называются логическими значениями.



**Алгебра логики** определяет правила записи, упрощения и преобразования высказываний и вычисления их значений.

Опирируя логическими переменными, которые могут быть равны только 0 или 1, алгебра логики позволяет свести обработку информации к операциям с двоичными данными. Именно аппарат алгебры логики положен в основу компьютерных устройств хранения и обработки данных. С применением элементов алгебры логики вы будете встречаться и во многих других разделах информатики.

## 1.3.2. Логические операции

Высказывания бывают простые и сложные. Высказывание называется простым, если никакая его часть сама не является высказыванием. Сложные (составные) высказывания строятся из простых с помощью логических операций.

Рассмотрим основные логические операции, определённые над высказываниями. Все они соответствуют связкам, употребляемым в естественном языке.

Название логической операции	Логическая связка
Инверсия	«не»; «неверно, что»
Конъюнкция	«и»; «а»; «но»; «хотя»
Дизъюнкция	«или»



### Конъюнкция

Рассмотрим два высказывания:  $A = \text{«Основоположником алгебры логики является Джордж Буль»}$ ,  $B = \text{«Исследования Клода Шеннона позволили применить алгебру логики в вычислительной технике»}$ . Очевидно, новое высказывание *«Основоположником алгебры логики является Джордж Буль, и исследования Клода Шеннона позволили применить алгебру логики в вычислительной технике»* истинно только в том случае, когда одновременно истинны оба исходных высказывания.

Самостоятельно установите истинность или ложность трёх рассмотренных выше высказываний.




---

**Конъюнкция** — логическая операция, ставящая в соответствие двум высказываниям новое высказывание, являющееся истинным тогда и только тогда, когда оба исходных высказывания истинны.

---



Для записи конъюнкции используются следующие знаки: И,  $\wedge$ ,  $\cdot$ ,  $\&$ . Например:  $A$  И  $B$ ,  $A \wedge B$ ,  $A \cdot B$ ,  $A\&B$ .

Конъюнкцию можно описать в виде таблицы, которую называют **таблицей истинности**:

$A$	$B$	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

В таблице истинности перечисляются все возможные значения исходных высказываний (столбцы  $A$  и  $B$ ), причём соответствующие им двоичные числа, как правило, располагают в порядке возрастания: 00, 01, 10, 11. В последнем столбце записан результат выполнения логической операции для соответствующих операндов.

Конъюнкцию также называют логическим умножением. Подумайте почему.



## Дизъюнкция

Рассмотрим два высказывания:  $A = \langle \text{Идея использования в логике математической символики принадлежит Готфриду Вильгельму Лейбницу} \rangle$ ,  $B = \langle \text{Лейбниц является основоположником бинарной арифметики} \rangle$ . Очевидно, новое высказывание  $\langle \text{Идея использования в логике математической символики принадлежит Готфриду Вильгельму Лейбницу или Лейбниц является основоположником бинарной арифметики} \rangle$  ложно только в том случае, когда одновременно ложны оба исходных высказывания.



Самостоятельно установите истинность или ложность трёх рассмотренных выше высказываний.



**Дизъюнкция** — логическая операция, которая двум высказываниям ставит в соответствие новое высказывание, являющееся ложным тогда и только тогда, когда оба исходных высказывания ложны.

Для записи дизъюнкции используются следующие знаки: ИЛИ,  $\vee$ ,  $|$ ,  $+$ . Например:  $A$  ИЛИ  $B$ ,  $A \vee B$ ,  $A|B$ ,  $A+B$ .

Дизъюнкция определяется следующей таблицей истинности:

$A$	$B$	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1



Дизъюнкцию также называют логическим сложением. Подумайте почему.

## Инверсия



**Инверсия** — логическая операция, которая высказыванию ставит в соответствие новое высказывание, значение которого противоположно исходному.

Для записи инверсии используются следующие знаки: НЕ,  $\neg$ ,  $\bar{\phantom{A}}$ . Например: НЕ  $A$ ,  $\neg A$ ,  $\bar{A}$ .

Инверсия определяется следующей таблицей истинности:

A	$\bar{A}$
0	1
1	0

Инверсию также называют логическим отрицанием.

Отрицанием высказывания «У меня дома есть компьютер» будет высказывание «Неверно, что у меня дома есть компьютер» или, что в русском языке то же самое, «У меня дома нет компьютера». Отрицанием высказывания «Я не знаю китайский язык» будет высказывание «Неверно, что я не знаю китайский язык» или, что в русском языке одно и то же, «Я знаю китайский язык». Отрицанием высказывания «Все юноши 8-х классов — отличники» является высказывание «Неверно, что все юноши 8-х классов — отличники», другими словами, «Не все юноши 8-х классов — отличники».

Таким образом, при построении отрицания к простому высказыванию либо используется речевой оборот «неверно, что ...», либо отрицание строится к сказуемому, тогда к соответствующему глаголу добавляется частица «не».

Любое сложное высказывание можно записать в виде логического выражения — выражения, содержащего логические переменные, знаки логических операций и скобки. Логические операции в логическом выражении выполняются в следующей очерёдности: инверсия, конъюнкция, дизъюнкция. Изменить порядок выполнения операций можно с помощью расстановки скобок.

Логические операции при выполнении имеют следующий приоритет: инверсия, конъюнкция, дизъюнкция.

**Пример 1.** Пусть  $A =$  «На web-странице встречается слово "крейсер"»,  $B =$  «На web-странице встречается слово "линкор"». Рассматривается некоторый сегмент сети Интернет, содержащий 5 000 000 web-страниц. В нём высказывание  $A$  истинно для 4800 страниц, высказывание  $B$  — для 4500 страниц, а высказывание  $A \vee B$  — для 7000 страниц. Для какого количества web-страниц в этом случае будут истинны следующие выражения и высказывание?



- а) НЕ ( $A$  ИЛИ  $B$ );
- б)  $A$  &  $B$ ;
- в) На web-странице встречается слово "крейсер" И не встречается слово "линкор".

*Решение.* Изобразим множество всех web-страниц рассматриваемого сектора сети Интернет кругом, внутри которого разместим два круга: одному из них соответствует множество web-страниц, где истинно высказывание  $A$ , второму — где истинно высказывание  $B$  (рис. 1.3).

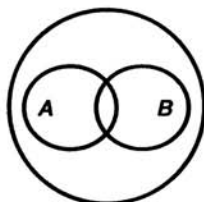


Рис. 1.3. Графическое изображение множеств web-страниц

Изобразим графически множества web-страниц, для которых истинны выражения и высказывание а) – в) (рис. 1.4).

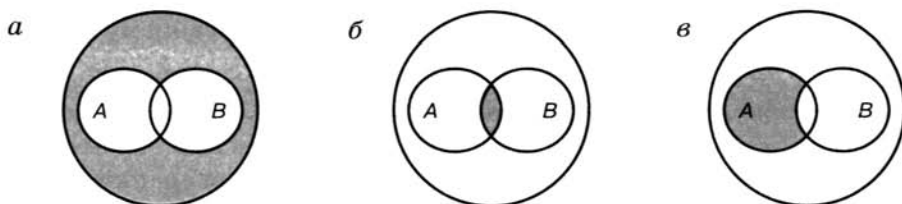


Рис. 1.4. Графические изображения множеств web-страниц, для которых истинны выражения и высказывание а) – в)

Построенные схемы помогут нам ответить на вопросы, содержащиеся в задании.

Выражение  $A$  ИЛИ  $B$  истинно для 7000 web-страниц, а всего страниц 5 000 000. Следовательно, выражение  $A$  ИЛИ  $B$  ложно для 4 993 000 web-страниц. Иначе говоря, для 4 993 000 web-страниц истинно выражение НЕ ( $A$  ИЛИ  $B$ ).

Выражение  $A \vee B$  истинно для тех web-страниц, где истинно  $A$  (4800), а также тех web-страниц, где истинно  $B$  (4500). Если бы все web-страницы были различны, то выражение  $A \vee B$  было бы истинно для 9300 (4800 + 4500) web-страниц. Но, согласно условию, таких web-страниц всего 7000. Это значит, что на 2300 (9300 – 7000)

web-страницах встречаются оба слова одновременно. Следовательно, выражение  $A \& B$  истинно для 2300 web-страниц.

Чтобы выяснить, для скольких web-страниц истинно высказывание  $A$  и одновременно ложно высказывание  $B$ , следует из 4800 вычесть 2300. Таким образом, высказывание «На web-странице встречается слово "крейсер" И не встречается слово "линкор"» истинно на 2500 web-страницах.

Самостоятельно запишите логическое выражение, соответствующее рассмотренному выше высказыванию.



На сайте Федерального центра информационно-образовательных ресурсов (<http://fcoir.edu.ru/>) размещён информационный модуль «Высказывание. Простые и сложные высказывания. Основные логические операции». Знакомство с этим ресурсом позволит вам расширить представления по изучаемой теме.



### 1.3.3. Построение таблиц истинности для логических выражений

Для логического выражения можно построить таблицу истинности, показывающую, какие значения принимает выражение при всех наборах значений входящих в него переменных. Для построения таблицы истинности следует:

- 1) подсчитать  $n$  — число переменных в выражении;
- 2) подсчитать общее число логических операций в выражении;
- 3) установить последовательность выполнения логических операций с учётом скобок и приоритетов;
- 4) определить число столбцов в таблице: число переменных + число операций;
- 5) заполнить шапку таблицы, включив в неё переменные и операции в соответствии с последовательностью, установленной в п. 3;
- 6) определить число строк в таблице (не считая шапки таблицы):  $m = 2^n$ ;
- 7) выписать наборы входных переменных с учётом того, что они представляют собой ряд целых  $n$ -разрядных двоичных чисел от 0 до  $2^n - 1$ ;
- 8) провести заполнение таблицы по столбцам, выполняя логические операции в соответствии с установленной последовательностью.



Построим таблицу истинности для логического выражения  $A \vee A \& B$ . В нём две переменные, две операции, причём сначала выполняется конъюнкция, а затем — дизъюнкция. Всего в таблице будет четыре столбца:

$A$	$B$	$A \& B$	$A \vee A \& B$
-----	-----	----------	-----------------

Наборы входных переменных — это целые числа от 0 до 3, представленные в двухразрядном двоичном коде: 00, 01, 10, 11.

Заполненная таблица истинности имеет вид:

$A$	$B$	$A \& B$	$A \vee A \& B$
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Обратите внимание, что последний столбец (результат) совпал со столбцом  $A$ . В таком случае говорят, что логическое выражение  $A \vee A \& B$  равносильно логической переменной  $A$ .

### 1.3.4. Свойства логических операций

Рассмотрим основные свойства логических операций, называемые также законами алгебры логики.

#### 1. Переместительный (коммутативный) закон:

- для логического умножения:  
 $A \& B = B \& A$ ;

- для логического сложения:  
 $A \vee B = B \vee A$ .

#### 2. Сочетательный (ассоциативный) закон:

- для логического умножения:  
 $(A \& B) \& C = A \& (B \& C)$ ;

- для логического сложения:  
 $(A \vee B) \vee C = A \vee (B \vee C)$ .

При одинаковых знаках операций скобки можно ставить произвольно или вообще опускать.

**3. Распределительный (дистрибутивный) закон:**

- для логического умножения:

$$A \& (B \vee C) = (A \& B) \vee (A \& C);$$

- для логического сложения:

$$A \vee (B \& C) = (A \vee B) \& (A \vee C).$$

**4. Закон двойного отрицания:**

$$\overline{\overline{A}} = A.$$

Двойное отрицание исключает отрицание.

**5. Закон исключённого третьего:**

- для логического умножения:

$$A \& \overline{A} = 0;$$

- для логического сложения:

$$A \vee \overline{A} = 1.$$

Из двух противоречивых высказываний об одном и том же предмете одно всегда истинно, а второе — ложно, третьего не дано.

**6. Закон повторения:**

- для логического умножения:

$$A \& A = A;$$

- для логического сложения:

$$A \vee A = A.$$

**7. Законы операций с 0 и 1:**

- для логического умножения:

$$A \& 0 = 0; A \& 1 = A;$$

- для логического сложения:

$$A \vee 0 = A; A \vee 1 = 1.$$

**8. Законы общей инверсии:**

- для логического умножения:

$$\overline{A \& B} = \overline{A} \vee \overline{B};$$

- для логического сложения:

$$\overline{A \vee B} = \overline{A} \& \overline{B}.$$

Законы алгебры логики могут быть доказаны с помощью таблиц истинности.

Докажем распределительный закон для логического сложения:

$$A \vee (B \& C) = (A \vee B) \& (A \vee C).$$



A	B	C	$B \& C$	$A \vee (B \& C)$	$A \vee B$	$A \vee C$	$(A \vee B) \& (A \vee C)$
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

Совпадение значений в столбцах, соответствующих логическим выражениям в левой и правой частях равенства, доказывает справедливость распределительного закона для логического сложения.



**Пример 2.** Найдём значение логического выражения  $(X < 3) \& \overline{(X < 2)}$  для числа  $X = 0$ .



**Решение.** При  $X = 0$  получаем следующее логическое выражение:  $(0 < 3) \& \overline{(0 < 2)}$ . Так как логические выражения  $0 < 3$ ,  $0 < 2$  истинны, то, подставив их значения в логическое выражение, получаем:  $1 \& \bar{1} = 1 \& 0 = 0$ .

### 1.3.5. Решение логических задач

Рассмотрим несколько способов решения логических задач.



**Задача 1.** Коля, Вася и Серёжа гостили летом у бабушки. Однажды один из мальчиков нечаянно разбил любимую бабушкину вазу. На вопрос, кто разбил вазу, они дали такие ответы:



Серёжа: 1) Я не разбивал. 2) Вася не разбивал.

Вася: 3) Серёжа не разбивал. 4) Вазу разбил Коля.

Коля: 5) Я не разбивал. 6) Вазу разбил Серёжа.

Бабушка знала, что один из её внуков, назовём его правдивым, оба раза сказал правду; второй, назовём его шутником, оба раза сказал неправду; третий, назовём его хитрецом, один раз сказал правду, а другой раз — неправду. Назовите имена правдивого, шутника и хитреца. Кто из внуков разбил вазу?

**Решение.** Пусть  $K$  = «Коля разбил вазу»,  $B$  = «Вася разбил вазу»,  $C$  = «Серёжа разбил вазу». Для решения задачи можно соста-



вить таблицу истинности, в которой представить высказывания каждого мальчика. Так как ваза разбита одним внуком, то чтобы выяснить, кто именно это сделал, достаточно фрагмента таблицы истинности, содержащего наборы значений входных переменных: 001, 010, 100.

К	В	С	Утверждения Серёжи		Утверждения Васи		Утверждения Коли	
			$\bar{C}$	$\bar{B}$	$\bar{C}$	К	$\bar{K}$	С
0	0	1	0	1	0	0	1	1
0	1	0	1	0	1	0	1	0
1	0	0	1	1	1	1	0	0

Исходя из того, что знает о внуках бабушка, следует искать в таблице строку, содержащую в каком-либо порядке три комбинации значений: 00 (слова шутника), 11 (слова правдивого внука), 01 или 10 (слова хитреца). Такая строка отмечена галочкой. Согласно этой строке, вазу разбил Серёжа, он же оказался хитрецом. Шутником оказался Вася. Имя правдивого внука — Коля.

**Задача 2.** В соревнованиях по гимнастике участвуют Алла, Валя, Сима и Даша. Болельщики высказали предположения о возможных победителях:

- 1) Сима будет первой, Валя — второй;
- 2) Сима будет второй, Даша — третьей;
- 3) Алла будет второй, Даша — четвёртой.

По окончании соревнований оказалось, что в каждом из предположений только одно из высказываний истинно, другое ложно. Какое место на соревнованиях заняла каждая из девушек, если все они оказались на разных местах?

*Решение.* Рассмотрим простые высказывания:

- $S_1$  = «Сима заняла первое место»;
- $B_2$  = «Валя заняла второе место»;
- $S_2$  = «Сима заняла второе место»;
- $D_3$  = «Даша заняла третье место»;
- $A_2$  = «Алла заняла второе место»;
- $D_4$  = «Даша заняла четвёртое место».

Так как в каждом из трёх предположений одно из высказываний истинно, а другое ложно, то можно заключить следующее:



$$1) C_1 + B_2 = 1, C_1 \cdot B_2 = 0;$$

$$2) C_2 + D_3 = 1, C_2 \cdot D_3 = 0;$$

$$3) A_2 + D_4 = 1, A_2 \cdot D_4 = 0.$$

Логическое произведение истинных высказываний будет истинным:

$$(C_1 + B_2) \cdot (C_2 + D_3) \cdot (A_2 + D_4) = 1.$$

На основании распределительного закона преобразуем левую часть этого выражения:

$$(C_1 \cdot C_2 + C_1 \cdot D_3 + B_2 \cdot C_2 + B_2 \cdot D_3) \cdot (A_2 + D_4) = 1.$$

Высказывание  $C_1 \cdot C_2$  означает, что Сима заняла и первое, и второе места. Согласно условию задачи, это высказывание ложно. Ложным является и высказывание  $B_2 \cdot C_2$ . Учитывая закон операций с константой 0, запишем:


$$(C_1 \cdot D_3 + B_2 \cdot D_3) \cdot (A_2 + D_4) = 1.$$

Дальнейшее преобразование левой части этого равенства и исключение заведомо ложных высказываний дают:

$$C_1 \cdot D_3 \cdot A_2 + C_1 \cdot D_3 \cdot D_4 + B_2 \cdot D_3 \cdot A_2 + B_2 \cdot D_3 \cdot D_4 = 1.$$

$$C_1 \cdot D_3 \cdot A_2 = 1.$$

Из последнего равенства следует, что  $C_1 = 1$ ,  $D_3 = 1$ ,  $A_2 = 1$ . Это означает, что Сима заняла первое место, Алла — второе, Даша — третье. Следовательно, Валя заняла четвертое место.

 Познакомьтесь с другими способами решения логических задач, а также принять участие в интернет-олимпиадах и конкурсах по их решению вы сможете на российской странице международного математического конкурса «Кенгуру» (<http://mathkang.ru/>).

На сайте <http://www.kaser.com/> вы сможете скачать демонстрационную версию очень полезной, развивающей логику и умение рассуждать логической головоломки Шерлок.

## 1.3.6. Логические элементы

Алгебра логики — раздел математики, играющий важную роль в конструировании автоматических устройств, разработке аппаратных и программных средств информационных и коммуникационных технологий.

Вы уже знаете, что любая информация может быть представлена в дискретной форме — в виде фиксированного набора отдельных значений. Устройства, которые обрабатывают такие значения (сигналы), называются дискретными. Дискретный преобразователь, который выдаёт после обработки двоичных сигналов значение одной из логических операций, называется логическим элементом.

На рисунке 1.5 приведены условные обозначения (схемы) логических элементов, реализующих логическое умножение, логическое сложение и инверсию.

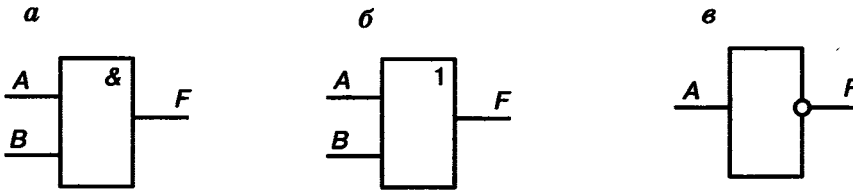


Рис. 1.5. Логические элементы

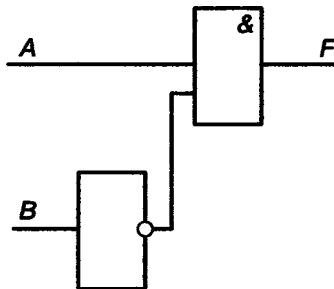
Логический элемент **И** (конъюнктор) реализует операцию логического умножения (рис. 1.5, а). Единица на выходе этого элемента появится только тогда, когда на всех входах будут единицы.

Логический элемент **ИЛИ** (дизъюнктор) реализует операцию логического сложения (рис. 1.5, б). Если хотя бы на одном входе будет единица, то на выходе элемента также будет единица.

Логический элемент **НЕ** (инвертор) реализует операцию отрицания (рис. 1.5, в). Если на входе элемента 0, то на выходе 1 и наоборот.

Компьютерные устройства, производящие операции над двоичными числами, и ячейки, хранящие данные, представляют собой электронные схемы, состоящие из отдельных логических элементов. Более подробно эти вопросы будут раскрыты в курсе информатики 10–11 классов.

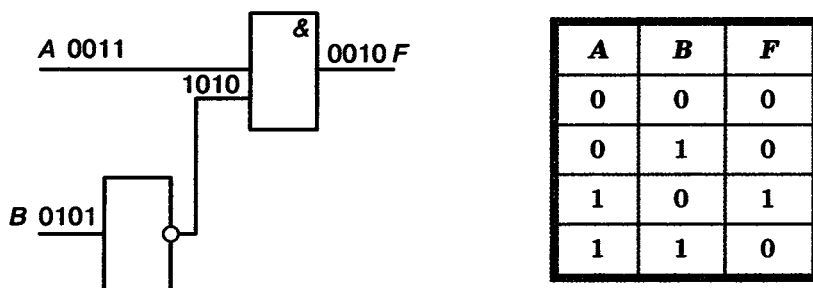
**Пример 3.** Проанализируем электронную схему, т. е. выясним, какой сигнал должен быть на выходе  $F$  при каждом возможном наборе сигналов  $A$  и  $B$  на входах.



**Решение.** Все возможные комбинации сигналов  $A$  и  $B$  на входах внесём в таблицу истинности. Проследим преобразование каждой пары сигналов при прохождении их через логические элементы



и запишем полученный результат в таблицу. Заполненная таблица истинности полностью описывает рассматриваемую электронную схему.



Таблицу истинности можно построить и по логическому выражению, соответствующему электронной схеме. Последний логический элемент в рассматриваемой схеме — конъюнктор. В него поступают сигналы от входа  $A$  и от инвертора. В свою очередь, в инвертор поступает сигнал от входа  $B$ . Таким образом,  $F = A \& \bar{B}$ .

Составить более полное представление о логических элементах и электронных схемах вам поможет работа с тренажёром «Логика» (<http://kpolyakov.narod.ru/prog/logic.htm>).

## САМОЕ ГЛАВНОЕ

Высказывание — это предложение на любом языке, содержание которого можно однозначно определить как истинное или ложное.

Основные логические операции, определённые над высказываниями: инверсия, конъюнкция, дизъюнкция.

Название логической операции	Логическая связка	Обозначение
Инверсия	«не», «неверно, что»	$\neg$ , $\bar{\quad}$
Конъюнкция	«и», «а», «но», «хотя»	$\&$
Дизъюнкция	«или»	$\vee$

Таблицы истинности для основных логических операций:

$A$	$\bar{A}$
0	1
1	0

$A$	$B$	$A \& B$	$A \vee B$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

При вычислении логических выражений сначала выполняются действия в скобках. Приоритет выполнения логических операций:  $\neg$ ,  $\&$ ,  $\vee$ .

Вопросы и задания

- Ознакомьтесь с материалами презентации к параграфу, содержащейся в электронном приложении к учебнику. Дополняет ли презентация информацию, содержащуюся в тексте параграфа?
- Объясните, почему следующие предложения не являются высказываниями.
  - Какого цвета этот дом?
  - Число  $X$  не превосходит единицы.
  - $4X + 3$ .
  - Посмотрите в окно.
  - Пейте томатный сок!
  - Эта тема скучна.
  - Рикки Мартин — самый популярный певец.
  - Вы были в театре?
- Приведите по одному примеру истинных и ложных высказываний из биологии, географии, информатики, истории, математики, литературы.
- В следующих высказываниях выделите простые высказывания, обозначив каждое из них буквой; запишите с помощью букв и знаков логических операций каждое составное высказывание.
  - Число 376 чётное и трёхзначное.
  - Зимой дети катаются на коньках или на лыжах.
  - Новый год мы встретим на даче или на Красной площади.



- 4) Неверно, что Солнце движется вокруг Земли.
- 5) Земля имеет форму шара, который из космоса кажется голубым.
- 6) На уроке математики старшеклассники отвечали на вопросы учителя, а также писали самостоятельную работу.
5. Постройте отрицания следующих высказываний.
- 1) Сегодня в театре идёт опера «Евгений Онегин».
- 2) Каждый охотник желает знать, где сидит фазан.
- 3) Число 1 есть простое число.
- 4) Натуральные числа, оканчивающиеся цифрой 0, не являются простыми числами.
- 5) Неверно, что число 3 не является делителем числа 198.
- 6) Коля решил все задания контрольной работы.
- 7) Во всякой школе некоторые ученики интересуются спортом.
- 8) Некоторые млекопитающие не живут на суше.
6. Пусть  $A = \text{«Ане нравятся уроки математики»}$ , а  $B = \text{«Ане нравятся уроки химии»}$ . Выразите следующие формулы на обычном языке:

- 1)  $\overline{A \ \& \ B}$ ;      4)  $A \vee \overline{B}$ ;      7)  $\overline{\overline{A \ \& \ B}}$ ;  
 2)  $\overline{A \ \& \ B}$ ;      5)  $\overline{A \ \vee \ B}$ ;      8)  $\overline{\overline{A \ \vee \ B}}$ ;  
 3)  $A \ \& \ \overline{B}$ ;      6)  $\overline{A \ \vee \ B}$ ;      9)  $(A \ \& \ B)$ .

7. Некоторый сегмент сети Интернет состоит из 1000 сайтов. Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов этого сегмента. Вот её фрагмент:

Ключевое слово	Количество сайтов, для которых данное слово является ключевым
сомики	250
меченосцы	200
гулли	500

По запросу *сомики & гулли* было найдено 0 сайтов, по запросу *сомики & меченосцы* — 20 сайтов, а по запросу *меченосцы & гулли* — 10 сайтов.

Сколько сайтов будет найдено по запросу *сомики | меченосцы | гулли*?

Для скольких сайтов рассматриваемого сегмента ложно высказывание «Сомики — ключевое слово сайта ИЛИ меченосцы — ключевое слово сайта ИЛИ гуппи — ключевое слово сайта»?

8. Постройте таблицы истинности для следующих логических выражений:



1)  $B \& (A \vee B)$ ;

2)  $A \& (B \vee \overline{B})$ ;

3)  $A \& (A \vee B \vee C)$ ;

4)  $A \vee B \vee \overline{C}$ .

9. Проведите доказательство рассмотренных в параграфе логических законов с помощью таблиц истинности.



10. Даны три числа в десятичной системе счисления:  $A = 23$ ,  $B = 19$ ,  $C = 26$ . Переведите  $A$ ,  $B$  и  $C$  в двоичную систему счисления и выполните поразрядно логические операции  $(A \vee B) \& C$ . Ответ дайте в десятичной системе счисления.



11. Найдите значения выражений:



1)  $(1 \vee 1) \vee (1 \vee 0)$ ;

2)  $((1 \vee 0) \vee 1) \vee 1$ ;

3)  $(0 \& 1) \& 1$ ;

4)  $1 \& (1 \& 1) \& 1$ ;

5)  $((1 \vee 0) \& (1 \& 1)) \& (0 \vee 1)$ ;

6)  $((1 \& 1) \vee 0) \& (0 \vee 1)$ ;

7)  $((0 \& 0) \vee 0) \& (1 \vee 1)$ ;

8)  $(A \vee 1) \vee (B \vee 0)$ ;

9)  $((1 \& A) \vee (B \& 0)) \vee 1$ ;

10)  $1 \vee A \& 0$ .

12. Найдите значение логического выражения  $\overline{(X < 3)} \& \overline{(X < 2)}$  для указанных значений числа  $X$ :



1) 1

2) 2

3) 3

4) 4

13. Пусть  $A =$  «Первая буква имени — гласная»,  $B =$  «Четвёртая буква имени согласная». Найдите значение логического выражения  $\overline{A} \vee B$  для следующих имён:



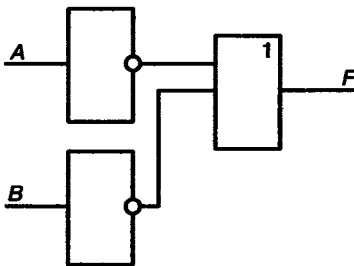
1) ЕЛЕНА

2) ВАДИМ

3) АНТОН

4) ФЁДОР

14. Разбирается дело Джона, Брауна и Смита. Известно, что один из них нашёл и утаил клад. На следствии каждый из подозреваемых сделал два заявления:  
Смит: «Я не делал этого. Браун сделал это».  
Джон: «Браун не виновен. Смит сделал это».  
Браун: «Я не делал этого. Джон не делал этого».  
Суд установил, что один из них дважды солгал, другой дважды сказал правду, третий один раз солгал, один раз сказал правду. Кто из подозреваемых должен быть оправдан?
15. Алёша, Боря и Гриша нашли в земле старинный сосуд. Рассматривая удивительную находку, каждый высказал по два предположения:  
1) Алёша: «Это сосуд греческий и изготовлен в V веке».  
2) Боря: «Это сосуд финикийский и изготовлен в III веке».  
3) Гриша: «Это сосуд не греческий и изготовлен в IV веке».  
Учитель истории сказал ребятам, что каждый из них прав только в одном из двух предположений. Где и в каком веке изготовлен сосуд?
16. Выясните, какой сигнал должен быть на выходе электронной схемы при каждом возможном наборе сигналов на входах. Составьте таблицу работы схемы. Каким логическим выражением описывается схема?





## Тестовые задания для самоконтроля



1. Совокупность знаков, с помощью которых записываются числа, называется:
  - а) системой счисления
  - б) цифрами системы счисления
  - в) алфавитом системы счисления
  - г) основанием системы счисления
2. Чему равен результат сложения двух чисел, записанных римскими цифрами:  $MCM + LXVIII$ ?
  - а) 1168
  - б) 1968
  - в) 2168
  - г) 1153
3. Число 301011 может существовать в системах счисления с основаниями:
  - а) 2 и 10
  - б) 4 и 3
  - в) 4 и 8
  - г) 2 и 4
4. Двоичное число 100110 в десятичной системе счисления записывается как:
  - а) 36
  - б) 38
  - в) 37
  - г) 46



5. В классе  $110010_2\%$  девочек и  $1010_2$  мальчиков. Сколько учеников в классе?
- а) 10
  - б) 20
  - в) 30
  - г) 40
6. Сколько цифр 1 в двоичном представлении десятичного числа 15?
- а) 1
  - б) 2
  - в) 3
  - г) 4
7. Чему равен результат сложения чисел  $110_2$  и  $12_8$ ?
- а)  $6_{10}$
  - б)  $10_{10}$
  - в)  $10000_2$
  - г)  $17_8$
8. Ячейка памяти компьютера состоит из однородных элементов, называемых:
- а) кодами
  - б) разрядами
  - в) цифрами
  - г) коэффициентами
9. Количество разрядов, занимаемых двухбайтовым числом, равно:
- а) 8
  - б) 16
  - в) 32
  - г) 64
10. В знаковый разряд ячейки для отрицательных чисел заносится:
- а) +
  - б) -
  - в) 0
  - г) 1

11. Вещественные числа представляются в компьютере в:
- а) естественной форме
  - б) развёрнутой форме
  - в) экспоненциальной форме с нормализованной мантиссой
  - г) виде обыкновенной дроби
12. Какое предложение не является высказыванием?
- а) Никакая причина не извиняет невежливость.
  - б) Обязательно стань отличником.
  - в) Рукописи не горят.
  - г)  $1011_2 = 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$
13. Какое высказывание является ложным?
- а) Знаком  $\vee$  обозначается логическая операция ИЛИ.
  - б) Логическую операцию ИЛИ также называют логическим сложением.
  - в) Дизъюнкцию также называют логическим сложением.
  - г) Знаком  $\vee$  обозначается логическая операция конъюнкция.
14. Для какого из указанных значений числа  $X$  истинно высказывание  $((X < 5) \vee (X < 3)) \wedge ((X < 2) \vee (X < 1))$ ?
- а) 1
  - б) 2
  - в) 3
  - г) 4
15. Для какого символического выражения верно высказывание: «НЕ (Первая буква согласная) И НЕ (Вторая буква гласная)»?
- а) abcde
  - б) bcade
  - в) babas
  - г) cabab
16. Некоторый сегмент сети Интернет состоит из 1000 сайтов. Поисковый сервер в автоматическом режиме составил таблицу ключевых слов для сайтов этого сегмента. Вот её фрагмент:

Ключевое слово	Количество сайтов, для которых данное слово является ключевым
сканер	200
принтер	250
монитор	450

Сколько сайтов будет найдено по запросу *принтер | сканер | монитор*, если по запросу *принтер | сканер* было найдено 450 сайтов, по запросу *принтер & монитор* — 40, а по запросу *сканер & монитор* — 50?

- а) 900
- б) 540
- в) 460
- г) 810



17. Какому логическому выражению соответствует следующая таблица истинности?

<i>A</i>	<i>B</i>	<i>F</i>
0	0	1
0	1	1
1	0	1
1	1	0

- а)  $A \& B$
- б)  $A \vee B$
- в)  $\overline{A \& B}$
- г)  $A \& \overline{B}$



18. Когда сломался компьютер, его хозяин сказал: «Оперативная память не могла выйти из строя». Сын хозяина компьютера предположил, что вышел из строя процессор, а жёсткий диск исправен. Пришедший специалист по обслуживанию сказал, что, скорее всего, с процессором всё в порядке, а оперативная память неисправна. В результате оказалось, что двое из них сказали всё верно, а третий — всё неверно. Что же сломалось?

- а) оперативная память
- б) процессор
- в) жёсткий диск
- г) процессор и оперативная память



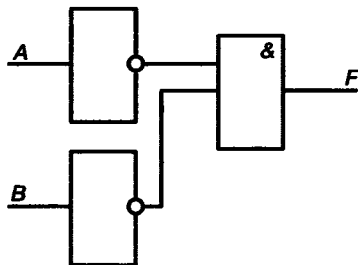
19. На перекрёстке произошло дорожно-транспортное происшествие, в котором участвовали автобус (*A*), грузовик (*Г*), легковой автомобиль (*Л*) и маршрутное такси (*М*). Свидетели происшествия дали следующие показания. Первый свидетель считал, что первым на перекрёсток выехал автобус, а маршрутное так-



си было вторым. Другой свидетель полагал, что последним на перекрёсток выехал легковой автомобиль, а вторым был грузовик. Третий свидетель уверял, что автобус выехал на перекрёсток вторым, а следом за ним — легковой автомобиль. В результате оказалось, что каждый из свидетелей был прав только в одном из своих утверждений. В каком порядке выехали машины на перекрёсток? В вариантах ответов перечислены подряд без пробелов первые буквы названий транспортных средств в порядке их выезда на перекрёсток:

- а) АМЛГ
- б) АГЛМ
- в) ГЛМА
- г) МЛГА

20. Какое логическое выражение соответствует следующей схеме?



- а)  $A \& B$
- б)  $A \vee B$
- в)  $\overline{A \& B}$
- г)  $\overline{A \& \overline{B}}$

Для проверки знаний и умений по теме «Математические основы информатики» вы можете воспользоваться интерактивным тестом к главе 1, содержащимся в электронном приложении к учебнику.



# Глава 2

## ОСНОВЫ АЛГОРИТМИЗАЦИИ

### § 2.1

#### Алгоритмы и исполнители

**Ключевые слова:**

- алгоритм
- свойства алгоритма
  - дискретность
  - понятность
  - определённости
  - результативность
  - массовость
- исполнитель
- характеристики исполнителя
  - круг решаемых задач
  - среда
  - режим работы
  - система команд
- формальное исполнение алгоритма

#### 2.1.1. Понятие алгоритма

Каждый человек в повседневной жизни, в учёбе или на работе решает огромное количество задач самой разной сложности. Сложные задачи требуют длительных размышлений для нахождения решения; простые и привычные задачи человек решает не задумываясь, автоматически. В большинстве случаев решение каждой задачи можно разбить на простые этапы (шаги). Для многих таких задач (установка программного обеспечения, сборка шкафа, создание сай-

та, эксплуатация технического устройства, покупка авиабилета через Интернет и т. д.) уже разработаны и предлагаются пошаговые инструкции, при последовательном выполнении которых можно прийти к желаемому результату.

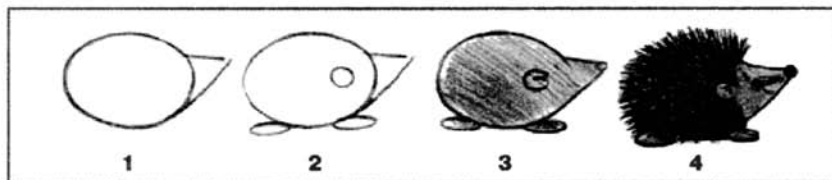
**Пример 1.** Задача «Найти среднее арифметическое двух чисел» решается в три шага:

- 1) задумать два числа;
- 2) сложить два задуманных числа;
- 3) полученную сумму разделить на 2.

**Пример 2.** Задача «Внести деньги на счёт телефона» подразделяется на следующие шаги:

- 1) подойти к терминалу по оплате платежей;
- 2) выбрать оператора связи;
- 3) ввести номер телефона;
- 4) проверить правильность введённого номера;
- 5) вставить денежную купюру в купюроприёмник;
- 6) дождаться сообщения о зачислении денег на счёт;
- 7) получить чек.

**Пример 3.** Этапы решения задачи «Нарисовать весёлого ёжика» представлены графически:



Нахождение среднего арифметического, внесение денег на телефонный счёт и рисование ежа — на первый взгляд совершенно разные процессы. Но у них есть общая черта: каждый из этих процессов описывается последовательностями кратких указаний, точное следование которым позволяет получить требуемый результат. Последовательности указаний, приведённые в примерах 1–3, являются алгоритмами решения соответствующих задач. Исполнитель этих алгоритмов — человек.

Алгоритм может представлять собой описание некоторой последовательности вычислений (пример 1) или шагов нематематического характера (примеры 2–3). Но в любом случае перед его разработкой должны быть чётко определены начальные условия (исходные

данные) и то, что предстоит получить (результат). Можно сказать, что алгоритм — это описание последовательности шагов в решении задачи, приводящих от исходных данных к требуемому результату.

В общем виде схему работы алгоритма можно представить следующим образом (рис. 2.1).



Рис. 2.1. Общая схема работы алгоритма

Алгоритмами являются изучаемые в школе правила сложения, вычитания, умножения и деления чисел, многие грамматические правила, правила геометрических построений и т. д.



Анимации «Работа с алгоритмом» (193576), «Наибольший общий делитель» (170363), «Наименьшее общее кратное» (170390) помогут вам вспомнить некоторые алгоритмы, изученные на уроках русского языка и математики (<http://sc.edu.ru/>).



**Пример 4.** Некоторый алгоритм приводит к тому, что из одной цепочки символов получается новая цепочка следующим образом:

1. Вычисляется длина (в символах) исходной цепочки символов.
2. Если длина исходной цепочки нечётна, то к исходной цепочке справа приписывается цифра 1, иначе цепочка не изменяется.
3. Символы попарно меняются местами (первый — со вторым, третий — с четвёртым, пятый — с шестым и т. д.).
4. Справа к полученной цепочке приписывается цифра 2.

Получившаяся таким образом цепочка является результатом работы алгоритма.

Так, если исходной была цепочка  $A\#B$ , то результатом работы алгоритма будет цепочка  $\#A1B2$ , а если исходной цепочкой была  $ABV@$ , то результатом работы алгоритма будет цепочка  $BA@B2$ .

### 2.1.2. Исполнитель алгоритма

Каждый алгоритм предназначен для определённого исполнителя.



**Исполнитель** — это некоторый объект (человек, животное, техническое устройство), способный выполнять определённый набор команд.



Различают формальных и неформальных исполнителей. Формальный исполнитель одну и ту же команду всегда выполняет одинаково. Неформальный исполнитель может выполнять команду по-разному.

Рассмотрим более подробно множество формальных исполнителей. Формальные исполнители необычайно разнообразны, но для каждого из них можно указать следующие характеристики: круг решаемых задач (назначение), среду, систему команд и режим работы.

**Круг решаемых задач.** Каждый исполнитель создаётся для решения некоторого круга задач — построения цепочек символов, выполнения вычислений, построения рисунков на плоскости и т. д.

**Среда исполнителя.** Область, обстановку, условия, в которых действует исполнитель, принято называть средой данного исполнителя. Исходные данные и результаты любого алгоритма всегда принадлежат среде того исполнителя, для которого предназначен алгоритм.

**Система команд исполнителя.** Предписание исполнителю о выполнении отдельного законченного действия называется командой. Совокупность всех команд, которые могут быть выполнены некоторым исполнителем, образует систему команд данного исполнителя (СКИ). Алгоритм составляется с учётом возможностей конкретного исполнителя, иначе говоря, в системе команд исполнителя, который будет его выполнять.

**Режимы работы исполнителя.** Для большинства исполнителей предусмотрены *режимы непосредственного управления* и *программного управления*. В первом случае исполнитель ожидает команд от человека и каждую поступившую команду немедленно выполняет. Во втором случае исполнителю сначала задаётся полная последовательность команд (программа), а затем он выполняет все эти команды в автоматическом режиме. Ряд исполнителей работает только в одном из названных режимов.

Рассмотрим примеры исполнителей.

**Пример 5.** Исполнитель **Черепашка** перемещается на экране компьютера, оставляя след в виде линии. Система команд Черепашки состоит из следующих команд:

Вперёд  $n$  (где  $n$  — целое число) — вызывает передвижение Черепашки на  $n$  шагов в направлении движения — в том направлении, куда развёрнуты её голова и корпус;



Направо  $m$  (где  $m$  — целое число) — вызывает изменение направления движения Черепашки на  $m$  градусов по часовой стрелке.

Запись Повтори  $k$  [ $\langle$ Команда1 $\rangle$   $\langle$ Команда2 $\rangle$  ...  $\langle$ Команда $n$  $\rangle$ ] означает, что последовательность команд в скобках повторится  $k$  раз.



Подумайте, какая фигура появится на экране после выполнения Черепашкой следующего алгоритма.

Повтори 12 [Направо 45 Вперёд 20 Направо 45]



**Пример 6.** Система команд исполнителя **Вычислитель** состоит из двух команд, которым присвоены номера:

- 1 — вычти 1
- 2 — умножь на 3

Первая из них уменьшает число на 1, вторая увеличивает число в 3 раза. При записи алгоритмов для краткости указываются лишь номера команд. Например, алгоритм 21212 означает следующую последовательность команд:

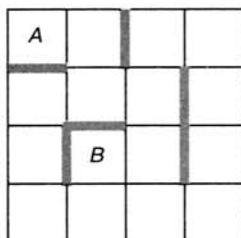
- умножь на 3
- вычти 1
- умножь на 3
- вычти 1
- умножь на 3

С помощью этого алгоритма число 1 будет преобразовано в 15:  
 $((1 \cdot 3 - 1) \cdot 3 - 1) \cdot 3 = 15$ .



**Пример 7.** Исполнитель **Робот** действует на клетчатом поле, между соседними клетками которого могут стоять стены. Робот передвигается по клеткам поля и может выполнять следующие команды, которым присвоены номера:

- 1 — вверх
- 2 — вниз
- 3 — вправо
- 4 — влево



При выполнении каждой такой команды Робот перемещается в соседнюю клетку в указанном направлении. Если же в этом направлении между клетками стоит стена, то Робот разрушается.

Что произойдёт с Роботом, если он выполнит последовательность команд 32323 (здесь цифры обозначают номера команд), начав движение из клетки *A*? Какую последовательность команд следует выполнить Роботу, чтобы переместиться из клетки *A* в клетку *B*, не разрушившись от встречи со стенами?



При разработке алгоритма:

- 1) выделяются фигурирующие в задаче объекты, устанавливаются свойства объектов, отношения между объектами и возможные действия с объектами;
- 2) определяются исходные данные и требуемый результат;
- 3) определяется последовательность действий исполнителя, обеспечивающая переход от исходных данных к результату;
- 4) последовательность действий записывается с помощью команд, входящих в систему команд исполнителя.



Можно сказать, что алгоритм — модель деятельности исполнителя алгоритмов.

### 2.1.3. Свойства алгоритма

Не любая инструкция, последовательность предписаний или план действий может считаться алгоритмом. Каждый алгоритм обязательно обладает следующими свойствами: дискретность, понятность, определённости, результативность и массовость.

**Свойство дискретности** означает, что путь решения задачи разделён на отдельные шаги (действия). Каждому действию соответствует предписание (команда). Только выполнив одну команду, исполнитель может приступить к выполнению следующей команды.

**Свойство понятности** означает, что алгоритм состоит только из команд, входящих в систему команд исполнителя, т. е. из таких команд, которые исполнитель может воспринять и по которым может выполнить требуемые действия.

**Свойство определённости** означает, что в алгоритме нет команд, смысл которых может быть истолкован исполнителем неоднозначно; недопустимы ситуации, когда после выполнения очередной команды исполнителю неясно, какую команду выполнять следующей. Благодаря этому результат алгоритма однозначно определяется набором исходных данных: если алгоритм несколько раз применяется к одному и тому же набору исходных данных, то на выходе всегда получается один и тот же результат.

**Свойство результативности** означает, что алгоритм должен обеспечивать получение результата после конечного, возможно, очень большого, числа шагов. При этом результатом считается не только

обусловленный постановкой задачи ответ, но и вывод о невозможности продолжения по какой-либо причине решения данной задачи.

Свойство массовости означает, что алгоритм должен обеспечивать возможность его применения для решения любой задачи из некоторого класса задач. Например, алгоритм нахождения корней квадратного уравнения должен быть применим к любому квадратному уравнению, алгоритм перехода улицы должен быть применим в любом месте улицы, алгоритм приготовления лекарства должен быть применим для приготовления любого его количества и т. д.



**Пример 8.** Рассмотрим один из методов нахождения всех простых чисел, не превышающих некоторое натуральное число  $n$ . Этот метод называется «решето Эратосфена» по имени предложившего его древнегреческого учёного Эратосфена (III в. до н. э.).

Для нахождения всех простых чисел, не больших заданного числа  $n$ , следуя методу Эратосфена, нужно выполнить следующие шаги:

- 1) выписать подряд все натуральные числа от 2 до  $n$  (2, 3, 4, ...,  $n$ );
- 2) заключить в рамку 2 — первое простое число;
- 3) вычеркнуть из списка все числа, делящиеся на последнее найденное простое число;
- 4) найти первое неотмеченное число (отмеченные числа — зачёркнутые числа или числа, заключённые в рамку) и заключить его в рамку — это будет очередное простое число;
- 5) повторять шаги 3 и 4 до тех пор, пока не останется неотмеченных чисел.



Более наглядное представление о методе нахождения простых чисел вы сможете получить с помощью размещённой в Единой коллекции цифровых образовательных ресурсов анимации «Решето Эратосфена» (180279).

Рассмотренная последовательность действий является алгоритмом, так как она удовлетворяет свойствам:

- дискретности — процесс нахождения простых чисел разбит на шаги;
- понятности — каждая команда понятна ученику 8 класса, выполняющему этот алгоритм;
- определённости — каждая команда трактуется и выполняется исполнителем однозначно; имеются указания об очередности выполнения команд;
- результативности — через некоторое число шагов достигается результат;

- массовости — последовательность действий применима для любого натурального  $n$ .

Рассмотренные свойства алгоритма позволяют дать более точное определение алгоритма.

---

**Алгоритм** — это предназначенное для конкретного исполнителя описание последовательности действий, приводящих от исходных данных к требуемому результату, которое обладает свойствами дискретности, понятности, определённости, результативности и массовости.

---



#### 2.1.4. Возможность автоматизации деятельности человека

Разработка алгоритма — как правило, трудоёмкая задача, требующая от человека глубоких знаний, изобретательности и больших временных затрат.

Решение задачи по готовому алгоритму требует от исполнителя только строгого следования заданным предписаниям.

**Пример 9.** Из кучки, содержащей любое, большее трёх, количество каких-либо предметов, двое играющих по очереди берут по одному или по два предмета. Выигрывает тот, кто своим очередным ходом сможет забрать все оставшиеся предметы.

Рассмотрим алгоритм, следуя которому первый игрок наверняка обеспечит себе выигрыш.

1. Если число предметов в кучке кратно 3, то уступить ход противнику, иначе начать игру, взяв 1 или 2 предмета так, чтобы осталось количество предметов, кратное 3.
2. Своим очередным ходом каждый раз дополнять число предметов, взятых соперником, до 3 (число оставшихся предметов должно быть кратно 3).

Исполнитель может не вникать в смысл того, что он делает, и не рассуждать, почему он поступает так, а не иначе, т. е. он может действовать формально. Способность исполнителя действовать формально обеспечивает возможность автоматизации деятельности человека. Для этого:

- 1) процесс решения задачи представляется в виде последовательности простейших операций;
- 2) создаётся машина (автоматическое устройство), способная выполнять эти операции в последовательности, заданной в алгоритме;



- 3) человек освобождается от рутинной деятельности, выполнение алгоритма поручается автоматическому устройству.

### САМОЕ ГЛАВНОЕ

**Исполнитель** — некоторый объект (человек, животное, техническое устройство), способный выполнять определённый набор команд.

**Формальный исполнитель** одну и ту же команду всегда выполняет одинаково. Для каждого формального исполнителя можно указать: круг решаемых задач, среду, систему команд и режим работы.

**Алгоритм** — предназначенное для конкретного исполнителя описание последовательности действий, приводящих от исходных данных к требуемому результату, которое обладает свойствами дискретности, понятности, определённости, результативности и массовости.

Способность исполнителя действовать формально обеспечивает возможность автоматизации деятельности человека.







### Вопросы и задания



1. Ознакомьтесь с материалами презентации к параграфу, содержащейся в электронном приложении к учебнику. Дополняет ли презентация информацию, содержащуюся в тексте параграфа? Какими слайдами вы могли бы дополнить презентацию?
2. Что называют алгоритмом?
3. Подберите синонимы к слову «предписание».
4. Приведите примеры алгоритмов, изучаемых вами в школе.
5. Кто может быть исполнителем алгоритма?
6. Приведите пример формального исполнителя. Приведите пример, когда человек выступает в роли формального исполнителя.
7. От чего зависит круг решаемых задач исполнителя «компьютер»?
8. Рассмотрите в качестве исполнителя текстовый процессор, имеющийся на вашем компьютере. Охарактеризуйте круг решаемых этим исполнителем задач и его среду.
9. Что такое команда, система команд исполнителя?



10. Какие команды должны быть у робота, выполняющего функции: а) кассира в магазине; б) дворника; в) охранника?
11. Перечислите основные свойства алгоритма.
12. К чему может привести отсутствие какого-либо свойства у алгоритма? Приведите примеры.
13. В чём важность возможности формального исполнения алгоритма?
14. Последовательность чисел строится по следующему алгоритму: первые два числа последовательности принимаются равными 1; каждое следующее число последовательности принимается равным сумме двух предыдущих чисел. Запишите 10 первых членов этой последовательности. Выясните, как называется эта последовательность. 
15. Некоторый алгоритм получает из одной цепочки символов новую цепочку следующим образом. Сначала записывается исходная цепочка символов, после нее записывается исходная цепочка символов в обратном порядке, затем записывается буква, следующая в русском алфавите за той буквой, которая в исходной цепочке стояла на последнем месте. Если в исходной цепочке на последнем месте стоит буква «Я», то в качестве следующей буквы записывается буква «А». Получившаяся цепочка является результатом работы алгоритма. Например, если исходная цепочка символов была «ДОМ», то результатом работы алгоритма будет цепочка «ДОММОДН». Дана цепочка символов «КОМ». Сколько букв «О» будет в цепочке символов, которая получится, если применить алгоритм к данной цепочке, а затем ещё раз применить алгоритм к результату его работы? 
16. Найдите в сети Интернет анимацию шагов алгоритма Эратосфена. С помощью алгоритма Эратосфена найдите все простые числа, не превышающие 50.
17. Что будет результатом исполнения Черепашкой (см. пример 5) алгоритма?  
Повтори 8 [Направо 45 Вперёд 45] 
18. Запишите алгоритм для исполнителя Вычислитель (см. пример 6), содержащий не более 5 команд:  
а) получения из числа 3 числа 16;  
б) получения из числа 1 числа 25. 



- 19.** Система команд исполнителя Конструктор состоит из двух команд, которым присвоены номера:

1 — приписать 2

2 — разделить на 2

По первой из них к числу приписывается справа 2, по второй число делится на 2. Как будет преобразовано число 8, если исполнитель выполнит алгоритм 22212? Составьте алгоритм в системе команд этого исполнителя, по которому число 1 будет преобразовано в число 16 (в алгоритме должно быть не более 5 команд).



- 20.** В какой клетке должен находиться исполнитель Робот (пример 7), чтобы после выполнения алгоритма 3241 в неё же и вернуться?



## § 2.2

### Способы записи алгоритмов

#### **Ключевые слова:**

- словесное описание
- построчная запись
- блок-схема
- школьный алгоритмический язык

Существуют различные способы записи алгоритмов. Основными среди них являются:

- словесные;
- графические;
- на алгоритмических языках.

Теоретические исследования нашего соотечественника Андрея Андреевича Маркова (младшего) (1903–1979), выполненные в середине прошлого века, показали, что в общем случае алгоритмы должны содержать предписания двух видов:

- 1) предписания, направленные на непосредственное преобразование информации (функциональные операторы);
- 2) предписания, определяющие дальнейшее направление действий (логические операторы).

Именно эти операторы положены в основу большинства способов записи алгоритмов.



#### 2.2.1. Словесные способы записи алгоритма

**Словесное описание.** Самой простой является запись алгоритма в виде набора высказываний на обычном разговорном языке. Сло-

весное описание имеет минимум ограничений и является наименее формализованным. Однако все разговорные языки обладают неоднозначностью, поэтому могут возникнуть различные толкования текста алгоритма, заданного таким образом. Алгоритм в словесной форме может оказаться очень объёмным и трудным для восприятия.

**Пример 1.** Словесное описание алгоритма нахождения наибольшего общего делителя (НОД) пары натуральных чисел (алгоритм Евклида).

Чтобы найти НОД двух чисел, составьте таблицу из двух столбцов и назовите столбцы  $X$  и  $Y$ . Запишите первое из заданных чисел в столбец  $X$ , а второе — в столбец  $Y$ . Если данные числа не равны, замените большее из них на результат вычитания из большего числа меньшего. Повторяйте такие замены до тех пор, пока числа не окажутся равными, после чего число из столбца  $X$  считайте искомым результатом.

**Построчная запись.** Это запись на естественном языке, но с соблюдением некоторых дополнительных правил:

- каждое предписание записывается с новой строки;
- предписания (шаги) алгоритма нумеруются;
- исполнение алгоритма происходит в порядке возрастания номеров шагов, начиная с первого (если не встречается никаких специальных указаний).

Кроме слов естественного языка предписания могут содержать математические выражения и формулы.

**Пример 2.** Построчная запись алгоритма Евклида.

1. Обозначить первое из заданных чисел  $X$ , второе обозначить  $Y$ .
2. Если  $X = Y$ , то перейти к п. 8.
3. Если  $X > Y$ , то перейти к п. 4, иначе перейти к п. 6.
4. Заменить  $X$  на  $X - Y$ .
5. Перейти к п. 2.
6. Заменить  $Y$  на  $Y - X$ .
7. Перейти к п. 2.
8. Считать  $X$  искомым результатом.

Построчная запись алгоритма позволяет избежать ряда неопределённостей; её восприятие не требует дополнительных знаний.

Вместе с тем использование построчной записи требует от человека большого внимания.

### 2.2.2. Блок-схемы

Наилучшей наглядностью обладают графические способы записи алгоритмов; самый распространённый среди них — блок-схема.

Блок-схема представляет собой графический документ, дающий представление о порядке работы алгоритма. Здесь предписания изображаются с помощью различных геометрических фигур, а последовательность выполнения шагов указывается с помощью линий, соединяющих эти фигуры. Направления линий связи *слева направо* и *сверху вниз* считаются стандартными, соответствующие им линии связи можно изображать *без стрелок*. Линии связи *справа налево* и *снизу вверх* изображаются *со стрелками*.

Рассмотрим некоторые условные обозначения, применяемые в блок-схемах.

Выполнение алгоритма всегда начинается с блока начала и оканчивается при переходе на блок конца (рис. 2.2, а). Из начального блока выходит одна линия связи; в конечный блок входит одна линия связи.

Внутри блока данных (рис. 2.2, б) перечисляются величины, значения которых должны быть введены (исходные данные) или выведены (результаты) в данном месте схемы. В блок данных входит одна линия связи, и из блока исходит одна линия связи.

В блоке обработки данных (рис. 2.2, в) содержится описание тех действий, которые должны быть выполнены при переходе на этот блок (выполнение определённой операции или группы операций, приводящее к изменению значения, формы или размещения информации). В блок обработки данных входит одна линия связи, и из блока исходит одна линия связи.

Проверка условия изображается с помощью блока принятия решения, внутри которого записывается это условие (рис. 2.2, г). В блок принятия решения входит одна линия, а выходят две линии, около которых записываются результаты проверки условия.

Комментарии (рис. 2.2, д) используются для добавления пояснительных записей, делающих блок-схему более понятной.

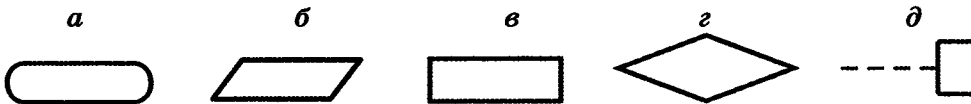


Рис. 2.2. Обозначения на блок-схемах



**Пример 3.** Запись алгоритма Евклида с помощью блок-схемы (рис. 2.3).

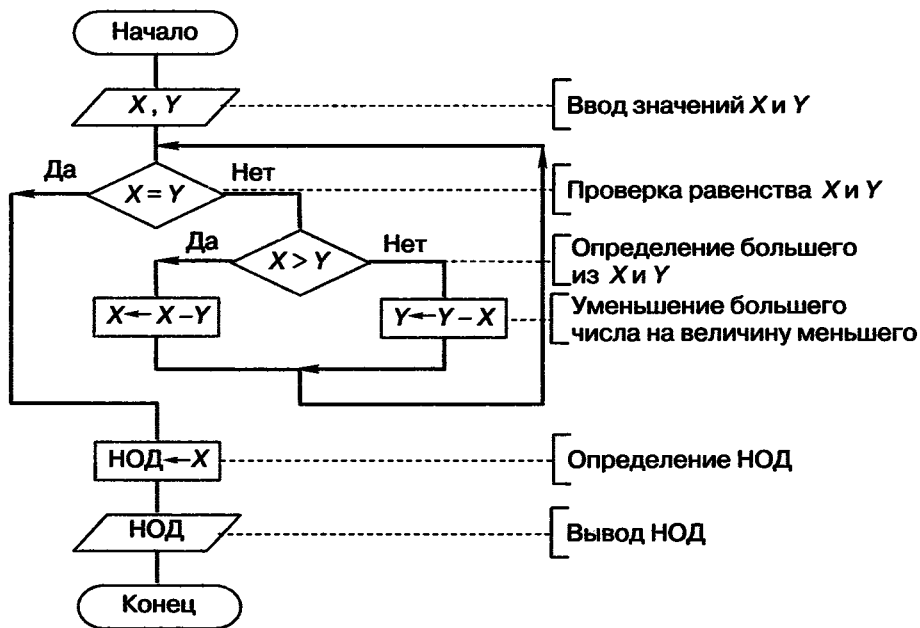


Рис. 2.3. Запись алгоритма Евклида с помощью блок-схемы

Создание детальной блок-схемы сложного алгоритма — трудоёмкая задача. Кроме того, блок-схема, не уместающаяся на одном стандартном листе, теряет своё основное преимущество — наглядность. При разработке сложных алгоритмов блок-схемы удобно использовать в качестве средства для наглядного представления решения задачи в общем виде.

### 2.2.3. Алгоритмические языки

Алгоритмические языки — формальные языки, предназначенные для записи алгоритмов. Каждый из них характеризуется:

- алфавитом — набором используемых символов;
- синтаксисом — системой правил, по которым из символов алфавита образуются правильные конструкции языка;
- семантикой — системой правил, строго определяющей смысл и способ употребления конструкций языка.

Класс алгоритмических языков очень широк. При изучении курса информатики в школах используются различные версии школьного (учебного) алгоритмического языка.

**Школьный алгоритмический язык.** Для записи алгоритмов на школьном алгоритмическом языке используется некоторое ограниченное множество слов, смысл и способ употребления которых заданы раз и навсегда. Это так называемые служебные слова: **алг** (алгоритм), **дано**, **надо**, **нач** (начало), **кон** (конец), **арг** (аргумент), **рез** (результат) и др. При записи алгоритмов в книгах служебные слова выделяются жирным шрифтом, в тетради и на доске — подчёркиванием.

В общем виде программу на школьном алгоритмическом языке можно представить так:

```
алг <название алгоритма>  
нач  
    <последовательность команд>  
кон
```

**Пример 4.** Алгоритм, позволяющий из полного сосуда ёмкостью 12 л отлить половину, пользуясь двумя пустыми сосудами ёмкостью 8 и 5 л.

```
алг переливания  
нач
```

```
    наполнить сосуд ёмкостью 8 л из сосуда ёмкостью 12 л  
    наполнить сосуд ёмкостью 5 л из сосуда ёмкостью 8 л  
    вылить всё из сосуда ёмкостью 5 л в сосуд ёмкостью 12 л  
    вылить всё из сосуда ёмкостью 8 л в сосуд ёмкостью 5 л  
    наполнить сосуд ёмкостью 8 л из сосуда ёмкостью 12 л  
    долить из сосуда ёмкостью 8 л в сосуд ёмкостью 5 л  
    вылить всё из сосуда ёмкостью 5 л в сосуд ёмкостью 12 л
```

```
кон
```

По ссылке <http://www.niisi.ru/kumir/> вы можете скачать систему КуМир (Комплект учебных Миров), в которой используется школьный алгоритмический язык, со встроенными исполнителями Робот, Чертёжник, Водолей и др. Кумир работает в операционных системах Windows и Linux.

Далее, говоря об алгоритмическом языке, мы будем иметь в виду именно школьный алгоритмический язык.



### САМОЕ ГЛАВНОЕ

Существуют различные способы записи алгоритмов: словесное описание, построчная запись, блок-схемы, школьный алгоритмический язык и др. Каждый из этих способов обладает своими достоинствами и недостатками.



### Вопросы и задания



1. Ознакомьтесь с материалами презентации к параграфу, содержащейся в электронном приложении к учебнику. Что вы можете сказать о формах представления информации в презентации и в учебнике? Какими слайдами вы могли бы дополнить презентацию?
2. Каковы основные способы записи алгоритмов?
3. Чем вызвано существование многих способов записи алгоритмов?
4. Дайте словесное описание алгоритма сложения двух обыкновенных дробей  $a/b$  и  $c/d$ .
5. Представьте в виде построчной записи алгоритм решения следующей задачи: «Имеются четыре арбуза различной массы. Как, пользуясь чашечными весами без гирь, путём не более пяти взвешиваний расположить их по возрастанию веса?».
6. Представьте с помощью блок-схемы алгоритм решения следующей задачи: «Из трёх монет одинакового достоинства одна фальшивая (более лёгкая). Как её найти с помощью одного взвешивания на чашечных весах без гирь?».
7. Запишите на алгоритмическом языке алгоритм построения окружности заданного радиуса  $r$ , проходящей через заданные точки  $A$  и  $B$ .
8. В среде КуМир запишите и выполните алгоритм переливаний (пример 4) для исполнителя Водолей.
9. Подготовьте краткую биографическую справку о Маркове А. А. (младшем).



## § 2.3

# Объекты алгоритмов

### *Ключевые слова:*

- величина
- константа
- переменная
- тип
- имя
- присваивание
- выражение
- таблица

### 2.3.1. Величины

Алгоритмы описывают последовательность действий, производимых над некоторыми объектами, определёнными условием задачи. Например, при решении задачи о начислении зарплаты сотрудникам предприятия такими объектами могут быть табельный номер сотрудника, его фамилия, имя, отчество, оклад, отработанное время и т. д.

---

В информатике отдельный информационный объект (число, символ, строка, таблица и др.) называется **величиной**.

---



Величины делятся на постоянные (константы) и переменные. **Постоянной (константой)** называется величина, значение которой указывается в тексте алгоритма и не меняется в процессе его исполнения. **Переменной** называется величина, значение которой меняется в процессе исполнения алгоритма. При исполнении алгоритма в каждый момент времени переменная обычно имеет значение, называемое текущим значением.

**Пример 1.** Величины, выражающие количество дней в неделе, ускорение свободного падения, количество дней в первой декаде месяца, являются константами. Величины, выражающие количество дней в месяце, пульс человека, количество дней в третьей декаде месяца, являются переменными.

В алгоритмах над величинами выполняются некоторые операции. Например:

- арифметические операции  $+$ ,  $-$ ,  $*$  (умножение),  $/$  (деление);
- операции отношения  $<$ ,  $>$ ,  $<=$ ,  $>=$ ,  $=$ ,  $;$ ;
- логические операции И, ИЛИ, НЕ.

Объекты, над которыми выполняются операции, называются операндами. Не всякий объект может быть операндом для выполнения любой операции. Например, текст не может быть объектом для выполнения арифметических операций; отрицательное число не может быть операндом для извлечения квадратного корня и т. д.

Множество величин, объединённых определённой совокупностью допустимых операций, называют величинами определённого типа. При составлении алгоритмов используют величины числового (целого и вещественного), символьного, литерного и логического типов.

В математике и физике оперируют числовыми величинами — натуральными, целыми, действительными числами. При составлении алгоритмов чаще всего используют числовые величины целого и вещественного<sup>1</sup> типов, которые в алгоритмическом языке обозначаются **цел** и **вещ** соответственно.

В задачах, возникающих в повседневной жизни, встречаются и нечисловые величины, значениями которых являются символы, слова, тексты и др. При составлении алгоритмов обработки текстовой информации используют величины символьного (**сим**) и литерного (**лит**) типов. Значением символьной величины является один символ: русская или латинская буква, цифра, знак препинания или другой символ. Значением литерной величины является последовательность символов. Иногда эту последовательность называют строкой или цепочкой. Литерные значения в алгоритме записывают в кавычках, например: 'алгоритм', 'литерная величина', '2011'.

Величины логического (**лог**) типа могут принимать всего два значения:

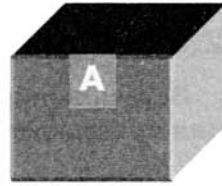
- ДА (ИСТИНА, TRUE, 1);
- НЕТ (ЛОЖЬ, FALSE, 0).

<sup>1</sup> Термин «вещественный» принято использовать наряду с термином «действительный».



Для ссылок на величины используют их имена (идентификаторы). Имя величины может состоять из одной или нескольких латинских букв, из латинских букв и цифр:  $A1$ ,  $M$ ,  $AP$ . Рекомендуется выбирать мнемонические имена, т. е. имена, отражающие суть объектов решаемой задачи, например,  $SUMMA$ ,  $PLAN$ ,  $CENA$  и т. д.

Если величину представить как ящик, содержащий которого является некоторое значение, то имя величины — это ярлык, повешенный на ящик.



### 2.3.2. Выражения

**Выражение** — языковая конструкция для вычисления значения с помощью одного или нескольких операндов.

Выражения состоят из операндов (констант, переменных, функций), объединённых знаками операций. Выражения записываются в виде линейных последовательностей символов (без подстрочных и надстрочных символов, обыкновенных дробей и т. д.); знаки операций пропускать нельзя. Порядок выполнения операций определяется скобками и приоритетом (старшинством) операций; операции одинакового приоритета выполняются слева направо.

Различают арифметические, логические и строковые выражения.

**Арифметические выражения** служат для определения числового значения. Например,  $2 * x + 3$  — арифметическое выражение, значение которого при  $x = 1$  равно пяти, а при  $x = -1$  — единице. Выражение  $\text{sqrt}(x)$  служит для обозначения операции извлечения квадратного корня из  $x$  ( $\sqrt{x}$ ).

**Логические выражения** описывают некоторые условия, которые могут удовлетворяться или не удовлетворяться. Логическое выражение может принимать одно из двух значений — ИСТИНА или ЛОЖЬ. Например, логическое выражение  $(x > 5)$  и  $(x < 10)$  определяет принадлежность точки  $x$  интервалу  $(5; 10)$ :



При  $x = 6$  значение этого выражения — ИСТИНА, а при  $x = 12$  — ЛОЖЬ.

**Строковые выражения** состоят из величин (констант, переменных) символьного и литерного типов, соответствующих функций и операций сцепления (присоединения). Операция сцепления обозначается знаком «+» и позволяет соединить в одну последовательность несколько последовательностей символов. Значениями стро-

ковых выражений являются последовательности символов. Например, если  $A = \text{'том'}$ , то значение строкового выражения  $\text{'a'+A}$  есть  $\text{'атом'}$ .

### 2.3.3. Команда присваивания

Задать конкретное значение величины можно с помощью операции присваивания, которая записывается так:

$\langle \text{имя переменной} \rangle := \langle \text{выражение} \rangle$

Знак  $\langle := \rangle$  читается: «присвоить». Например, запись  $A := B + 5$  читается так: «переменной  $A$  присвоить значение выражения  $B$  плюс 5».

Знаки присваивания  $\langle := \rangle$  и равенства  $\langle = \rangle$  — разные знаки:

- знак  $\langle = \rangle$  означает равенство двух величин, записанных по обе стороны от этого знака;
- знак  $\langle := \rangle$  предписывает выполнение операции присваивания.

Например, запись  $A := A + 1$  выражает не равенство значений  $A$  и  $A + 1$ , а указание увеличить значение переменной  $A$  на единицу.

При выполнении команды присваивания сначала вычисляется значение выражения, стоящего справа от знака  $\langle := \rangle$ , затем результат присваивается переменной, стоящей слева от знака  $\langle := \rangle$ . При этом тип выражения должен быть совместим с типом соответствующей переменной.

Свойства присваивания:

- 1) пока переменной не присвоено значение, она остаётся неопределённой;
- 2) значение, присвоенное переменной, сохраняется в ней вплоть до выполнения следующего присваивания этой переменной нового значения;
- 3) если мы присваиваем некоторой переменной очередное значение, то предыдущее её значение теряется безвозвратно.

**Пример 2.** Составим алгоритм, в результате которого переменные  $A$  и  $B$  литерного типа обменяются своими значениями.

Решение вида

$A := B$

$B := A$

неверно, так как после выполнения первой команды присваивания первоначальное значение переменной  $A$  будет безвозвратно утеряно. Вторая команда присвоит переменной  $B$  текущее значение переменной  $A$ . В результате обе переменные получат одно и то же значение.

Для поиска правильного решения воспользуемся аналогией. Если требуется перелить жидкость из сосуда 1 в сосуд 2, а из сосуда 2 — в сосуд 1, то без дополнительного сосуда 3 здесь не обойтись. Алгоритм переливаний представлен на рис. 2.4.

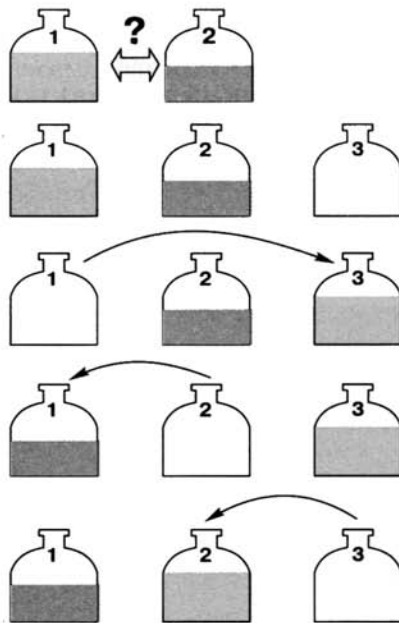


Рис. 2.4. Алгоритм переливаний жидкостей

Для решения исходной задачи введём промежуточную переменную  $M$ . Алгоритм обмена значениями переменных  $A$  и  $B$  запишем так:

**алг** обмен значениями (лит  $A$ ,  $B$ )

**арг**  $A$ ,  $B$

**рез**  $A$ ,  $B$

**нач** лит  $M$

$M := A$

$A := B$

$B := M$

**кон**

Если  $A$  и  $B$  — числовые величины, то обмен их значениями можно организовать и без промежуточной переменной, например так:

$A := A + B$

$B := A - B$

$A := A - B$

### 2.3.4. Табличные величины

В практической деятельности человек часто использует всевозможные таблицы. Это, например, список учащихся в классном журнале, табель успеваемости, таблица результатов спортивных соревнований и т. д. Чаще всего встречаются линейные и прямоугольные таблицы.

**Линейная таблица (одномерный массив)** представляет собой набор однотипных данных, записанных в одну строку или один столбец. Элементы строки (столбца) всегда нумеруются. Например, с помощью линейной таблицы могут быть представлены дни недели (рис. 2.5, а) или количество уроков, пропущенных учеником в течение 5-дневной учебной недели (рис. 2.5, б).

<i>a</i>		<i>b</i>										
1	Понедельник											
2	Вторник											
3	Среда											
4	Четверг											
5	Пятница											
6	Суббота											
7	Воскресенье											
	Васечкин	<table style="border-collapse: collapse; text-align: center;"> <tr> <td style="border: 1px solid black; padding: 2px;">1</td> <td style="border: 1px solid black; padding: 2px;">2</td> <td style="border: 1px solid black; padding: 2px;">3</td> <td style="border: 1px solid black; padding: 2px;">4</td> <td style="border: 1px solid black; padding: 2px;">5</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">6</td> <td style="border: 1px solid black; padding: 2px;">6</td> <td style="border: 1px solid black; padding: 2px;">1</td> <td style="border: 1px solid black; padding: 2px;">0</td> <td style="border: 1px solid black; padding: 2px;">0</td> </tr> </table>	1	2	3	4	5	6	6	1	0	0
1	2	3	4	5								
6	6	1	0	0								

Рис. 2.5. Примеры линейных таблиц

**Прямоугольная таблица (двумерный массив)** — это упорядоченный некоторым образом набор строк (столбцов), содержащих одинаковое количество элементов. Строки прямоугольных таблиц имеют свою нумерацию, столбцы — свою. Например, с помощью прямоугольной таблицы можно представить количество уроков, пропущенных всеми учениками 8 класса в течение 5-дневной учебной недели (рис. 2.6).

	1	2	3	4	5
1. Васечкин	6	6	1	0	0
2. Ионов	0	0	0	0	6
3. Радугина	0	0	1	0	0
...	...	...	...	...	...
19. Чабанюк	0	0	0	0	0

Рис. 2.6. Пример прямоугольной таблицы

Всей совокупности элементов табличной величины даётся одно имя. Элементы различают по их номерам, называемым индексами. Индекс записывается в квадратных скобках сразу за именем таблицы.

Если первую из рассмотренных нами таблиц (см. рис. 2.5, *a*) назвать *WEEK*, то  $WEEK[1] = \text{'понедельник'}$ ,  $WEEK[6] = \text{'суббота'}$ . Назовём третью из рассмотренных таблиц *LES*. Тогда  $LES[1,1] = 6$ ,  $LES[2,5] = 6$ ,  $LES[3,4] = 0$ .

Образно линейная и прямоугольная таблицы показаны на рис. 2.7.

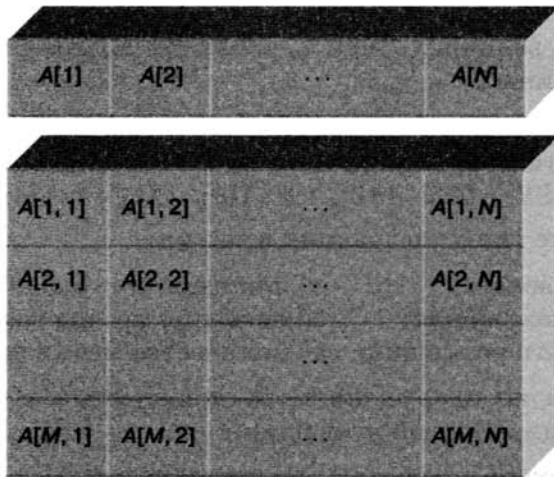


Рис. 2.7. Образное представление линейной и прямоугольной таблиц

## САМОЕ ГЛАВНОЕ

В информатике отдельный информационный объект (число, символ, строка, таблица и др.) называется величиной.

Величины делятся на постоянные (их значения указываются в тексте алгоритма и не меняются в процессе его исполнения) и переменные (их значения меняются в процессе исполнения алгоритма). При составлении алгоритмов используют величины целого, вещественного, логического, символического и литерного типов.

Для ссылок на величины используют их имена (идентификаторы). Имя величины может состоять из одной или нескольких латинских букв, из латинских букв и цифр.

Таблица (массив) — набор некоторого числа однотипных элементов, которым присвоено одно имя. Положение элемента в таблице однозначно определяется его индексами.



### Вопросы и задания



1. Ознакомьтесь с материалами презентации к параграфу, содержащейся в электронном приложении к учебнику. Используйте эти материалы при подготовке ответов на вопросы и выполнении заданий.
2. Что такое величина? Чем отличаются постоянные и переменные величины?
3. Величины каких типов используются при записи алгоритмов?
4. Укажите тип величины, если её значение равно: 2010; 14.48; 'ДА'; FALSE, -125; '142';  $1,4 \cdot 10^5$ ; .123E-2; 'пять'.
5. Определите типы следующих величин:
  - а) вес человека;
  - б) марка автомобиля;
  - в) год вашего рождения;
  - г) площадь фигуры;
  - д) название месяца года;
  - е) количество мест в самолёте.
6. Приведите по одному примеру допустимых и недопустимых значений для каждой из величин:
  - а) температура человека;
  - б) скорость автомашины;
  - в) площадь страны;
  - г) название дня недели.
7. Для чего предназначена команда присваивания? Каковы её основные свойства?
8. Какие команды присваивания составлены правильно?
  - а)  $A := B$
  - б)  $A = B$
  - в)  $A = B + 1$
  - г)  $A + 1 := A$
9. Придумайте свой алгоритм обмена значениями числовых переменных  $A$  и  $B$ .
10. Сколько промежуточных переменных потребуется для того, чтобы переменной  $A$  было присвоено значение переменной  $B$ , переменной  $B$  — значение переменной  $C$ , а переменной  $C$  — значение переменной  $A$ ? Запишите соответствующий алгоритм на алгоритмическом языке.



11. После выполнения команды присваивания  $x:=x+y$  значение переменной  $x$  равно 3, а значение переменной  $y$  равно 5. Чему были равны значения переменных  $x$  и  $y$  до выполнения указанной команды присваивания?



12. Что называют выражением? Каковы основные правила записи выражений?

13. Переведите из линейной записи в общепринятую:

а)  $a * b / c$ ;

г)  $(a + b) / c$ ;

б)  $a / b * c$ ;

д)  $a + b / c + d$ ;

в)  $a + b / c$ ;

е)  $(a + b) / (c + d)$ .

14. Запишите на алгоритмическом языке:

а)  $ax^2 + bx + c$ ;

б)  $v + \frac{at^2}{2}$ ;

в)  $\frac{1}{2}(a + b)h$ ;

г)  $\frac{1 + x_1x_2}{b^2c}$ ;

д)  $\sqrt{a^2 + b^2}$ .

15. Запишите логическое выражение, истинное при выполнении указанного условия и ложное в противном случае:



а)  $x$  принадлежит отрезку  $[0, 1]$ ;

б)  $x$  лежит вне отрезка  $[0, 1]$ ;

в) каждое из чисел  $x, y$  положительно;

г) хотя бы одно из чисел  $x, y$  положительно;

д) ни одно из чисел  $x, y$  не является положительным;

е) только одно из чисел  $x, y$  положительно.

16. Изобразите в декартовой прямоугольной системе координат область, в которой и только в которой истинно следующее логическое выражение:

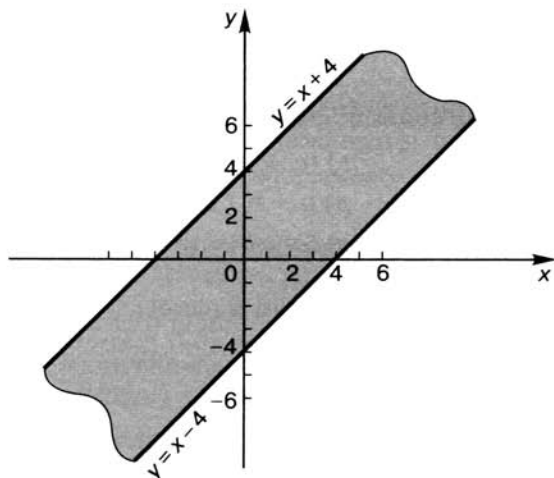


а)  $(x > -1) \text{ и } (x <= 1) \text{ и } (y > -1) \text{ и } (y <= 1)$ ;

б)  $(y >= x) \text{ и } (y >= -x) \text{ и } (y <= 1)$ .



17. Запишите логическое выражение, принимающее значение TRUE, когда точка с координатами  $(x, y)$  принадлежит закрашенной области.



18. Запишите команду присваивания, в результате выполнения которой логическая переменная  $t$  получает значение TRUE, если выполняется указанное условие, и значение FALSE в противном случае:
- $x$  — положительное число;
  - хотя бы одно из чисел  $x, y, z$  равно нулю;
  - числа  $x, y, z$  равны между собой.
19. Какие из приведённых ниже величин целесообразно представлять с помощью таблиц?

*Величины:* список учеников класса, рост учеников класса, средний рост учеников класса, оценка ученика по физике, средний балл ученика по физике, оценки учеников за контрольную работу по информатике, длины сторон треугольника, длины сторон нескольких треугольников, названия дней недели, имя человека, площадь фигуры, периметры нескольких прямоугольников, самая холодная температура воздуха в январе, количество девочек в классе, самая дождливая декада июня.



## Основные алгоритмические конструкции

### Ключевые слова:

- следование
- ветвление
- повторение
- линейные алгоритмы
- разветвляющиеся алгоритмы
- циклические алгоритмы

Человеку в жизни приходится решать множество различных задач. Решение каждой из них описывается своим алгоритмом, и разнообразие этих алгоритмов очень велико. Вместе с тем для записи любого алгоритма достаточно трёх основных алгоритмических конструкций (структур): следования, ветвления, повторения. Это положение выдвинул и доказал Э. Дейкстра в 70-х гг. прошлого века.



Эдсгер Вибе Дейкстра (1930–2002) — выдающийся нидерландский учёный, идеи которого оказали огромное влияние на развитие компьютерной индустрии.

### 2.4.1. Следование

**Следование** — алгоритмическая конструкция, отображающая естественный, последовательный порядок действий. Алгоритмы, в которых используется только структура «следование», называются **линейными алгоритмами**.

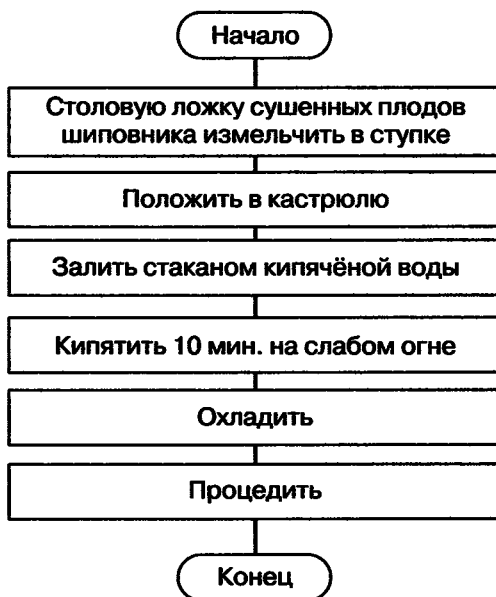


Графическое представление алгоритмической конструкции «следование» приведено на рис. 2.8.



Рис. 2.8. Алгоритмическая конструкция «следование»

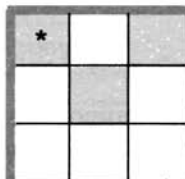
**Пример 1.** Линейный алгоритм приготовления отвара шиповника.



Обратите внимание, что многие из предписаний этого алгоритма могут потребовать детализации — представления в виде некоторой совокупности более мелких предписаний.

**Пример 2.** У исполнителя Робот есть четыре команды перемещения (вверх, вниз, влево и вправо), при выполнении каждой из них Робот перемещается на одну клетку в соответствующем направлении. По команде закрасить Робот закрашивает клетку, в которой он находится. Запишем линейный алгоритм, исполняя который

Робот нарисует на клетчатом поле следующий узор и вернётся в исходное положение, обозначенное звёздочкой:



```

алг узор
нач
    закрасить
    вправо
    вправо
    закрасить
    вниз
    влево
    закрасить
    вверх
    влево
кон
    
```

**Пример 3.** Дан фрагмент линейного алгоритма:

```

x:=2
y:=x*x
y:=y*y
x:=y*x
s:=x+y
    
```

Выясним, какое значение получит переменная  $s$  после выполнения этого фрагмента алгоритма. Для этого составим таблицу значений переменных, задействованных в алгоритме:

Шаг алгоритма	Переменные		
	$x$	$y$	$s$
1	2	–	–
2		4	–
3		16	–
4	32		–
5			48

Составленная нами таблица значений переменных моделирует работу исполнителя этого алгоритма.





**Пример 4.** Некоторый исполнитель может выполнять над целыми числами кроме операций сложения, вычитания, умножения и деления ещё две операции: с помощью операции `div` вычисляется целое частное, с помощью операции `mod` — остаток.

Например:  $5 \text{ div } 2 = 2$ ;  $5 \text{ mod } 2 = 1$ ;  $2 \text{ div } 5 = 0$ ;  $2 \text{ mod } 5 = 2$ .

Покажем, как с помощью этих операций можно реализовать алгоритм работы кассира, выдающего покупателю сдачу ( $s$ ) наименьшим количеством банкнот по 500 ( $k500$ ), 100 ( $k100$ ), 50 ( $k50$ ) и 10 ( $k10$ ) рублей.

```
k500:=s div 500
s:=s mod 500
k100:=s div 100
s:=s mod 100
k50:=s div 50
s:=s mod 50
k10:=s div 10
```



Исполните алгоритм для  $s = 745$  и  $s = 1864$ . Составьте соответствующие таблицы значений переменных.



Ознакомьтесь с имеющимся в Единой коллекции цифровых образовательных ресурсов модулем для коллективной работы «Линейные алгоритмы» (217039). Совместно с друзьями постарайтесь составить алгоритмы для имеющихся в модуле задач. Пройдите тестирование.

### 2.4.2. Ветвление



**Ветвление** — алгоритмическая конструкция, в которой в зависимости от результата проверки условия («да» или «нет») предусмотрен выбор одной из двух последовательностей действий (ветвей). Алгоритмы, в основе которых лежит структура «ветвление», называют **разветвляющимися**.

Блок-схема ветвления представлена на рис. 2.9. Каждая ветвь может быть любой степени сложности (рис. 2.9, а), а может вообще не содержать предписаний (рис. 2.9, б).

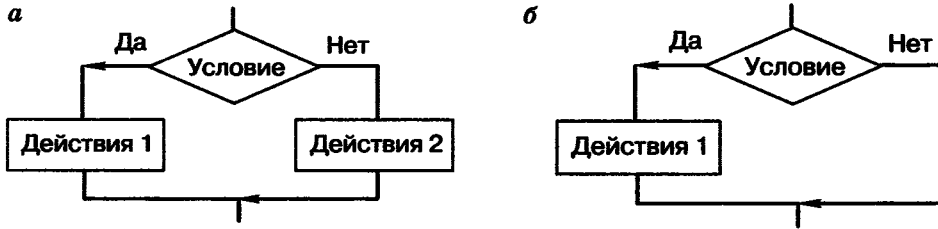


Рис. 2.9. Структура «ветвление»: а — полная форма ветвления; б — неполная форма ветвления

На алгоритмическом языке команда ветвления записывается так:

Полная форма ветвления:

```
если <условие>
    то <действия 1>
    иначе <действия 2>
все
```

Неполная форма ветвления:

```
если <условие>
    то <действия 1>
все
```

### Пример 5

```
алг правописание приставок НЕ, НИ
нач
    если приставка под ударением
        то писать НЕ
        иначе писать НИ
    все
кон
```

### Пример 6

```
алг сборы на прогулку
нач
    если на улице дождь
        то взять зонтик
    все
кон
```



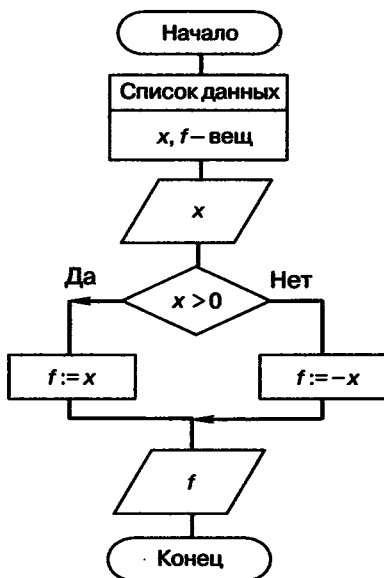
Для записи условий, в зависимости от результатов проверки которых выбирается та или иная последовательность действий, используются операции сравнения:

- $A < B$  —  $A$  меньше  $B$ ;
- $A \leq B$  —  $A$  меньше или равно  $B$ ;
- $A = B$  —  $A$  равно  $B$ ;
- $A > B$  —  $A$  больше  $B$ ;
- $A \geq B$  —  $A$  больше или равно  $B$ ;
- $A \neq B$  —  $A$  не равно  $B$ .

Здесь буквы  $A$  и  $B$  можно заменять на любые переменные, числа и арифметические выражения. Приведённые операции сравнения допускаются и для символьных переменных.



**Пример 7.** Алгоритм вычисления функции  $f(x) = |x|$  для произвольного числа  $x$ .

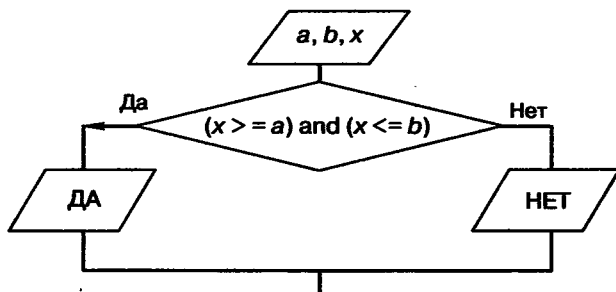


Обратите внимание на второй блок этой блок-схемы. В нём представлены имена и типы величин (данных), обрабатываемых в алгоритме.

Условия, состоящие из одной операции сравнения, называются простыми. В качестве условий при организации ветвлений можно использовать и составные условия. Составные условия получаются из простых с помощью логических связок **and** (**и**), **or** (**или**), **not** (**не**): **and** означает одновременное выполнение всех условий, **or** — выполнение хотя бы одного условия, а **not** означает отрицание условия, записанного за словом **not**.

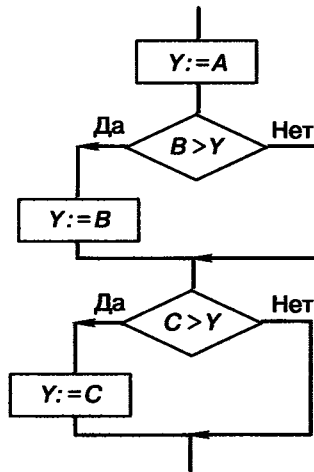


**Пример 8.** Алгоритм определения принадлежности точки  $x$  отрезку  $[a, b]$ . Если точка  $x$  принадлежит данному отрезку, то выводится ответ ДА, в противном случае — НЕТ.



Существует достаточно много ситуаций, в которых приходится выбирать не из двух, а из трёх и более вариантов. Есть разные способы построения соответствующих алгоритмов. Один из них — составить комбинацию из нескольких ветвлений.

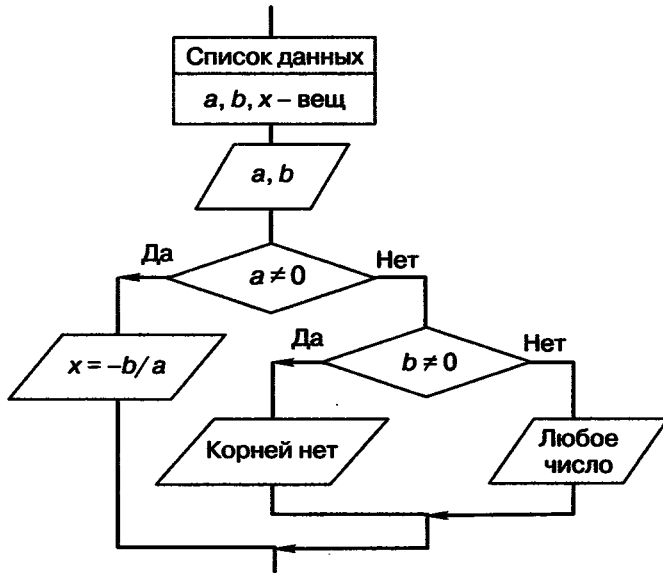
**Пример 9.** Алгоритм, в котором переменной  $Y$  присваивается значение большей из трёх величин  $A$ ,  $B$  и  $C$ .



Пусть  $A = 10$ ,  $B = 30$  и  $C = 20$ . Тогда процесс выполнения алгоритма можно представить в следующей таблице:

Шаг алгоритма	Константы			Переменная $Y$	Условие
	$A$	$B$	$C$		
1	10	30	20	10	
2					$30 > 10$ (Да)
3				30	
4					$20 > 30$ (Нет)

**Пример 10.** Алгоритм решения линейного уравнения  $ax + b = 0$ .



**Пример 11.** Исполнитель Робот может выполнять ту или иную последовательность действий в зависимости от выполнения следующих простых условий:

справа свободно  
слева свободно  
сверху свободно  
снизу свободно  
клетка чистая

справа стена  
слева стена  
сверху стена  
снизу стена  
клетка закрашена

Также Робот может действовать в зависимости от выполнения составных условий.

**?** Подумайте, в какую клетку переместится Робот из клетки, обозначенной звёздочкой, при выполнении следующего фрагмента алгоритма.

**если** справа свободно **или** снизу свободно  
**то** закрасить  
**все**  
**если** справа стена  
**то** влево  
**все**  
**если** слева стена  
**то** вправо  
**все**