

# ФЕДЕРАЛЬНЫЙ

ИНСТИТУТ ПЕДАГОГИЧЕСКИХ ИЗМЕРЕНИЙ

С.С. Крылов, Д.М. Ушаков

# ИНФОРМАТИКА



## ТЕМАТИЧЕСКАЯ РАБОЧАЯ ТЕТРАДЬ

учени \_\_\_\_\_ класса \_\_\_\_\_

\_\_\_\_\_ ШКОЛЫ \_\_\_\_\_

- Основные понятия
- Тематические задания с разбором решений
- Задания в форме ЕГЭ
- Ответы



ЭКЗАМЕН

УНИКАЛЬНАЯ МЕТОДИКА ПОДГОТОВКИ,  
РАЗРАБОТАННАЯ СПЕЦИАЛИСТАМИ ФИПИ

# ФЕДЕРАЛЬНЫЙ

ИНСТИТУТ ПЕДАГОГИЧЕСКИХ ИЗМЕРЕНИЙ



# ЕГЭ

## ТЕМАТИЧЕСКАЯ РАБОЧАЯ ТЕТРАДЬ

С.С. Крылов, Д.М. Ушаков

# ИНФОРМАТИКА

*Основные понятия*

*Тематические задания  
с разбором решений*

*Задания в форме ЕГЭ*

*Ответы*

*Издательство  
«ЭКЗАМЕН»  
МОСКВА, 2010*

УДК 373.8:002

ББК 74.263.2

К85

**Крылов, С.С.**

К85 ЕГЭ. Информатика. Тематическая рабочая тетрадь ФИПИ / С.С. Крылов, Д.М. Ушаков. — М.: Издательство «Экзамен», 2010. — 151, [1] с. (Серия «ЕГЭ. Тематическая рабочая тетрадь»)

ISBN 978-5-377-02980-9

Тематическая рабочая тетрадь по информатике, созданная специалистами Федерального института педагогических измерений, ориентирована на подготовку учащихся средней школы к успешной сдаче ЕГЭ.

Книга содержит множество тематических заданий для отработки каждого элемента содержания ЕГЭ по информатике.

Уникальная методика подготовки, разработанная специалистами ФИПИ, поможет учащимся научиться правильно оформлять работу, выявлять критерии оценивания, акцентировать внимание на формулировках ряда заданий и избегать ошибок, связанных с невнимательностью и рассеянностью на экзамене.

Использовать представленную рабочую тетрадь можно как в классе, так и дома.

Рабочая тетрадь ориентирована на один учебный год, однако при необходимости позволит в кратчайшие сроки выявить пробелы в знаниях ученика и отработать те задания, в которых допускается больше всего ошибок, непосредственно за несколько дней до экзамена.

Книга предназначена для учащихся средней школы, учителей информатики, родителей и репетиторов.

**УДК 373.8:002**

**ББК 74.263.2**

---

Формат 60x90/8. Гарнитура «Школьная».

Бумага газетная. Уч.-изд. л. 7,30. Усл. печ. л. 19.

Тираж 20 000 экз. Заказ № 17874.

---

ISBN 978-5-377-02980-9

© ФИПИ, 2009

© Крылов С.С., Ушаков Д.М., 2009

© Издательство «**ЭКЗАМЕН**», 2009

© Художественное оформление «Издательство «**ЭКЗАМЕН**», 2009

# Содержание

Введение.....	4
1. Количество информации. Скорость передачи информации .....	5
2. Системы счисления .....	13
3. Кодирование информации.....	31
4. Основы логики.....	41
5. Алгоритмизация и программирование .....	57
6. Информационные модели .....	99
7. Определение выигрышной стратегии игры. (Анализ и построение дерева игры) .....	106
8. Файловая система компьютера.....	119
9. Электронные таблицы .....	125
10. Базы данных .....	134
11. Сетевые технологии.....	140
Ответы .....	148

# **Введение**

При создании этого учебного пособия использован более чем пятилетний опыт проведения Единого государственного экзамена по информатике. В «Рабочей тетради» приведены задания по всем темам экзамена, каждая глава соответствует определенной теме. Порядок расположения заданий внутри каждой главы соответствует логике изложения материала в учебниках, поэтому «Рабочую тетрадь» можно использовать не только для обобщающего повторения перед экзаменом, но и в качестве учебного пособия в течение всего учебного года.

Поскольку данное издание представляет собой не только сборник типовых заданий в формате ЕГЭ для контроля и самоконтроля, но и учебное пособие, многие задания сформулированы как обычные задачи, в том виде, который наиболее способствует пониманию учебного материала.

В рабочей тетради не рассматриваются вопросы обучения программированию на каком-либо языке, поскольку в разных школах изучаются различные языки программирования, а рассмотрение нескольких языков не предусматривается форматом данной книги. В теме «Программирование» в качестве теоретического материала приведены общие принципы технологии программирования, соблюдение которых необходимо при выполнении заданий ЕГЭ по информатике.

Большинство заданий рабочей тетради снабжено ответами. Часть заданий намеренно оставлена авторами без ответов, для того чтобы их можно было использовать при проведении проверочных работ.

Авторы уверены, что использование данной «Рабочей тетради» и добросовестное выполнение тренировочных заданий обязательно улучшат ваши результаты на экзамене.

**Желаем успеха!**

# 1. Количество информации. Скорость передачи информации

## Основные понятия

**Информация** — первичное понятие, не имеющее общепринятого, классического определения. Одно из возможных определений: сведения об окружающем нас мире.

Для измерения информации используются специальные единицы: бит, байт, килобайт, мегабайт, гигабайт и т.д.

Минимальной (и основной) единицей измерения информации является один бит.

**Бит** — количество информации, которое уменьшает неопределенность в два раза.

Другое определение:

**Бит** — количество информации, необходимое для передачи сообщения «Да»/«Нет».

Бит может принимать только два возможных значения. Обычно их обозначают «1» и «0».

Восемь бит составляют один байт.

Для обозначения больших единиц измерения информации используются префиксы кило-, мега-, гига- и другие, например, килобайт, мегабайт, гигабайт. Однако, в отличие от общепризнанных значений префиксов («кило-» =  $10^3$ , «мега-» =  $10^6$ , «гига-» =  $10^9$ ) при использовании их применительно к битам и байтам они обозначают близкие степени «двойки»: «кило-» =  $2^{10}$ , «мега-» =  $2^{20}$ , «гига-» =  $2^{30}$ . То есть

$$1024 \text{ байт} = 1 \text{ килобайт (кбайт, Кб)}$$

$$1024 \text{ килобайт} = 1 \text{ мегабайт (Мбайт, Мб)}$$

$$1024 \text{ мегабайт} = 1 \text{ гигабайт (Гбайт, Гб)}$$

Для вычисления количества информации применяют несколько различных способов, в зависимости от ситуации.

- Для вычисления количества информации в сообщении об одном из равновероятных событий, общее количество которых равно  $N$ , используйте формулу:

$$2^i = N,$$

где  $i$  — количество информации в сообщении.

- Для вычисления количества информации в сообщении об одном из неравновероятных событий, вероятность которого равна  $p$ , используйте формулу:

$$i = -[\log_2 p],$$

где  $i$  — количество информации в сообщении, квадратные скобки обозначают ближайшее целое, меньшее или равное значению выражения в скобках.

- Для вычисления количества информации в сообщении из  $k$  символов некоторого алфавита, в котором  $N$  различных знаков, используйте формулу:

$$I = ki,$$

где  $I$  — количество информации в сообщении,  $i$  — найти из формулы  $2^i = N$ .

Скорость передачи информации измеряется в битах в секунду и вычисляется по формуле:

$$V = I/t,$$

где  $V$  — скорость передачи информации,  $I$  — количество информации в сообщении,  $t$  — время передачи сообщения.

















## 2. Системы счисления

### Основные понятия

**Система счисления** (ССч) — набор знаков, используемых для записи чисел и правила записи чисел. Эти знаки называются *цифрами*.

Набор этих цифр называется *алфавитом системы счисления*.

Количество цифр в алфавите называется *мощностью алфавита*.

Различают позиционные и непозиционные системы счисления.

Если для каждого числа системы счисления выполняется правило: вес цифры (ее значение) зависит от положения цифры в числе, такая система счисления называется *позиционной*. Если хотя бы для одного числа это правило не выполняется, система счисления называется *непозиционной*.

Пример непозиционной системы счисления — римская. В ней для числа II вес каждой цифры одинаков (равен единице).

Количество цифр в позиционной системе счисления называется *основанием* системы счисления. Именно во столько раз вес каждого разряда больше веса соседнего.

Основание — основная характеристика позиционной системы счисления.

Система счисления, которой мы пользуемся в повседневной жизни и которую изучаем в школе, — десятичная позиционная. Десятичная — потому что в ней используется десять цифр для записи чисел (от «0» до «9») и именно в десять раз вес каждого разряда отличается от соседнего (вес сотен в десять раз больше веса десятков).

Самое маленькое основание позиционной системы счисления — 2.

Это самая простая система счисления для записи чисел, в ней всего два знака — «0» и «1». Поэтому именно двоичная система счисления используется для хранения чисел в компьютере.

Если нужно записывать числа в системе счисления, основание которой больше 10, привычных арабских цифр (от 0 до 9) не хватает и принято использовать буквы латинского алфавита: десять — A, одиннадцать — B, и т.д.

Обычно используется 16-теричная система счисления.

При записи чисел в различных системах счисления принято записывать основание системы счисления справа внизу возле числа. Например, число 6 в восьмеричной системе счисления записывают:  $6_8$ . Если основание системы счисления справа внизу возле числа не указано, считается, что это десятичная система счисления.

Для перевода числа из какой-либо системы счисления в десятичную необходимо:

- 1) пронумеровать разряды числа справа налево, начиная с нуля;
- 2) умножить каждую цифру числа на основание его системы счисления, возведенное в степень номера этого разряда;
- 3) сложить полученные числа.

Для перевода десятичного числа в другую систему счисления необходимо:

- 1) делить нацело с остатком число на нужное основание системы счисления;
- 2) получившееся частное (целое) тоже делить нацело с остатком на это основание;
- 3) продолжать деления до тех пор, пока частное не получится равно нулю;
- 4) выписать остатки в порядке, обратном их получению.

## Практическая часть

**2.1.** Расставьте необходимые термины напротив их определений.

Термины: система счисления, алфавит системы счисления, мощность алфавита, основание системы счисления.

- \_\_\_\_\_ а) количество цифр, используемых при записи чисел
- \_\_\_\_\_ б) набор цифр, используемых при записи чисел и правила записи чисел
- \_\_\_\_\_ в) правила записи цифр
- \_\_\_\_\_ г) набор цифр, используемых при записи чисел
- \_\_\_\_\_ д) количество цифр в алфавите позиционной системы счисления

**2.2.** Чем отличается позиционная система счисления от непозиционной?

Ответ: \_\_\_\_\_

**2.3.** В позиционной системе счисления во сколько раз вес (значение) каждого разряда больше предыдущего?

Ответ: \_\_\_\_\_

**2.4.** В  $N$ -ичной системе счисления для записи чисел используется \_\_\_\_\_ различных цифр. Самая маленькая цифра равна \_\_\_\_\_. Самая большая цифра равна \_\_\_\_\_.

**2.5.** В  $N$ -ичной системе счисления число, которое на 1 больше, чем самая старшая цифра, записывается как \_\_\_\_\_.

**2.6.** В двоичной системе счисления для записи чисел используются только \_\_\_\_\_ и \_\_\_\_\_.

**2.7.** В шестнадцатеричной системе счисления кроме обычных десяти арабских цифр (от 0 до 9) используются также \_\_\_\_\_ букв латинского алфавита: от \_\_\_\_\_ до \_\_\_\_\_. Цифра «десять» записывается как \_\_\_\_\_, цифра «\_\_\_\_\_» записывается как  $F$ .

**2.8.** Обозначьте на рисунке следующие термины (обведите и подпишите или напишите термин и стрелками укажите их): разряд, номер разряда, основание системы счисления.

4 3 2 1 0  
5 2 4 3 6 8

**2.9.** В шестнадцатеричной системе счисления между числами  $2B_{16}$  и  $2E_{16}$  находятся числа \_\_\_\_\_<sub>16</sub> и \_\_\_\_\_<sub>16</sub>.

**2.10.** Заполните пустые клетки таблицы последовательными числами в системах счисления с основанием 3, 4, 5 (таблицу соответствия между десятичной, двоичной и шестнадцатеричной системами счисления мы рекомендуем вам выучить наизусть).

**Система счисления**

10	2	8	16	3	4	5
0	0	0	0	0	0	0
1	1	1	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			

**2.11.** После числа:  $100111_2$      $212_3$      $37_8$      $BF_{16}$      $21333_4$      $66_7$   
 следует число: \_\_\_\_\_<sub>2</sub>    \_\_\_\_\_<sub>3</sub>    \_\_\_\_\_<sub>8</sub>    \_\_\_\_\_<sub>16</sub>    \_\_\_\_\_<sub>4</sub>    \_\_\_\_\_<sub>7</sub>

**2.12.** Числу:  $10100_2$      $2100_3$      $520_8$      $A00_{16}$      $3120_4$      $50_7$   
 предшествует  
 число: \_\_\_\_\_<sub>2</sub>    \_\_\_\_\_<sub>3</sub>    \_\_\_\_\_<sub>8</sub>    \_\_\_\_\_<sub>16</sub>    \_\_\_\_\_<sub>4</sub>    \_\_\_\_\_<sub>7</sub>

**2.13.** Между числами:  $1111_2$  и  $10001_2$      $BF_{16}$  и  $C1_{16}$      $2221_3$  и  $10000_3$      $109_{16}$  и  $10B_{16}$   
 стоит число: \_\_\_\_\_<sub>2</sub>    \_\_\_\_\_<sub>16</sub>    \_\_\_\_\_<sub>3</sub>    \_\_\_\_\_<sub>16</sub>

**2.14.** В каждом столбце обведите большее число и подчеркните меньшее:

$10010_2$	$2010_3$	$507_8$	$A00_{16}$	$10000_4$	$1FF_{16}$
$1111_2$	$1112_3$	$277_8$	$E0F_{16}$	$30000_4$	$F00_{16}$
$10100_2$	$2212_3$	$374_8$	$10D_{16}$	$23012_4$	$333_{16}$
$11000_2$	$1222_3$	$630_8$	$CFF_{16}$	$32100_4$	$ABC_{16}$



**2.15.** Расставьте цифры — порядок выполнения действий при переводе из любой системы счисления в десятичную:

- сложить все числа
- пронумеровать разряды числа справа налево, начиная с нуля
- цифру каждого разряда умножить на основание системы счисления, возведенное в степень номера разряда

**2.16.** Расставьте цифры — порядок выполнения действий при переводе из десятичной системы счисления в любую другую:

- выписать остатки от деления слева направо в порядке, обратном их получению
- повторять действие \_\_\_ до тех пор, пока частное от деления не будет равно нулю
- поделить число нацело с остатком на основание системы счисления, в которую переводим

**Напоминание.** При переводе из двоичной системы счисления в десятичную цифры, которые нужно умножать на число 2 в какой-то степени равны 0 или 1. Все, что умножено на 0, все равно дает ноль. Поэтому эти слагаемые лучше просто опустить. Все, что умножено на 1, таким же и остается. Поэтому эти умножения тоже лучше опустить. Получается, достаточно просто сложить степени «двойки», в разрядах которых стоят «единицы».

**2.17.** Переведите числа в десятичную систему счисления (заполните пропущенные в клетках цифры):

$$\begin{array}{c} \square_2 \square_0 \\ 2314_5 = \square \cdot 5^3 + 3 \cdot 5^{\square} + 1 \cdot \square^1 + \square \cdot 5^0 = 250 + \square\square + \square + 4 = \square\square\square_{10} \end{array}$$

$$\begin{array}{c} \square\square\square\square_2 \square\square \\ 101001_2 = 2^6 + 2^{\square} + 2^{\square} + 1 = \square\square + 16 + \square + 1 = \square\square_{10} \end{array}$$

$$\begin{array}{c} \square\square\square \\ 2AB_{16} = \square \cdot 16^2 + 10 \cdot \square\square^{\square} + \square\square = 512 + \square\square\square + \square\square = \square\square\square_{10} \end{array}$$

**2.18.** Переведите числа в десятичную систему счисления:

$$1010110_2 = \underline{\hspace{10em}}$$

$$21020_3 = \underline{\hspace{10em}}$$

$$526_8 = \underline{\hspace{10em}}$$

$$3CE_{16} = \underline{\hspace{10em}}$$

$$3021_4 = \underline{\hspace{10em}}$$

$$256_7 = \underline{\hspace{10em}}$$



Получив сумму степеней «двойки» рекомендуем такой прием: начинаем про себя называть степени двойки, с самой старшей (в данном примере, с шести), по убывающей, подряд, до нуля. Для каждой названной степени записываем друг за другом цифры: «1» — если такая степень есть в сумме, или «0» — если такой степени нет.

В данном примере для степеней 6, 4, 1 и 0 записали «1», а для 5, 3 и 2 — «0».

Для эффективного использования этого метода необходимо наизусть знать степени числа «2». Вообще говоря, это знание очень поможет вам решать множество задач ЕГЭ по информатике. Мы рекомендуем заранее выучить степени двойки хотя бы до 10-й, а лучше — до 16-й.

**Напоминание.** Две системы счисления будем называть **родственными**, если основание одной системы счисления равно степени основания другой. Например, 2 и 8, 2 и 16, 3 и 9.

Для произвольной пары систем счисления, чтобы перевести число из одной системы в другую, нужно осуществлять два перевода — сначала из исходной в десятичную, потом из десятичной в нужную. В родственных системах счисления можно осуществлять перевод напрямую. Чтобы не описывать процесс в общем виде (который вам, вероятно, никогда не понадобится) мы остановимся на системах счисления  $2 \leftrightarrow 8$ ,  $2 \leftrightarrow 16$ .

**Замечание.** Вы будете гораздо быстрее осуществлять действия по переводу чисел в родственных системах счисления, если выучите наизусть таблицу соответствия цифр от 0 до 15 в двоичной, восьмеричной и шестнадцатеричной системах счисления (таблицу мы привели несколькими страницами ранее). Это существенно сэкономит вам время на экзамене.

При решении нижеследующих задач настоятельно рекомендуем этой таблицей не пользоваться — либо выучите ее наизусть, либо научитесь переводить числа «на лету». Если вы будете просто заглядывать в таблицу — не научитесь делать это сами и не сможете осуществлять перевод на экзамене.

**2.25.** Расставьте цифры — порядок выполнения действий при переводе из двоичной системы счисления в восьмеричную:

- записать получившиеся цифры в том же порядке, в котором записаны группы разрядов
- сгруппировать разряды группами по 3, справа налево
- двоичное число в каждой группе перевести в десятичную систему счисления

**2.26.** Переведите число  $10001101_2$  в восьмеричную систему счисления:

$$\underbrace{10001101}_2 \quad \overset{2}{1} \overset{10}{01}_2 = 2^2 + 2^0 = 4 + 1 = 5$$

$5_8$

**2.27.** Переведите числа  $11111100_2$  и  $1010110_2$  в восьмеричную систему счисления:

$$\underbrace{11111100}_2 \quad \underbrace{1010110}_2$$

$8$             $8$

**2.28.** Расставьте цифры — порядок выполнения действий при переводе из двоичной системы счисления в шестнадцатеричную:

- записать получившиеся цифры в том же порядке, в котором записаны группы разрядов
- сгруппировать разряды группами по \_\_\_\_\_, справа налево
- двоичное число в каждой группе перевести в десятичную систему счисления
- получившееся десятичное число перевести в 16-ю цифру

**2.29.** Переведите число  $11010101101_2$  в шестнадцатеричную систему счисления:

$$\begin{array}{ccc} & & \begin{matrix} 3 & 2 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{matrix} \\ \underline{11010101101}_2 & & 1101_2 = 2^3 + 2^2 + 2^0 = 8 + 4 + 1 = 13 \\ \begin{matrix} \square & \square & 13 \\ \parallel & \parallel & \parallel \\ \square & \square & D_{16} \end{matrix} & & \end{array}$$

**2.30.** Переведите числа  $111111100_2$  и  $11101011110_2$  в шестнадцатеричную систему счисления:

$$\begin{array}{ccc} \underline{111111100}_2 & & \underline{11101011110}_2 \\ \square & \square & \square_{16} \quad \square & \square & \square_{16} \end{array}$$

**2.31.** Расставьте цифры — порядок выполнения действий при переводе из восьмеричной системы счисления в двоичную:

- выписать тройки двоичных разрядов друг за другом в том же порядке, в котором стоят цифры восьмеричного числа
- каждую цифру восьмеричного числа перевести в двоичную систему счисления
- если двоичное представление цифры состоит меньше, чем из трех разрядов, дописать слева один или два нуля, чтобы всего получилось ровно три разряда

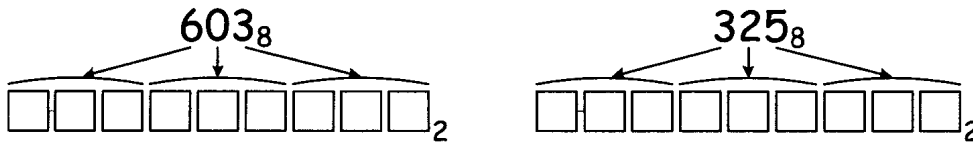
*Пример.*

$$\begin{array}{ccc} & & \begin{matrix} 5 & 3 & 2 \\ 8 & & & \end{matrix} \\ & \swarrow & \downarrow & \searrow \\ \underline{101011010}_2 & & & \end{array} \quad \begin{array}{l} 5 = 4 + 1 = 2^2 + 2^0 = 101_2 \\ 3 = 2 + 1 = 2^1 + 2^0 = 11_2 \\ 11_2 = 011_2 \end{array}$$

**2.32.** Переведите число  $714_8$  в двоичную систему счисления:

$$\begin{array}{ccc} & & \begin{matrix} 7 & 1 & 4 \\ 8 & & \end{matrix} \\ & \swarrow & \downarrow & \searrow \\ \square & \square & 1 & \square & \square & \square & 1 & \square & \square & \square_2 & & \end{array} \quad \begin{array}{l} 7 = 4 + \square + 1 = 2^{\square} + 2^1 + 2^{\square} = \square\square_2 \\ 4 = 2^{\square} = 1\square\square_2 \\ 1 = 1_2 = \square\square\square_2 \end{array}$$

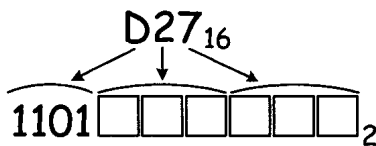
**2.33.** Переведите числа  $603_8$  и  $325_8$  в двоичную систему счисления:



**2.34.** Расставьте цифры — порядок выполнения действий при переводе из шестнадцатеричной системы счисления в двоичную:

- выписать четверки двоичных разрядов друг за другом в том же порядке, в котором стоят цифры шестнадцатеричного числа
- если двоичное представление цифры состоит меньше, чем из \_\_\_ разрядов, дописать слева один или два нуля, чтобы всего получилось ровно \_\_\_ разряда
- каждую цифру шестнадцатеричного числа перевести в двоичную систему счисления

**2.35.** Переведите число  $D27_{16}$  в двоичную систему счисления:

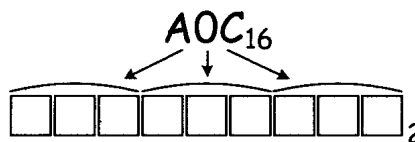
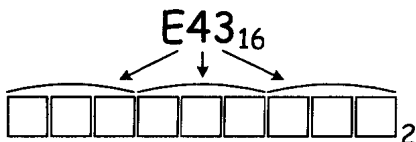


$$D_{16} = 13 = 8 + 4 + 1 = 2^3 + 2^2 + 2^0 = 1101_2$$

$$2 = 2^1 = \square\square\square\square_2$$

$$7 = \square + 2^1 + 1 = 2^2 + 2^1 + 2^0 = \square\square\square\square_2$$

**2.36.** Переведите числа  $E43_{16}$  и  $A0C_{16}$  в двоичную систему счисления:



**2.37.** Дано  $a = D7_{16}$ ,  $b = 331_8$ . Какое из чисел  $c$ , записанных в двоичной системе, отвечает условию  $a < c < b$ ?

- 1) 11011001
- 2) 11011100
- 3) 11010111
- 4) 11011000

*Рекомендация.* Так как все варианты ответов даны в двоичной системе счисления, имеет смысл перевести числа  $a$  и  $b$  в двоичную систему счисления, выбрать из них меньшее, прибавить к нему столбиком «1» и поискать результат среди вариантов ответа.





Это число должно быть не больше 25. Получаем неравенство:  $16 \cdot X + 5 \leq 25$ . Это равносильно тому, что  $16 \cdot X \leq 20$ . При этом  $X$  — целое неотрицательное число. Простым подбором получаем, что нас устраивают  $X = 0$  и  $X = 1$ . Значит, искомые числа:  $16 \cdot 0 + 5$  и  $16 \cdot 1 + 5$ . То есть 5 и 21.

**2.51.** Укажите через запятую в порядке возрастания все десятичные числа, не превосходящие 25, запись которых в двоичной системе счисления оканчивается на 110.

<i>Ответ:</i>	
---------------	--

**2.52.** Укажите через запятую в порядке возрастания все десятичные числа, не превосходящие 25, запись которых в троичной системе счисления оканчивается на 21.

<i>Ответ:</i>	
---------------	--

**2.53.** Укажите через запятую в порядке возрастания все десятичные числа, не превосходящие 35, запись которых в системе счисления с основанием пять оканчивается на 13.

<i>Ответ:</i>	
---------------	--

**2.54.** Укажите через запятую в порядке возрастания все десятичные числа, не превосходящие 25, запись которых в двоичной системе счисления оканчивается на 101.

<i>Ответ:</i>	
---------------	--

**2.55.** Укажите через запятую в порядке возрастания все основания систем счисления, в которых запись числа 23 оканчивается на 2.

<i>Ответ:</i>	
---------------	--

*Пример решения.* При переводе из десятичной в другую систему счисления нужно делить нацело с остатком исходное число на основание нужной системы счисления. При этом самый первый получающийся остаток — это самый младший разряд результирующего числа. То есть именно на него результат будет оканчиваться. В данном случае остаток будет равен двум. Это означает, что исходное число 23 должно быть (по определению целочисленного деления с остатком) представлено в виде:  $23 = nk + 2$ . Где  $n$  и  $k$  — натуральные числа. Вычитая из обеих частей равенства 2, получаем  $21 = nk$ . То есть достаточно найти все делители числа 21. При этом нужно не забыть, что остаток не может быть меньше делителя (основания системы счисления). Значит, нас устраивают все делители числа 21, которые больше двух. Это 3, 7, 21.





**2.62.** В системе счисления с некоторым основанием число 30 записывается в виде 110. Укажите это основание.

Ответ:	
--------	--

**2.63.** В системе счисления с некоторым основанием число 21 записывается в виде 111. Укажите это основание.

Ответ:	
--------	--

**2.64.** В системе счисления с некоторым основанием число 57 записывается в виде 321. Укажите это основание.

Ответ:	
--------	--

**Напоминание.** Для правильного осуществления действий в системах счисления необходимы знания 1-го класса — как складывать и вычитать числа «столбиком». Вспомним это. Пусть необходимо сложить числа 2465 и 637. Обычные десятичные числа.

Если вы в лоб примените наработанные с детства методы в другой системе счисления — вы гарантированно ошибетесь. Например, с первого взгляда очевидно, что  $5 + 7 = 12$ . Но вот  $5_8 + 7_8$  вовсе уже не равно  $12_8$ .  $5_8 + 7_8 = 14_8$ . Не поленитесь, аккуратно разберите следующий пример — как это нужно делать по действиям.

Сначала запишем оба исходных числа друг под другом, выровняв их по правому краю (по младшему разряду):

$$\begin{array}{r} 2465 \\ + 637 \\ \hline \end{array}$$

Начинаем с младшего разряда (самого правого). Сложим цифры.  $5 + 7 = 12$ . Цифру «2» запишем в этом разряде, «1» (которая на самом деле 10) — это перенос в следующий разряд:

$$\begin{array}{r} 2465 \\ + 637 \\ \hline 2 \\ \wedge \end{array}$$

В следующем разряде складываем  $6 + 3 + 1$  (из предыдущего разряда) = 10.

Цифру «0» пишем в этом разряде, «1» — перенос в следующий разряд:

$$\begin{array}{r} 2465 \\ + 637 \\ \hline 02 \\ \wedge \wedge \end{array}$$

Складываем  $4 + 6 + 1 = 11$ . «1» пишем, «1» — перенос в следующий разряд:

$$\begin{array}{r} 2465 \\ + 637 \\ \hline 102 \\ \wedge \wedge \wedge \\ 25 \end{array}$$

В самом левом (старшем) разряде стоит «2», плюс «1» перенесли из предыдущего разряда. Итого «3»:

$$\begin{array}{r} 2465 \\ + 637 \\ \hline 3102 \\ \wedge \wedge \wedge \end{array}$$

Теперь попробуем сделать то же самое, но в восьмеричной системе счисления:

$$2465_8 + 637_8$$

Запишем оба исходных числа друг под другом, выровняв их по правому краю (по младшему разряду):

$$\begin{array}{r} 2465_8 \\ + 637_8 \end{array}$$

Начинаем с младшего разряда (самого правого). Сложим цифры.  $5 + 7 = 12$ . Но вот только это совсем теперь не значит, что «2» — пишем, а «1» — переносим. Потому что получившееся у нас 12 — это десятичное число. А нам нужно восьмеричное. Поэтому мы переводим 12 в восьмеричную систему счисления. В общем случае нужно бы, конечно, делить «уголком» и выписывать остатки от деления. Но при сложении двух чисел в любой системе счисления не может получиться перенос в следующий разряд больше, чем «1». Поэтому результат деления всегда будет давать «1» в качестве частного (если, конечно, случился перенос). А это значит, что вместо деления «уголком» достаточно просто вычесть из получившейся суммы основание системы счисления. В данном случае,  $12 - 8 = 4$ . То есть «4» пишем, «1» — переносим:

$$\begin{array}{r} 2465_8 \\ + 637_8 \\ \hline 4_8 \\ \wedge \end{array}$$

В следующем разряде складываем  $6 + 3 + 1$  (из предыдущего разряда) = 10. Это  $\geq 8$ . Значит, перенос есть. Вычитаем 8:  $10 - 8 = 2$ .

Цифру «2» пишем в этом разряде, «1» — перенос в следующий разряд:

$$\begin{array}{r} 2465_8 \\ + 637_8 \\ \hline 24_8 \\ \wedge \wedge \end{array}$$

Складываем  $4 + 6 + 1 = 11$ . Перенос есть. Вычитаем 8. «3» пишем, «1» — перенос в следующий разряд:

$$\begin{array}{r} 2465_8 \\ + 637_8 \\ \hline 324_8 \\ \wedge \wedge \wedge \end{array}$$

В самом левом (старшем) разряде стоит «2», плюс «1» перенесли из предыдущего разряда. Итого «3». Переноса нет:

$$\begin{array}{r} 2465_8 \\ + 637_8 \\ \hline 3324_8 \\ \wedge \wedge \wedge \end{array}$$

**2.65.** Чему равна сумма чисел  $43_8$  и  $56_8$  (в восьмеричной системе счисления)?

Ответ:	
--------	--



ния. Если сумма двух исходных чисел в двоичной системе счисления не совпадет с двоичным вариантом, ее нужно будет перевести (группированием разрядов) в восьмеричную систему счисления и сравнить с другими вариантами. Если и здесь не будет совпадений — перевести сумму в шестнадцатеричную систему счисления и сравнить с имеющимся вариантом.

**Общая рекомендация к заданиям части А.** Если у вас отпали 3 варианта из четырех, не полнитесь, проверьте 4-й вариант на соответствие условию задачи. Этим вы сильно уменьшите вероятность совершения ошибки.

**2.70.** Чему равна сумма чисел  $52_8$  и  $74_{16}$ ?

- 1)  $126_8$                       2)  $5E_{16}$                       3)  $1011110_2$                       4)  $236_8$

**2.71.** Чему равна сумма чисел  $61_8$  и  $45_{16}$ ?

- 1)  $75_{16}$                       2)  $1110110_2$                       3)  $165_8$                       4)  $177_8$

**2.72.** Чему равна сумма чисел  $70_8$  и  $23_{16}$ ?

- 1)  $173_8$                       2)  $7B_{16}$                       3)  $133_8$                       4)  $1111011_2$

**2.73.** Чему равна сумма чисел  $23_8$  и  $67_{16}$ ?

- 1)  $7A_{16}$                       2)  $1111000_2$                       3)  $170_8$                       4)  $38_{16}$

**2.74.** Вычислите сумму чисел  $x$  и  $y$ , при  $x = 1D_{16}$ ,  $y = 72_8$ . Ответ приведите в двоичной системе счисления.

Ответ:	
--------	--

*Рекомендация.* Если нужно найти сумму чисел, представленных в разных системах счисления, то нужно привести их в одну систему счисления. Если нужно сложить число в шестнадцатеричной и в восьмеричной системах счисления, проще всего перевести оба числа в двоичную систему счисления.

**2.75.** Вычислите сумму чисел  $x$  и  $y$ , при  $x = A6_{16}$ ,  $y = 75_8$ . Ответ приведите в двоичной системе счисления.

Ответ:	
--------	--

**2.76.** Чему равна сумма чисел  $43_8$  и  $56_{16}$ ? Ответ приведите в двоичной системе счисления.

Ответ:	
--------	--

**Напоминание.** После сложения перейдем к вычитанию. Вспомним, как вычесть из 2007 число 29. Запишем меньшее под большим, выравнивая по правой цифре.

Начинаем с младшего (правого) разряда. Пытаемся вычесть из семи девять. Не получается ( $9 > 7$ ). Значит, нужно занять в предыдущем разряде. Но там стоит ноль. Значит, нужно занять в еще более предыдущем разряде. Но там тоже ноль. Значит, занимаем в еще более предыдущем:

$$\begin{array}{r} - 2007 \\ \quad 29 \end{array}$$

Из старшего разряда заняли «1». Значит, там осталось еще «1», в следующий разряд «свалилось» 10. Но из него заняли «1» (осталось 9). В следующий разряд тоже «свалилось» 10, заняли «1», осталось 9. В младший разряд «свалилось» 10. Плюс там уже есть 7. Итого  $10 + 7 = 17$ . Из этих 17 теперь вычитаем 9:  $17 - 9 = 8$ . Это «8» пишем в младший разряд:

$$\begin{array}{r} - 2007 \\ \quad 29 \\ \hline \quad \quad 8 \end{array}$$

Переходим к следующему разряду. Там над «0» стоит точка. Это напоминание, что мы из этого разряда занимали. Значит, сверху осталось «9» (самая старшая цифра нашей системы счисления), вычитаем из нее «2»:  $9 - 2 = 7$ . Пишем это «7» в данный разряд:

$$\begin{array}{r} - 2007 \\ \quad 29 \\ \hline \quad 78 \end{array}$$

В следующем разряде вычитать уже ничего не нужно. Сверху стоит «0» с точкой над ним. Значит, это «9». Списываем это «9» вниз, в результат:

$$\begin{array}{r} - 2007 \\ \quad 29 \\ \hline 978 \end{array}$$

В старшем разряде «2», над которой стоит точка. Значит, из этих «2» осталось «1». Снизу вычитать уже ничего не надо, списываем эту «1» вниз, в результат:

$$\begin{array}{r} - 2007 \\ \quad 29 \\ \hline 1978 \end{array}$$

Рассмотрим то же действие, но теперь в шестнадцатеричной системе счисления:

$$2007_{16} - 29_{16}$$

Запишем меньшее под большим, выровняв по правой цифре.

Начинаем с младшего (правого) разряда. Пытаемся вычесть из семи девять. Не получается ( $9 > 7$ ). Значит, нужно занять в предыдущем разряде. Но там стоит ноль. Значит, нужно занять в еще более предыдущем разряде. Но там тоже ноль. Значит, занимаем в еще более предыдущем. Пока все как в десятичной системе счисления:

$$\begin{array}{r} - 2007_{16} \\ \quad 29_{16} \end{array}$$

Из старшего разряда заняли «1». Значит, там осталось еще «1», в следующий разряд «свалилось»... нет, не 10, как раньше. Ведь у нас не десятичная система счисления и вес каждого более старшего разряда не в 10 раз больше младшего, а в 16! То есть в следующий разряд «свалилось» 16. Но из него заняли «1» (осталось 15). В следующий разряд тоже «свалилось» 16, заняли «1», осталось 15. В младший разряд «свалилось» 16. Плюс там уже есть 7. Итого  $16 + 7 = 23$ . Из этих 23 теперь вычитаем 9:  $23 - 9 = 14$ . Но это «14» мы не пишем в младший разряд. Потому что у нас шестнадцатеричная система счисления и 14 — это должна быть одна цифра. То есть «E»:

$$\begin{array}{r} - 2007_{16} \\ \quad 29_{16} \\ \hline \quad \quad E_{16} \end{array}$$

Переходим к следующему разряду. Там над «0» стоит точка. Это напоминание, что мы из этого разряда занимали. Значит, сверху осталось 15, вычитаем из нее «2»:  $15 - 2 = 13$ . Это «D». Пишем это «D» в данный разряд:

$$\begin{array}{r} - \overset{\cdot}{2} \overset{\cdot}{0} \overset{\cdot}{0} \overset{\cdot}{7}_{16} \\ \quad \quad \quad \underline{2 \ 9_{16}} \\ \quad \quad \quad \text{D E}_{16} \end{array}$$

В следующем разряде вычитать уже ничего не нужно. Сверху стоит «0» с точкой над ним. Значит, это «F» (самая старшая цифра в шестнадцатеричной системе счисления). Списываем это «F» вниз, в результат:

$$\begin{array}{r} - \overset{\cdot}{2} \overset{\cdot}{0} \overset{\cdot}{0} \overset{\cdot}{7}_{16} \\ \quad \quad \quad \underline{2 \ 9_{16}} \\ \quad \quad \quad \text{F D E}_{16} \end{array}$$

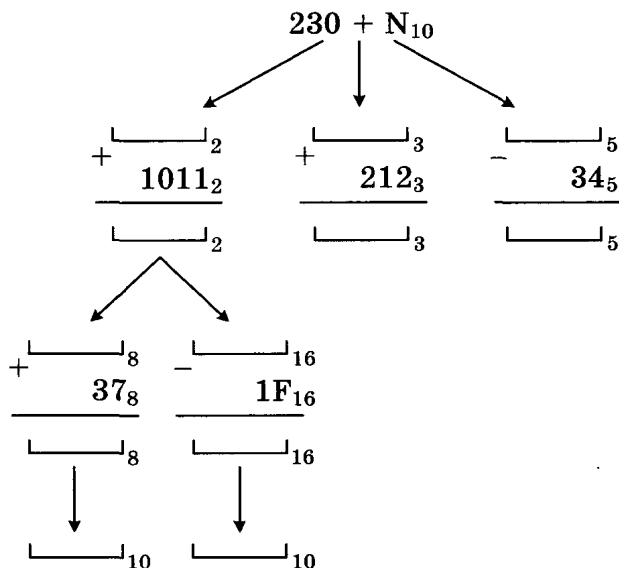
В старшем разряде «2», над которой стоит точка. Значит, из этих «2» осталось «1». Снизу вычитать уже ничего не надо, списываем эту «1» вниз, в результат:

$$\begin{array}{r} - \overset{\cdot}{2} \overset{\cdot}{0} \overset{\cdot}{0} \overset{\cdot}{7}_{16} \\ \quad \quad \quad \underline{2 \ 9_{16}} \\ \quad \quad \quad \text{1 F D E}_{16} \end{array}$$

**2.77.** Комплексное задание в системах счисления.

- 1) Получить исходное число:  $230 + N$  (выберите  $N$  от 1 до 25).
- 2) Перевести число (столбиком) в двоичную, троичную, пятеричную ССч.
- 3) К полученному двоичному числу прибавить  $1011_2$  (столбиком).
- 4) Результат перевести в восьмеричную и шестнадцатеричную ССч (объединением разрядов).
- 5) К полученному восьмеричному числу прибавить (столбиком)  $37_8$ .
- 6) Результат перевести в десятичную ССч.
- 7) Из полученного шестнадцатеричного числа отнять (столбиком)  $1F_{16}$ .
- 8) Результат перевести в десятичную ССч.
- 9) К полученному троичному числу прибавить (столбиком)  $212_3$ .
- 10) Из полученного пятеричного числа отнять (столбиком)  $34_5$ .

Схема выполнения задания:



# 3. Кодирование информации

## Основные понятия

Для удобства представления информации в компьютере все возможные виды информации переводятся в числовую форму, и эти числа хранятся в компьютере в двоичном виде.

Для кодирования текстовой информации используются специальные таблицы, в которых каждому символу (в т.ч. каждой букве) поставлен в соответствие некоторый номер. Так как эти номера хранятся в двоичном виде, принято говорить, что каждому символу соответствует свой двоичный код. Длина этих двоичных кодов одинакова для всех символов таблицы.

Основной (традиционной) таблицей кодирования символов является таблица ASCII (американский стандартный код обмена информацией, читается «АСКИ»). В ней для хранения каждого символа выделено 8 бит (1 байт). Таким образом, общее количество различных символов, которое кодируется таблицей ASCII, равно  $2^8 = 256$ .

Для кодирования русских букв, в зависимости от операционной системы, используется одна из разновидностей таблицы ASCII (так называемая *кодовая страница*): КОИ-8, Windows-1251, ISO, DOS, MAC.

Из-за ограниченности количества различных символов в таблице ASCII, на смену ей придумана кодовая таблица Unicode (Юникод). В ней для хранения каждого символа используется 16 бит (2 байта). Таким образом, общее количество различных символов, которое кодируется таблицей Unicode, равно  $2^{16} = 65\ 536$ .

Для кодирования информации о цвете используется таблица цветов. Количество цветов  $N$ , которое может быть закодировано при помощи  $i$  бит, вычисляется по формуле:

$$2^i = N.$$

Для кодирования полноцветных изображений используют цветовые модели (способ представления информации о цвете через несколько характеристик цвета). Одна из основных цветовых моделей — RGB. Она хранит информацию о цвете в виде яркости трех базовых цветов — красного (*red*), зеленого (*green*) и синего (*blue*), каждая из которых может принимать значения от 0 до 255. Эти три цвета «складываются» при восприятии их человеком, и он воспринимает их как некоторый оттенок цвета. Эта цветовая модель (RGB) называется *аддитивной*, потому что яркости базовых цветов в ней складываются (*add*). Чем больше суммарная яркость, тем светлее общий оттенок цвета и наоборот.

То есть, если все три яркости равны нулю, получается черный цвет, если 255, то белый. Если все три яркости одинаковы и лежат между 0 и 255, получается оттенок серого цвета. Если в коде цвета присутствует только одна яркость из трех, получается оттенок этого цвета.

Для кодирования растровых изображений цвет каждого пикселя изображения записывается своим кодом (одинаковой длины для всех пикселей изображения). Таким образом, общий объем памяти, необходимый для хранения неупакованного растрового изображения можно вычислить, умножив количество пикселей на длину кода цвета одного пикселя. Число пикселей в растровом изображении — ширина, умноженная на высоту. Так как длина кода цвета обычно записана в битах, а объем памяти принято считать в байтах, получившуюся величину нужно поделить на 8. Получаем формулу:

$$V = H \times W \times C/8,$$

где  $H$  и  $W$  — высота и ширина изображения в пикселях,

$C$  — число бит, используемое для хранения кода цвета одного пикселя,

$V$  — объем памяти для хранения этого изображения, в байтах.







порядке, что и буквы исходного сообщения, согласно этой таблице. Вспомнив процедуру перевода из двоичной системы счисления в шестнадцатеричную, сгруппируйте получившиеся двоичные цифры по 4 и вместо каждой группы напишите соответствующую шестнадцатеричную цифру.

**3.8.** Для кодирования букв А, Б, В, Г решили использовать двухразрядные последовательные двоичные числа (от 00 до 11, соответственно). Если таким способом закодировать последовательность символов ВАБВГАБГ и записать результат шестнадцатеричным кодом, то получится:

- 1) C0BC
- 2) 20123013
- 3) A8D5
- 4) 86C7

**3.9.** Сообщение состоит из символов М, А, Ы, Л, Р, У, точек и пробелов. Каким минимальным количеством байт можно закодировать сообщение, если оно состоит из 1000 таких символов?

<i>Ответ:</i>	
---------------	--

**Напоминание.** Если вероятность появления каждого символа сообщения не одинакова, то символы кодируют кодовыми словами разной длины. В этом случае нужно использовать такой способ кодирования, чтобы принятое сообщение можно было однозначным образом декодировать. Самый распространенный способ — использование префиксного кода (когда никакое кодовое слово не является началом никакого другого кодового слова).

**3.10.** Для 5 букв латинского алфавита заданы их двоичные коды (для некоторых букв из двух бит, для некоторых — из трех). Эти коды представлены в таблице:

А	В	С	D	E
000	01	100	10	011

Определить, какой набор букв закодирован двоичной строкой 0110100011000

- 1) EBCEA
- 2) BDDEA
- 3) VDCEA
- 4) EBAEA

**Рекомендация.** Выпишите передаваемое сообщение и начните разделять его на кодовые слова, подбирая их по таблице. Обратите внимание — код в данном случае постфиксный. Поэтому сообщение нужно однозначно декодировать справа налево.

**3.11.** Для 5 букв русского алфавита заданы их двоичные коды (для некоторых букв из двух бит, для некоторых — из трех). Эти коды представлены в таблице:

В	К	А	Р	Д
000	11	01	001	10

Из четырех полученных сообщений в этой кодировке только одно прошло без ошибки и может быть корректно декодировано. Найдите его:

- |                       |                       |
|-----------------------|-----------------------|
| 1) 110100000100110011 | 2) 111010000010010011 |
| 3) 110100001001100111 | 4) 110110000100110010 |

**Напоминание.** Для кодирования цвета используются цветовые модели.

**Цветовая модель** — способ сопоставления каждому используемому в модели цвету своего кодового слова.

Считается, что человеческий глаз способен различать 16 миллионов оттенков цвета.

Так как составить таблицу всех 16 миллионов цветов и назвать каждый представляется затруднительным и малоэффективным способом, придумали правила, по которым значение цвета хранится как числовое значение его определенных характеристик.

Самый распространенный такой способ — представлять цвет как интенсивность свечения трех базовых цветов, из которых можно составить любой из 16 миллионов цветов. (Вообще говоря, правильнее сказать «почти любой». Но это слишком сложно, чтобы изучать в курсе школьной информатики). Такая цветовая модель называется RGB. В ней для указания цвета используются яркости (интенсивности свечения) красного (*Red*), зеленого (*Green*) и синего (*Blue*) цветов. Каждая яркость может принимать значение от нуля (отсутствие свечения) до 255 (максимальное свечение). То есть всего по 256 возможных состояний.

Вы, вероятно, понимаете, что количество цветов (равновероятных событий) и количество бит для хранения кода цвета (количество информации) связано соотношением

$$2^{\text{число бит}} = \text{число цветов} \quad (3.1)$$

То есть для хранения интенсивности каждого базового цвета требуется 8 бит ( $2^{\text{число бит}} = 256$ ) или 1 байт. Значит, для хранения кода всего цвета требуется  $3 \cdot 8 = 24$  бита или 3 байта. Заметим, что  $2^{24}$  как раз примерно равно 16 миллионам.

В цветовой модели RGB работают все современные мониторы и телевизоры. В этой же цветовой модели представляется цвет в различных программах. В частности, для указания цвета фона web-страницы (в файле формата HTML) тоже используется цветовая модель RGB. Для этого используется атрибут `bgcolor` тега `<body>`, который записывается в виде `#RRGGVB`. Здесь RR — шестнадцатеричный код красной цветовой компоненты, GG — шестнадцатеричный код зеленой цветовой компоненты, VB — шестнадцатеричный код синей цветовой компоненты. Напоминаем, чем больше значение компоненты, тем больше интенсивность свечения соответствующего базового цвета. 00 — отсутствие свечения, FF — максимальное свечение ( $FF_{16} = 255_{10}$ ). Рассмотрим коды основных цветов, получающиеся из этих соображений:

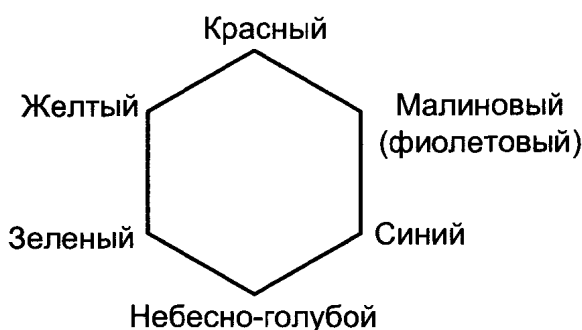
- #FF0000 красный (красная составляющая максимальная, остальные равны нулю)
- #000000 черный (ни одна компонента не светится)
- #FFFFFF белый (все составляющие максимальны и одинаковы, наиболее яркий цвет)
- #404040 темно-серый (все составляющие одинаковы, значит, ни один из базовых цветов не выделяется из остальных (серый); число  $40_{16}$  меньше, чем среднее значение яркости ( $80_{16}$ ), значит, это темный цвет)

**3.12.** Установите соответствие между кодами цвета и их названиями (соедините линиями)

Код цвета	Название цвета
а) #0000FF	1) средней яркости серый
б) #00FF00	2) почти черный
в) #808080	3) темно-синий
г) #C0C0C0	4) светло-серый
д) #010101	5) синий
е) #000080	6) зеленый

**Напоминание.** Принципы смешения базовых цветов отличаются от тех принципов смешения красок, которые привиты каждому ребенку на уроках рисования. Если смешать красный и синий, то все же получится фиолетовый. А вот если смешать желтый и синий, то получится вовсе не зеленый, а белый цвет. (Различие связано с тем, что на рисовании смешиваются краски, а не само-светящиеся источники. Для красок физический принцип получения цвета и другая цветовая модель — СМΥК.)

Чтобы попытаться понять эти принципы, рекомендуем запомнить схему смешения цветов:



Если смешиваются две базовые компоненты с равными яркостями, получится цвет, расположенный на схеме между ними, светлость/темнота которого зависит от этих яркостей (чем ярче, тем светлее, чем меньше яркость, тем темнее).

Если нужно получить, например, светло-красный — очевидно, что нужно увеличить общую яркость по сравнению с красным цветом. Увеличивать яркость красной компоненты (#FF0000) уже некуда (она максимальна). Если увеличить одну из оставшихся компонент — в цвете будет присутствовать этот оттенок. Значит, нужно увеличить яркости обеих оставшихся компонент — #FF8080.

**3.13.** Установите соответствие между кодами цвета и их названиями (соедините линиями)

Код цвета	Название цвета
а) #FFFF00	1) темно-малиновый
б) #FFFF80	2) малиновый
в) #40FF40	3) желтый
г) #FF00FF	4) светло-желтый
д) #800080	5) светло-зеленый

Если две компоненты смешиваются в разных пропорциях, получится оттенок между базовым цветом и тем, который указан между ними. Самый понятный пример — оранжевый цвет. Все понимают, что это нечто среднее между желтым и красным.

Желтый — #FFFF00, красный — #FF0000. Можно просто найти среднее арифметическое по каждой компоненте в отдельности:  $R = (FF + FF) / 2 = FF$ ,  $G = (FF + 00) / 2 \approx 80$ ,  $B = (0 + 0) / 2 = 0$ . То есть оранжевый — #FF8000.

**3.14.** Для кодирования цвета фона страницы Интернет используется атрибут `bgcolor = "#XXXXXX"`, где в кавычках задаются шестнадцатеричные значения интенсивности цветовых компонент в 24-битной RGB-модели. Какой цвет будет у страницы, заданной тэгом `<body bgcolor = "#FFFFFF">`?

- 1) белый                                        2) зеленый                                        3) красный                                        4) синий

**3.15.** Для кодирования цвета фона страницы Интернет используется атрибут `bgcolor = "#XXXXXX"`, где в кавычках задаются шестнадцатеричные значения интенсивности цветовых компонент в 24-битной RGB-модели. Какой цвет будет у страницы, заданной тэгом `<body bgcolor = "#404000">`?

- 1) серый                                        2) фиолетовый                                        3) темно-красный                                        4) коричневый

**3.16.** Для кодирования цвета фона страницы Интернет используется атрибут `bgcolor = "#XXXXXX"`, где в кавычках задаются шестнадцатеричные значения интенсивности цветовых компонент в 24-битной RGB-модели. Какой цвет будет у страницы, заданной тэгом `<body bgcolor = "#800080">`?

- 1) серый                                        2) фиолетовый                                        3) темно-красный                                        4) коричневый

**Напоминание.** Растровое изображение состоит из прямоугольного набора точек (пикселей), для каждой из которых хранится код цвета этой точки.

Формула соотношения количества бит, выделенных для хранения кода цвета и количества цветов, которые могут быть таким образом закодированы, мы уже приводили:

$$2^{\text{число бит}} = \text{число цветов}$$

**3.17.** Какой минимальный объем памяти (в битах) необходим для хранения одной точки изображения, в котором 16 различных цветов?


Ответ:

**3.18.** Какой минимальный объем памяти (в битах) необходим для хранения одной точки изображения, в котором 2 различных цвета?


Ответ:

**3.19.** Какой минимальный объем памяти (в битах) необходим для хранения одной точки изображения, в котором 256 различных цветов?


Ответ:

**3.20.** Какой минимальный объем памяти (в битах) необходим для хранения одной точки изображения, в котором 32 различных цвета?

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Ответ: \_\_\_\_\_

**3.21.** Какой минимальный объем памяти (в битах) необходим для хранения одной точки изображения, в котором 128 различных цветов?

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Ответ: \_\_\_\_\_

**3.22.** Какое наибольшее количество различных цветов можно закодировать, используя 4 бита?

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Ответ: \_\_\_\_\_

**3.23.** Какое наибольшее количество различных цветов можно закодировать, используя 3 бита?

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Ответ: \_\_\_\_\_

**3.24.** Какое наибольшее количество различных цветов можно закодировать, используя 1 бит?

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Ответ: \_\_\_\_\_

**3.25.** Какое наибольшее количество различных цветов можно закодировать, используя 8 бит?

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Ответ: \_\_\_\_\_

**3.26.** Какое наибольшее количество различных цветов можно закодировать, используя 16 бит?

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Ответ: \_\_\_\_\_

**Напоминание.** В растровом изображении (неупакованном) каждый пиксель кодируется одинаковым количеством бит. Изображение при этом состоит из прямоугольной таблицы пикселей (таблицы). Значит, чтобы посчитать количество бит, необходимых для хранения растрового изображения, нужно количество пикселей умножить на количество бит в одном пикселе.

Количество пикселей в изображении можно получить, перемножив количества пикселей по ширине и по высоте. При этом размер необходимой памяти принято измерять не в битах, а в байтах. Чтобы осуществить перевод в байты, нужно результат поделить на 8.

Итоговая формула для расчета количества памяти ( $V$ ), которое необходимо выделить для хранения неупакованного растрового изображения размером  $H \times W$  пикселей, в котором для хранения одного пикселя требуется  $C$  бит:

$$V = H \times W \times C / 8 \quad (3.2)$$

**3.27.** Для хранения растрового изображения размером  $32 \times 32$  пикселя потребовалось 512 байт памяти. Каково максимально возможное число цветов в палитре изображения?

- 1) 256                      2) 2                      3) 16                      4) 4

*Рекомендация.* Для формулы (1.2) в данной задаче имеются  $H = 32$ ,  $W = 32$ ,  $V = 512$ . Найдём  $C$ :

$$512 = 32 \times 32 \times C / 8 \Rightarrow C = 4$$

По формуле (1.1) получаем, что число цветов равно  $2^4 = 16$ .

**3.28.** Какой объем информации (в килобайтах) занимает неупакованное полноцветное ( $2^{24}$  цветов) растровое изображение размером  $1024 \times 512$  пикселей?

<i>Ответ:</i>	
---------------	--

**3.29.** Какой объем информации (в килобайтах) занимает неупакованное растровое изображение размером  $256 \times 512$  пикселей в цветовом режиме *HighColor* ( $2^{16}$  цветов)?

<i>Ответ:</i>	
---------------	--

**3.30.** Какой объем информации (в килобайтах) занимает неупакованное растровое изображение размером  $160 \times 2048$  пикселей в цветовом режиме *GrayScale* (256 оттенков серого)?

<i>Ответ:</i>	
---------------	--

**3.31.** Рассчитайте объем (в килобайтах) 4-цветного неупакованного растрового изображения размером  $256 \times 128$  пикселей.

<i>Ответ:</i>	
---------------	--



**3.32.** Рассчитайте объем (в килобайтах) 8-цветного неупакованного растрового изображения размером  $256 \times 512$  пикселей.

Ответ:																								

**3.33.** Какова ширина (в пикселях) прямоугольного 64-цветного неупакованного растрового изображения, занимающего на диске 1,5 мегабайта, если его высота вдвое меньше ширины?

Ответ:																								

**3.34.** Какова ширина (в пикселях) прямоугольного 16-цветного неупакованного растрового изображения, занимающего на диске 1 мегабайт, если его высота вдвое больше ширины?

Ответ:																								

**3.35.** Скольких различных цветов могут быть пиксели неупакованного растрового изображения, имеющего размер  $1024 \times 256$  пикселей и занимающего на диске 160 килобайт?

Ответ:																								

**3.36.** Скольких различных цветов могут быть пиксели неупакованного растрового изображения, имеющего размер  $128 \times 512$  пикселей и занимающего на диске 24 килобайта?

Ответ:																								

**3.37.** Скольких различных цветов могут быть пиксели неупакованного растрового изображения, имеющего размер  $128 \times 4096$  пикселей и занимающего на диске 448 килобайт?

Ответ:																								

## 4. Основы логики

### Основные понятия

В алгебре логики изучаются логические операции, производимые над высказываниями. Высказывания могут быть истинными или ложными. Применяя к простым высказываниям логические операции, можно строить составные высказывания.

Основными логическими операциями являются:

#### 1. Отрицание (инверсия, «не»)

Обозначение<sup>1</sup>:  $\neg$

Таблица истинности:

A	$\neg A$
0	1
1	0

#### 2. Логическое сложение (дизъюнкция, «или»)

Обозначение:  $\vee$

Таблица истинности:

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

#### 3. Логическое умножение (конъюнкция, «и»)

Обозначение:  $\wedge$

Таблица истинности:

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

Кроме основных логических операций используются также дополнительные, которые можно выразить через основные. Пример такой операции — *импликация*.

Обозначение:  $\rightarrow$

Таблица истинности:

A	B	$A \rightarrow B$
0	0	1
1	0	0
0	1	1
1	1	1

<sup>1</sup> В различных учебниках информатики обозначения и терминология иногда отличаются. Здесь и далее мы будем применять обозначения и терминологию, используемую в КИМ ЕГЭ.

При вычислении логического выражения операции выполняются в следующем порядке: отрицание, логическое умножение, логическое сложение, импликация. Как и в обычной алгебре, для изменения порядка операций используются скобки.

Импликация выражается через отрицание и логическое сложение следующим образом:

$$A \rightarrow B = \neg A \vee B$$

### Основные законы алгебры логики

Название закона	Формулировка
Переместительный закон	$A \vee B = B \vee A$ $A \wedge B = B \wedge A$
Сочетательный закон	$(A \vee B) \vee C = A \vee (B \vee C)$ $(A \wedge B) \wedge C = A \wedge (B \wedge C)$
Распределительный закон	$A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$ $A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$
Закон непротиворечия. Этот закон выражает тот факт, что высказывание не может быть одновременно истинным и ложным	$A \wedge \neg A = 0$
Закон исключенного третьего. Этот закон означает, что либо высказывание, либо его отрицание должно быть истинным	$A \vee \neg A = 1$
Закон двойного отрицания	$\neg(\neg A) = A$
Законы де Моргана	$\neg(A \vee B) = \neg A \wedge \neg B$ $\neg(A \wedge B) = \neg A \vee \neg B$

Основные свойства конъюнкции, дизъюнкции и импликации:

$$A \wedge A = A$$

$$A \wedge 0 = 0$$

$$A \vee A = A$$

$$A \vee 1 = 1$$

$$A \vee 0 = A$$

$$A \wedge 1 = A$$

$$0 \rightarrow A = 1$$

## Практическая часть

**4.1.** Расставьте над символами логических операций их номера в порядке выполнения операций при вычислении выражения.

а)  $\neg A \vee (B \vee \neg C)$

б)  $A \vee B \wedge C$

в)  $A \wedge \neg (B \vee \neg C) \vee D$

г)  $\neg (A \vee B) \rightarrow C$

**4.2.** Заполните таблицу истинности логических выражений.

a)

A	B	$A \wedge B$	$\neg (A \wedge B)$

б)

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

в)

A	B	C	$A \vee \neg B \vee C$

г)

A	B	C	$\neg A \wedge \neg B \wedge C$

**4.3.** Укажите значения переменных A и B, если:

- а)  $A \wedge B = 1$   $\Rightarrow$  A = \_\_\_ B = \_\_\_  
 б)  $A \vee B = 0$   $\Rightarrow$  A = \_\_\_ B = \_\_\_  
 в)  $A \rightarrow B = 0$   $\Rightarrow$  A = \_\_\_ B = \_\_\_  
 г)  $\neg (A \wedge B) = 0$   $\Rightarrow$  A = \_\_\_ B = \_\_\_

**4.4.** Найдите все пары значений  $A$  и  $B$ , при которых указанное выражение принимает ложное значение.

а)  $\neg A \vee \neg B$

Ответ: \_\_\_\_\_.

б)  $\neg A \rightarrow \neg B$

Ответ: \_\_\_\_\_.

в)  $(A \rightarrow B) \wedge B$

Ответ: \_\_\_\_\_.

г)  $\neg (A \rightarrow B) \wedge \neg B$

Ответ: \_\_\_\_\_.

*Пример.* Найдите все тройки значения  $L$ ,  $M$  и  $N$ , при которых указанное выражение принимает ложное значение.

$$L \rightarrow (M \vee N \vee L)$$

*Решение*

Из таблицы истинности импликации сразу получаем, что  $L = 0$ , а  $(M \vee N \vee L) = 1$ . Подставляем полученное значение  $L$  в выражение в скобках:

$$M \vee N \vee L = M \vee N \vee 0 = M \vee N = 1.$$

Из таблицы истинности дизъюнкции следует, что выражение  $M \vee N$  истинно тогда и только тогда, когда  $(M = 1, N = 0)$  или  $(M = 0, N = 1)$  или  $(M = 1, N = 1)$ . Поскольку  $L = 0$ , ответом будут три тройки:  $(L = 0, M = 1, N = 0)$ ,  $(L = 0, M = 0, N = 1)$ ,  $(L = 0, M = 1, N = 1)$ .

**4.5.** Найдите все тройки значения  $L$ ,  $M$  и  $N$ , при которых указанное выражение принимает истинное значение:

а)  $(L \rightarrow M) \wedge N$

Ответ: \_\_\_\_\_.

б)  $\neg ((L \rightarrow M) \rightarrow (L \wedge N))$

Ответ: \_\_\_\_\_.

в)  $(L \vee M) \rightarrow (\neg L \wedge N)$

Ответ: \_\_\_\_\_.

г)  $(L \vee \neg M) \rightarrow N$

Ответ: \_\_\_\_\_.

*Пример.* Найдите все целые числа  $X$ , для которых истинно высказывание:

$$(X > 5) \wedge (X^2 \leq 49)$$

*Решение*

Данное составное высказывание истинно, если одновременно истинны оба элементарных высказывания:

$$X > 5 \text{ и } X^2 \leq 49.$$

Решив эту элементарную систему неравенств (например, графически), получаем ответ.

Ответ: 6; 7.

**4.6.** Найдите все целые числа  $X$ , для которых истинно высказывание

а)  $(X > 5) \wedge (X + 3 < 11)$

Ответ:																				

б)  $\neg (X > 1) \wedge (X^2 < 40)$

Ответ:																				

в)  $\neg ((X > 5) \rightarrow (X > 8))$

Ответ:																				

г)  $\neg ((X < 6) \rightarrow (X < 2))$

Ответ:																				

### Задания в формате ЕГЭ

*Пример.* Какое логическое выражение равносильно выражению  $\neg (A \vee B) \vee \neg C$ ?

- 1)  $(\neg A \wedge \neg B) \vee \neg C$
- 2)  $\neg A \vee \neg B \vee \neg C$
- 3)  $A \vee B \wedge C$
- 4)  $(A \wedge B) \vee C$

*Решение*

Логические выражения называются равносильными, если при любых значениях, входящих в них переменных, значения этих выражений равны.

Преобразуем выражение  $\neg (A \vee B)$  в соответствии с законом де Моргана:  $\neg (A \vee B) = (\neg A \wedge \neg B)$ , поэтому правилен ответ под номером 1.

**4.7.**

а) Какое логическое выражение равносильно выражению  $\neg (\neg A \vee B) \vee \neg C$ ?

- 1)  $(A \wedge \neg B) \vee \neg C$
- 2)  $\neg A \vee B \vee \neg C$
- 3)  $A \vee \neg B \wedge C$
- 4)  $(\neg A \wedge B) \vee C$

б) Какое логическое выражение равносильно выражению  $(\neg A \vee B) \vee C$ ?

- 1)  $(A \wedge \neg B) \vee C$
- 2)  $\neg A \vee B \vee C$
- 3)  $A \vee \neg B \vee C$
- 4)  $(\neg A \wedge B) \vee C$

в) Какое логическое выражение равносильно выражению  $\neg (A \vee B \vee C)$ ?

- 1)  $A \vee B \vee C$
- 2)  $\neg A \vee \neg B \vee \neg C$
- 3)  $A \wedge B \wedge C$
- 4)  $\neg A \wedge \neg B \wedge \neg C$

г) Какое логическое выражение равносильно выражению  $(\neg A \vee \neg B) \vee \neg C$ ?

- 1)  $\neg (A \wedge B) \vee \neg C$
- 2)  $\neg A \wedge \neg B \vee \neg C$
- 3)  $\neg A \wedge (\neg B \wedge \neg C)$
- 4)  $(A \wedge B) \wedge C$

#### 4.8.

а) Какое логическое выражение равносильно выражению  $\neg (A \wedge \neg C) \wedge B$ ?

- 1)  $\neg A \vee C \wedge B$
- 2)  $\neg A \wedge B \vee \neg C$
- 3)  $(A \wedge \neg C) \wedge \neg B$
- 4)  $\neg A \vee B \wedge C$

б) Какое логическое выражение равносильно выражению  $\neg A \wedge \neg (\neg B \wedge C)$ ?

- 1)  $\neg A \wedge B \wedge \neg C$
- 2)  $\neg A \wedge \neg C \vee B$
- 3)  $\neg A \wedge B \vee \neg C$
- 4)  $(\neg A \wedge B) \wedge C$

в) Какое логическое выражение равносильно выражению  $\neg (A \wedge \neg B) \wedge \neg C$ ?

- 1)  $(\neg A \vee B) \vee \neg C$
- 2)  $(B \wedge \neg C) \vee (\neg A \wedge \neg C)$
- 3)  $(\neg A \vee B) \vee C$
- 4)  $(\neg A \vee B) \wedge C$

г) Какое логическое выражение равносильно выражению  $C \wedge \neg (\neg A \wedge B)$ ?

- 1)  $(A \vee \neg B) \vee \neg C$
- 2)  $(\neg A \vee B) \wedge C$
- 3)  $(\neg A \vee B) \vee C$
- 4)  $(\neg B \wedge C) \vee (A \wedge C)$

#### 4.9.

- а) Символом F обозначено одно из указанных ниже логических выражений от трех аргументов: X, Y, Z. Дан фрагмент таблицы истинности выражения F:

X	Y	Z	F
0	0	0	0
1	1	0	1
1	0	0	1

Какое выражение соответствует F?

- 1)  $\neg X \vee \neg Y \vee \neg Z$     2)  $X \wedge \neg Y \wedge \neg Z$     3)  $X \vee Y \vee Z$     4)  $X \wedge Y \wedge Z$

- б) Символом F обозначено одно из указанных ниже логических выражений от трех аргументов: X, Y, Z. Дан фрагмент таблицы истинности выражения F:

X	Y	Z	F
0	0	0	1
1	1	0	1
1	0	0	1

Какое выражение соответствует F?

- 1)  $\neg X \wedge \neg Y \wedge \neg Z$     2)  $X \wedge \neg Y \wedge \neg Z$     3)  $X \vee Y \vee \neg Z$     4)  $\neg X \vee Y \vee Z$

- в) Символом F обозначено одно из указанных ниже логических выражений от трех аргументов: X, Y, Z. Дан фрагмент таблицы истинности выражения F:

X	Y	Z	F
0	0	0	0
0	1	0	1
1	0	0	0

Какое выражение соответствует F?

- 1)  $\neg X \wedge Y \wedge \neg Z$     2)  $X \wedge \neg Y \wedge \neg Z$     3)  $X \vee Y \vee \neg Z$     4)  $\neg X \vee Y \vee Z$

- г) Символом F обозначено одно из указанных ниже логических выражений от трех аргументов: X, Y, Z. Дан фрагмент таблицы истинности выражения F:

X	Y	Z	F
0	1	0	0
1	1	1	0
1	0	0	1

Какое выражение соответствует F?

- 1)  $\neg X \wedge Y \wedge \neg Z$     2)  $X \wedge \neg Y \wedge \neg Z$     3)  $X \vee Y \vee \neg Z$     4)  $\neg X \vee Y \vee Z$



#### 4.10.

- а) Символом  $F$  обозначено одно из указанных ниже логических выражений от трех аргументов:  $X, Y, Z$ . Дан фрагмент таблицы истинности выражения  $F$ :

$X$	$Y$	$Z$	$F$
0	0	1	1
0	1	0	0
1	0	1	1

Какое выражение соответствует  $F$ ?

- 1)  $\neg X \vee \neg Y \wedge Z$       2)  $\neg X \vee Y \vee Z$       3)  $\neg X \wedge \neg Y \vee \neg Z$       4)  $X \wedge \neg Y \vee Z$

- б) Символом  $F$  обозначено одно из указанных ниже логических выражений от трех аргументов:  $X, Y, Z$ . Дан фрагмент таблицы истинности выражения  $F$ :

$X$	$Y$	$Z$	$F$
0	1	0	1
1	1	0	1
0	0	1	0

Какое выражение соответствует  $F$ ?

- 1)  $\neg X \vee Y \wedge \neg Z$       2)  $\neg X \vee Y \vee Z$       3)  $X \wedge \neg Z \vee Y$       4)  $\neg X \wedge \neg Z \vee \neg Y$

- в) Символом  $F$  обозначено одно из указанных ниже логических выражений от трех аргументов:  $A, B, C$ . Дан фрагмент таблицы истинности выражения  $F$ :

$A$	$B$	$C$	$F$
0	1	1	1
1	0	1	1
0	1	0	0

Какое выражение соответствует  $F$ ?

- 1)  $\neg A \wedge B \wedge C$       2)  $A \vee B \wedge C$       3)  $\neg A \wedge B \vee \neg C$       4)  $\neg A \vee B \vee \neg C$

- г) Символом  $F$  обозначено одно из указанных ниже логических выражений от трех аргументов:  $A, B, C$ . Дан фрагмент таблицы истинности выражения  $F$ :

$A$	$B$	$C$	$F$
0	1	1	1
1	1	0	0
0	1	0	1

Какое выражение соответствует  $F$ ?

- 1)  $\neg A \wedge B \wedge \neg C$       2)  $\neg A \vee B \vee \neg C$       3)  $A \vee B \wedge C$       4)  $\neg A \wedge B \vee C$

*Пример.* Найдите наименьшее целое число  $X$ , при котором истинно высказывание

$$((X > 7) \vee (X < 7)) \rightarrow (X > 8)$$

*Решение.* Импликация истинна в трех случаях:

- 1)  $0 \rightarrow 1$  (ложь  $\rightarrow$  истина)      2)  $0 \rightarrow 0$  (ложь  $\rightarrow$  ложь)      3)  $1 \rightarrow 1$  (истина  $\rightarrow$  истина)

Рассмотрим 1-й случай:

Предположим, что  $(X > 7) \vee (X < 7) = \text{ложь}$  и  $X > 8 = \text{истина}$

$(X > 7) \vee (X < 7) = \text{ложь}$  только при  $X = 7$ .  $7 < 8$ , поэтому в данном случае решения нет.

Рассмотрим 2-й случай:

Предположим, что  $(X > 7) \vee (X < 7) = \text{ложь}$  и  $X > 8 = \text{ложь}$ . В этом случае  $X = 7$  является единственным решением.

Рассмотрим 3-й случай:

Предположим, что  $(X > 7) \vee (X < 7) = \text{истина}$ , тогда  $X$  — любое целое число, кроме 7.

Из условия  $X > 8$  получаем, что наименьшим решением будет 9. Но во втором случае мы получили решение 7.  $7 < 9$ , поэтому окончательный ответ: 7.

*Второй вариант решения*

Для начала приведем выражение в более удобную форму, вспомнив, что  $(X > 7 \text{ ИЛИ } X < 7)$  в математике записывается как  $X \neq 7$ .  $(X \neq 7) \rightarrow (X > 8)$ .

Преобразуем данное выражение по законам алгебры логики.

Вспользуемся законом « $A \rightarrow B = \neg A \vee B$ »  $\neg (X \neq 7) \vee (X > 8) \Leftrightarrow (X = 7) \vee (X > 8)$ . Таким образом, решением является число 7 и все числа больше 8. Наименьшим из этих чисел является число 7.

**4.11.**

а) Найдите наименьшее целое число  $X$ , при котором истинно высказывание:

$((X > 3) \vee (X < 3)) \rightarrow (X > 4)$ .

Ответ:																			

б)  $A, B, C$  — целые числа, для которых истинно высказывание:

$\neg (A = B) \wedge ((B < A) \rightarrow (2C > A)) \wedge ((A < B) \rightarrow (A > 2C))$ .

Чему равно  $A$ , если  $C = 7, B = 16$ ?

Ответ:																			

в) Каково наибольшее целое число  $X$ , при котором истинно высказывание:

$(99 < X \cdot X) \rightarrow (X < (X - 1))$ ?

Ответ:																			

г) Укажите наименьшее целое число  $X$ , для которого истинно высказывание:

$\neg ((X > 7) \rightarrow (X > 8))$ .

Ответ:																			

#### 4.12.

а) Для какого слова истинно высказывание:

$\neg$  (Первая буква слова согласная  $\rightarrow$  (Вторая буква слова гласная  $\vee$  Последняя буква слова гласная))?

- 1) ГОРА                      2) БРИКЕТ                      3) ТРУБКА                      4) ПАРАД

б) Для какого слова ложно высказывание:

Первая буква слова согласная  $\rightarrow$  (Вторая буква слова гласная  $\wedge$  Последняя буква слова гласная)?

- 1) ЖАРА                      2) ОРДА                      3) ОГОРОД                      4) ПАРАД

в) Для какого слова истинно высказывание:

$\neg$  (Первая и последняя буквы слова согласные  $\rightarrow$  Первая и последняя буквы слова совпадают)?

- 1) КОМОК                      2) ПРИВЕТ                      3) ТРУБКА                      4) ОКНО

г) Для какого слова ложно высказывание:

Первая буква слова гласная  $\rightarrow$  (Вторая буква слова гласная  $\vee$  Последняя буква слова гласная)?

- 1) ЖАРА                      2) ОРДА                      3) ОГОРОД                      4) ПАРАД

**4.13.** Найдите значения логических переменных A, B, C, D, при которых указанное логическое выражение ложно. Ответ запишите в виде строки из четырех символов: значений переменных A, B, C и D (в указанном порядке). Так, например, строка 0101 соответствует тому, что  $A = 0, B = 1, C = 0, D = 1$ .

а)  $\neg (A \vee B \vee C) \rightarrow (C \vee \neg D)$

Ответ: \_\_\_\_\_.

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

Ответ: \_\_\_\_\_.

в)  $\neg B \rightarrow (\neg A \vee \neg C \vee D)$

Ответ: \_\_\_\_\_.

г)  $(A \vee C) \rightarrow (C \vee B \vee \neg D)$

Ответ: \_\_\_\_\_.



## Решение логических задач

Для задач, в условии которых сказано, что часть утверждений персонажей ложна, а часть истинна, удобно использовать *метод рассуждений*. Идея этого метода заключается в том, что делается предположение об истинности одного из утверждений, и далее, на основе этого предположения анализируются остальные утверждения. Анализ остальных утверждений может привести к двум случаям — либо задача оказалась решена, либо встретилось противоречие. Если встретилось противоречие, это означает, что первоначальная гипотеза об истинности одного из утверждений была неверна, и это утверждение на самом деле ложно. Далее, с учетом этой информации продолжаем анализ утверждений, пока не решим задачу.

*Пример.* На олимпиаде по информатике участвовали пятеро учеников: Андрей (А), Коля (К), Виктор (В), Егор (Е), Степан (С). Об итогах олимпиады имеется пять высказываний:

- 1) Второе место занял Андрей, а Егор оказался третьим.
- 2) Выиграл Виктор, а Коля поднялся на второе место.
- 3) Степан занял только второе место, а Виктор был последним.
- 4) Все-таки на первом месте был Егор, а Коля был четвертым.
- 5) Да, Коля был действительно четвертым, а Андрей вторым.

Если известно, что в каждом высказывании одно утверждение правильное, а другое нет, то кто занял второе место и на каком месте был Андрей?

Ответ запишите в виде первой буквы имени второго призера и через запятую, места, занятого Андреем.

## Решение

Предположим, что в 1-м высказывании истинна первая половина (второе место занял Андрей), а вторая ложна, т.е. Егор — не третий. Заметим, что Егор и не второй, так как мы предположили, что это место занято Андреем.

Рассмотрим 2-е высказывание. Так как мы предположили, что второе место занял Андрей, значит утверждение «Коля поднялся на второе место» — ложно, а истинно — «Выиграл Виктор».

Подведем промежуточные итоги. По нашей гипотезе Виктор — первый, Андрей — второй, Егор — четвертый или пятый, Коля — третий, четвертый или пятый.

Перейдем к 3-му высказыванию. Виктор не может быть последним (из нашей гипотезы следует, что он первый), следовательно, истинна первая половина высказывания «Степан занял только второе место», но это противоречит нашей гипотезе, согласно которой на втором месте Андрей. Значит, гипотеза была неверна, а верна противоположная ей: в 1-м высказывании ложна первая половина, а вторая истинна. Таким образом, мы теперь достоверно знаем, что Егор занял третье место, а Андрей не второй и не третий.

Посмотрим, в каких высказываниях еще упоминается место Егора. В первой части 4-го высказывания говорится, что Егор был на первом месте. Но мы уже точно знаем, что это неправда, значит, истинна вторая часть высказывания: «Коля был четвертым».

Аналогично заключаем из 2-го высказывания, что Виктор — первый.

Далее из 3-го высказывания получаем, что вторым был Степан.

Итак, тройка лидеров выглядит следующим образом:

1. Виктор
2. Степан
3. Егор

Из 5-го высказывания следует, что Коля был четвертым, следовательно, Андрею остается последнее место.

Перед тем как записывать ответ, лучше сделать проверку, убедившись, что действительно в каждом высказывании из условия задачи одно из утверждений истинно, а второе — ложно.

Ответ: С, 5.

Более формализованным способом решения логических задач является метод таблиц.

*Пример.* В бюро переводов приняли на работу троих сотрудников — Диму, Сашу и Юру. Каждый из них знает ровно два иностранных языка из следующего набора — немецкий, шведский, японский, китайский, французский и греческий, при этом каждым языком владеет только один переводчик.

Известно, что:

- 1) Ни Дима, ни Юра не знают японского
- 2) Переводчик со шведского старше переводчика с немецкого
- 3) Переводчик с китайского, переводчик с французского и Саша родом из одного города
- 4) Переводчик с греческого, переводчик с немецкого и Юра учились втроем в одном институте
- 5) Дима — самый молодой из всех троих, и он не знает греческого
- 6) Юра знает два европейских языка

В ответе запишите первую букву имени переводчика с шведского языка и через запятую, первую букву имени переводчика с китайского.

*Решение*

Составим таблицу, в строках которой имена переводчиков, в столбцах — языки. Знание переводчиком языка будем отмечать единицей в соответствующей клеточке, незнание — нулем.

Из 1-го условия:

	Немецкий	Шведский	Японский	Китайский	Французский	Греческий
Дима			0			
Юра			0			
Саша						

Дополним таблицу информацией из 2-го и 5-го условия:

	Немецкий	Шведский	Японский	Китайский	Французский	Греческий
Дима		0	0			0
Юра			0			
Саша						

Дополним таблицу информацией из 3-го условия:

	Немецкий	Шведский	Японский	Китайский	Французский	Греческий
Дима		0	0			0
Юра			0			
Саша				0	0	

Дополним таблицу информацией из 4-го условия:

	Немецкий	Шведский	Японский	Китайский	Французский	Греческий
Дима		0	0			0
Юра	0		0			0
Саша				0	0	

Из таблицы видно что, ни Дима, ни Юра не знают японского. Следовательно, его должен знать Саша. Аналогично получаем, что Саша знает греческий.

	Немецкий	Шведский	Японский	Китайский	Французский	Греческий
Дима		0	0			0
Юра	0		0			0
Саша			1	0	0	1

Поскольку каждый переводчик знает только два иностранных языка, в строке Саши остальные клетки заполняем нулями.

	Немецкий	Шведский	Японский	Китайский	Французский	Греческий
Дима		0	0			0
Юра	0		0			0
Саша	0	0	1	0	0	1

Из таблицы видно, что Дима должен знать немецкий, а Юра – шведский

	Немецкий	Шведский	Японский	Китайский	Французский	Греческий
Дима	1	0	0			0
Юра	0	1	0			0
Саша	0	0	1	0	0	1

Из 6-го условия получаем, что Юра не знает китайского. Из таблицы следует, что тогда в качестве второго языка он должен знать французский.

	Немецкий	Шведский	Японский	Китайский	Французский	Греческий
Дима	1	0	0			0
Юра	0	1	0	0	1	0
Саша	0	0	1	0	0	1

Окончательный вид таблицы:

	Немецкий	Шведский	Японский	Китайский	Французский	Греческий
Дима	1	0	0	1	0	0
Юра	0	1	0	0	1	0
Саша	0	0	1	0	0	1

Рекомендуется выполнить проверку своего решения и убедиться, что полученная таблица не противоречит условиям задачи.

**4.15.**

- а) Три школьника, Петя (П), Толя (Т) и Сергей (С), оставшиеся в классе на перемене, были вызваны к директору по поводу разбитого в это время окна в кабинете. На вопрос директора о том, кто это сделал, мальчики ответили следующее:

Петя: «Я не бил окно, и Толя тоже...»

Толя: «Петя не разбивал окно, это Сергей разбил футбольным мячом!»

Сергей: «Я не делал этого, стекло разбил Петя».

Стало известно, что один из ребят сказал чистую правду, второй в одной части заявления соврал, а другое его высказывание истинно, а третий оба факта искажил. Зная это, директор смог докопаться до истины.

Кто разбил стекло в классе? В ответе запишите только первую букву имени.

<i>Ответ:</i>
---------------

- б) Трое друзей спорили, как распределятся места среди трех команд школьного первенства.

— Первым будет «Рассвет», а «Комета» займет последнее место, — сказал Семён.

— Победителем будет «Комета», а «Спутник» будет третьим, — сказал Василий.

— Первым будет кто угодно, но только не «Рассвет», — заключил Андрей.

После завершения соревнований оказалось, что предположения двоих ребят оправдались, а третий был неправ.

Как распределились места в первенстве? В ответе укажите последовательно первые буквы названий команд, занявших первое, второе и третье место.

<i>Ответ:</i>
---------------

- в) В бюро переводов приняли на работу троих сотрудников Ивана, Антона и Петра. Каждый из них знает ровно два иностранных языка из следующего набора – немецкий, шведский, японский, китайский, французский и греческий, при этом каждым языком владеет только один переводчик.

Известно, что:

- 1) Петр самый высокий
- 2) Переводчик с французского ниже ростом переводчика со шведского
- 3) Переводчик со шведского, переводчик с французского и Антон родом из одного города
- 4) Переводчик с японского, переводчик с китайского и Петр учились втроем в одном институте
- 5) Антон не знает ни китайского, ни греческого







г) -2008 ++ ·

Ответ:	
--------	--

**5.2.** Составьте для устройства, описанного в предыдущем задании, программу, длиной не более 6 символов, преобразующую заданное исходное число в заданный результат:

а) из 0 получить 17.

Ответ: \_\_\_\_\_.

б) из 3 получить 30.

Ответ: \_\_\_\_\_.

в) из -5 получить -30.

Ответ: \_\_\_\_\_.

г) из -2 получить 5.

Ответ: \_\_\_\_\_.

*Рассмотрим пример решения задачи с исполнителем РОБОТ*

Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости:

вверх	вниз	влево	вправо
-------	------	-------	--------

При выполнении этой команды РОБОТ перемещается на соответствующую клетку.

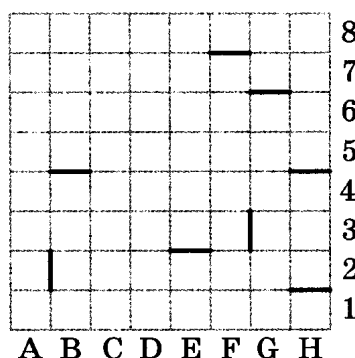
Команды проверки истинности условия на наличие стены у той клетки, где он находится:

сверху свободно	снизу свободно	слева свободно	справа свободно
-----------------	----------------	----------------	-----------------

Если РОБОТ начнет движение в сторону стены, то он разрушится.

а) Сколько клеток данного лабиринта соответствуют требованию, что, выполнив предложенную программу, РОБОТ остановится в той же клетке, с которой он начал движение?

НАЧАЛО  
 ПОКА справа свободно  
     ДЕЛАТЬ вправо  
 ПОКА снизу свободно  
     ДЕЛАТЬ вниз  
 ПОКА слева свободно  
     ДЕЛАТЬ влево  
 ПОКА сверху свободно  
     ДЕЛАТЬ вверх  
 КОНЕЦ



В ответе запишите число — количество таких клеток, а далее, через запятые, их адреса (сначала идет латинская буква столбца, а затем цифра строки).

Например, нижний левый угол лабиринта имеет адрес А1.

### Решение

Сначала определим свойства клеток, отвечающих требованию задачи.

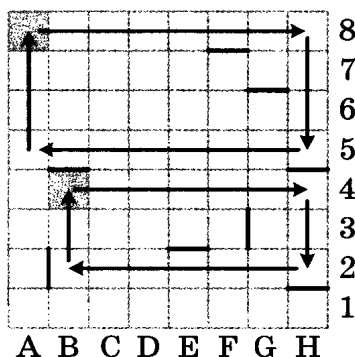
Поскольку последний цикл программы РОБОТА

**ПОКА сверху свободно**

**ДЕЛАТЬ вверх**

искомые клетки должны иметь границу сверху.

Таких клеток 14: вся верхняя горизонталь и клетки F7, G6, B4, H4, E2 и H1.



Предпоследний цикл программы РОБОТА

**ПОКА слева свободно**

**ДЕЛАТЬ влево**

Это значит, что в вертикали искомой клетки ниже нее обязательно должна быть клетка с левой границей, от которой «отскочил» РОБОТ перед тем, как пойти вверх.

Из ранее найденных клеток этому условию удовлетворяют 4 — A8, B8, B4, G6

При этом B8 надо тоже отбросить, поскольку в ее вертикали на пути от клетки B2 стоит горизонтальная граница между B4 и B5.

Далее используем остальные фрагменты программы. Выполнив программу целиком для трех оставшихся клеток, получаем, что условию задачи удовлетворяют только две из них — A8 и B4

Ответ: 2, A8, B4.

### 5.3. Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости:

вверх	вниз	влево	вправо
-------	------	-------	--------

При выполнении любой из этих команд РОБОТ перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

сверху свободно	снизу свободно	слева свободно	справа свободно
-----------------	----------------	----------------	-----------------

Цикл

ПОКА < условие > команда

выполняется, пока условие истинно, иначе происходит переход на следующую строку. Если РОБОТ начнет движение в сторону стены, то он разрушится и программа прервется.

Нарисуйте путь робота в лабиринте, если он начал движение по программе из указанной клетки.

а)

Лабиринт	Программа	Начальная клетка
<p>6 5 4 3 2 1 A B C D E F</p>	<p>НАЧАЛО ПОКА &lt; справа свободно &gt; вправо ПОКА &lt; снизу свободно &gt; вниз ПОКА &lt; слева свободно &gt; влево ПОКА &lt; сверху свободно &gt; вверх КОНЕЦ</p>	<p>B4</p>

б)

Лабиринт	Программа	Начальная клетка
<p>6 5 4 3 2 1 A B C D E F</p>	<p>НАЧАЛО ПОКА &lt; слева свободно &gt; влево ПОКА &lt; сверху свободно &gt; вверх ПОКА &lt; справа свободно &gt; вправо ПОКА &lt; снизу свободно &gt; вниз КОНЕЦ</p>	<p>F1</p>

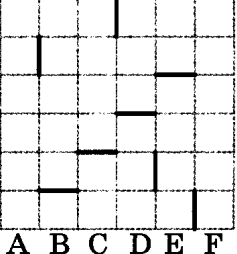
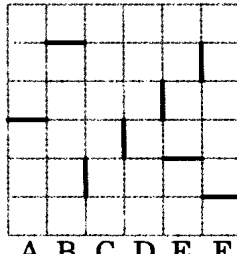
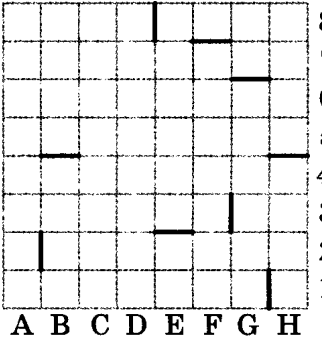
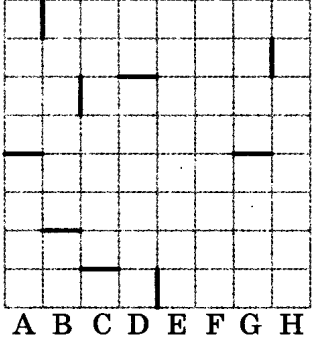
в)

Лабиринт	Программа	Начальная клетка
<p>6 5 4 3 2 1 A B C D E F</p>	<p>НАЧАЛО ПОКА &lt; слева свободно &gt; влево ПОКА &lt; сверху свободно &gt; вверх ПОКА &lt; справа свободно &gt; вправо ПОКА &lt; снизу свободно &gt; вниз КОНЕЦ</p>	<p>D3</p>

г)

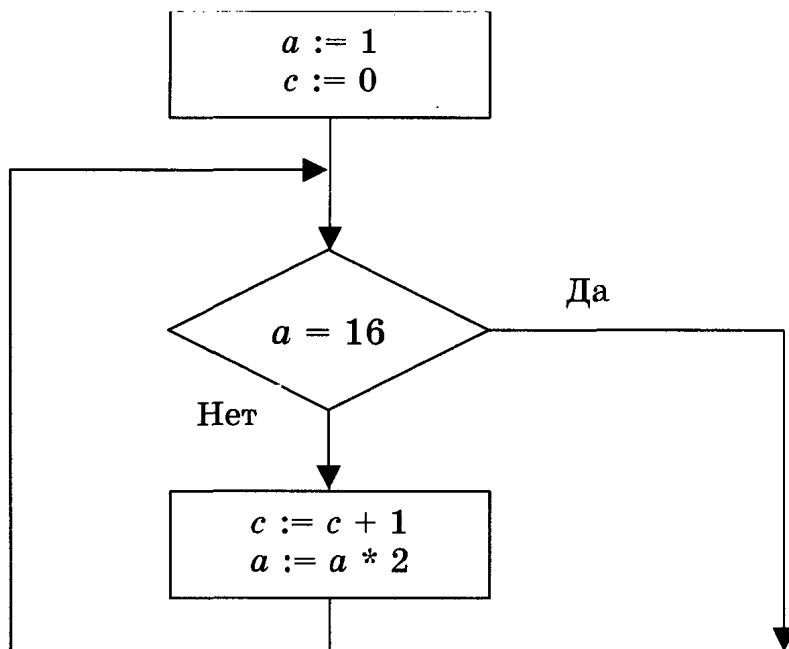
Лабиринт	Программа	Начальная клетка
<p>6 5 4 3 2 1 A B C D E F</p>	<p>НАЧАЛО ПОКА &lt; слева свободно &gt; влево ПОКА &lt; справа свободно &gt; вправо ПОКА &lt; снизу свободно &gt; вниз ПОКА &lt; сверху свободно &gt; вверх КОНЕЦ</p>	<p>C5</p>

**5.4.** Для исполнителя РОБОТ, описанного в предыдущем задании составить программу не более чем из семи строк (включая строки «НАЧАЛО» и «КОНЕЦ»), обеспечивающую нужное перемещение по заданному лабиринту

а)	<p style="text-align: center;"><b>Лабиринт</b></p> 	<b>Программа</b>	<b>Начальная клетка</b>	<b>Конечная клетка</b>
			A1	F6
б)	<p style="text-align: center;"><b>Лабиринт</b></p> 	<b>Программа</b>	<b>Начальная клетка</b>	<b>Конечная клетка</b>
			C2	E5
в)	<p style="text-align: center;"><b>Лабиринт</b></p> 	<b>Программа</b>	<b>Начальная клетка</b>	<b>Конечная клетка</b>
			H1	D8
г)	<p style="text-align: center;"><b>Лабиринт</b></p> 	<b>Программа</b>	<b>Начальная клетка</b>	<b>Конечная клетка</b>
			D1	C2

Для решения задач на определение значения переменной после выполнения фрагмента программы или блок-схемы удобно использовать таблицу значений переменных.

*Пример.* Дана блок-схема алгоритма. Требуется найти значение переменной  $c$  после завершения алгоритма.



*Решение*

*1-й способ*

Составим таблицу значений переменных, добавив в нее для удобства результаты вычисления логического выражения.

№ шага	Значение $a$	Значение $c$	$a = 16 ?$
0	1	0	нет
1	1	1	
2	2	1	нет
3	2	2	
4	4	2	нет
5	4	3	
6	8	3	нет
7	8	4	
7	16	4	да

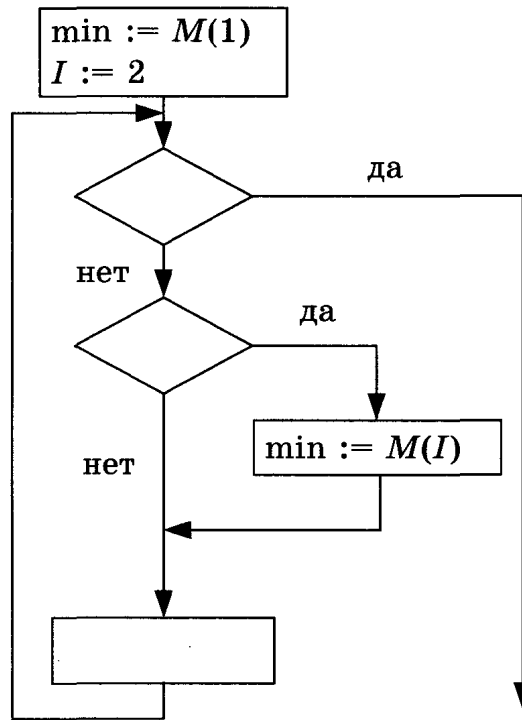
Итак, ответ  $c = 4$ .

*2-й способ*

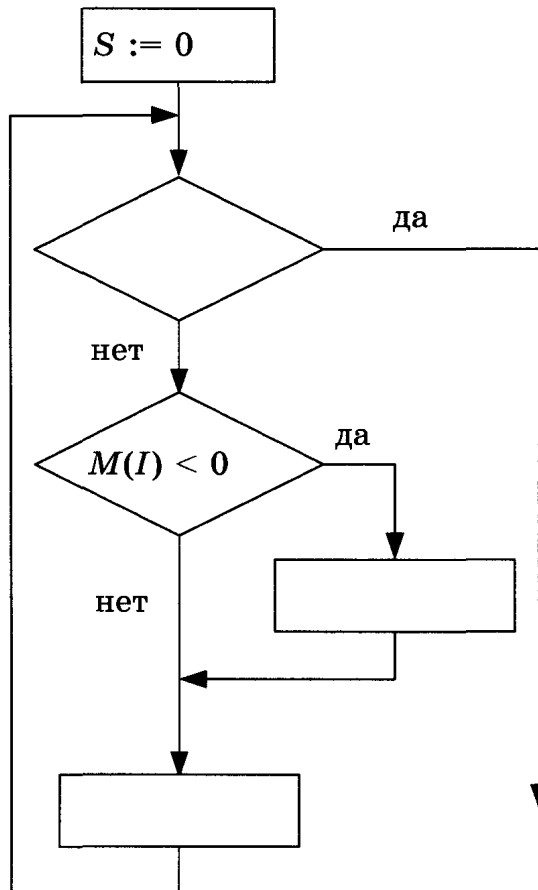
Внимательно проанализировав блок-схему можно заметить, что она иллюстрирует последовательное удвоение переменной  $a$ , изначально равной единице, до тех пор, пока она не достигнет значения  $16 = 2^4$ . Отсюда,  $c = \log_2 16 = 4$ .

5.5. Впишите во фрагмент блок-схемы пропущенные инструкции, необходимые для правильного решения поставленной задачи.

а) Найти минимальный элемент массива  $M$  из 24 элементов.

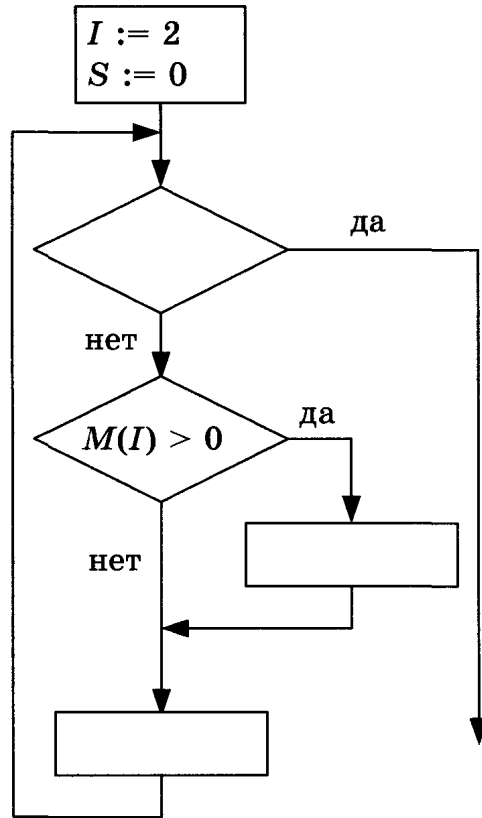


б) Найти сумму отрицательных элементов массива  $M$  из 24 элементов.

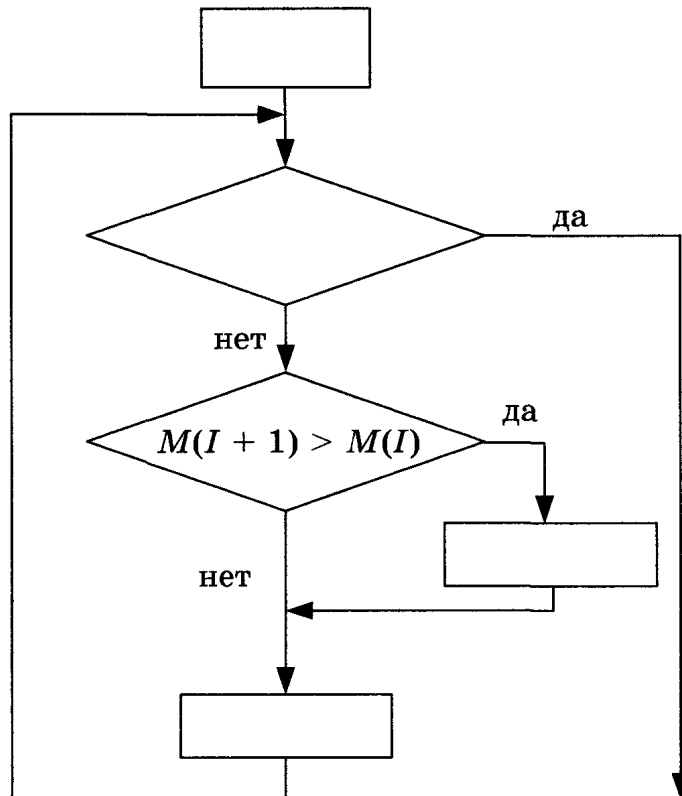




- в) Найти сумму положительных элементов массива  $M$  из 22 элементов, имеющих четные номера (нумерация элементов массива начинается с единицы).

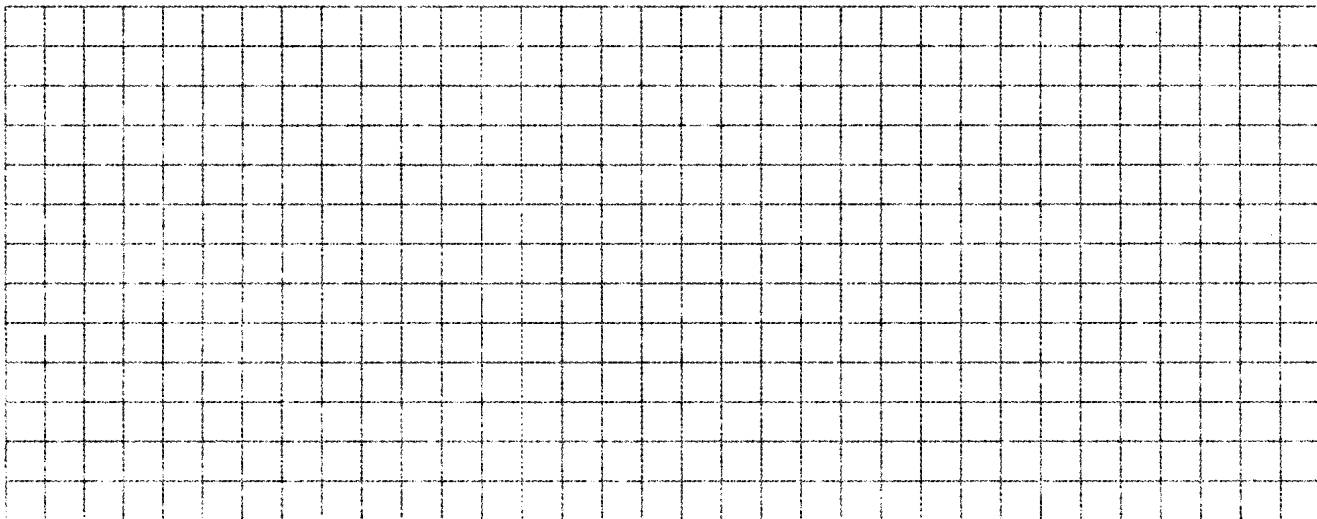


- г) В массиве  $M$  45 элементов. Найти, сколько подряд идущих элементов этого массива, начиная с первого, образуют возрастающую последовательность.

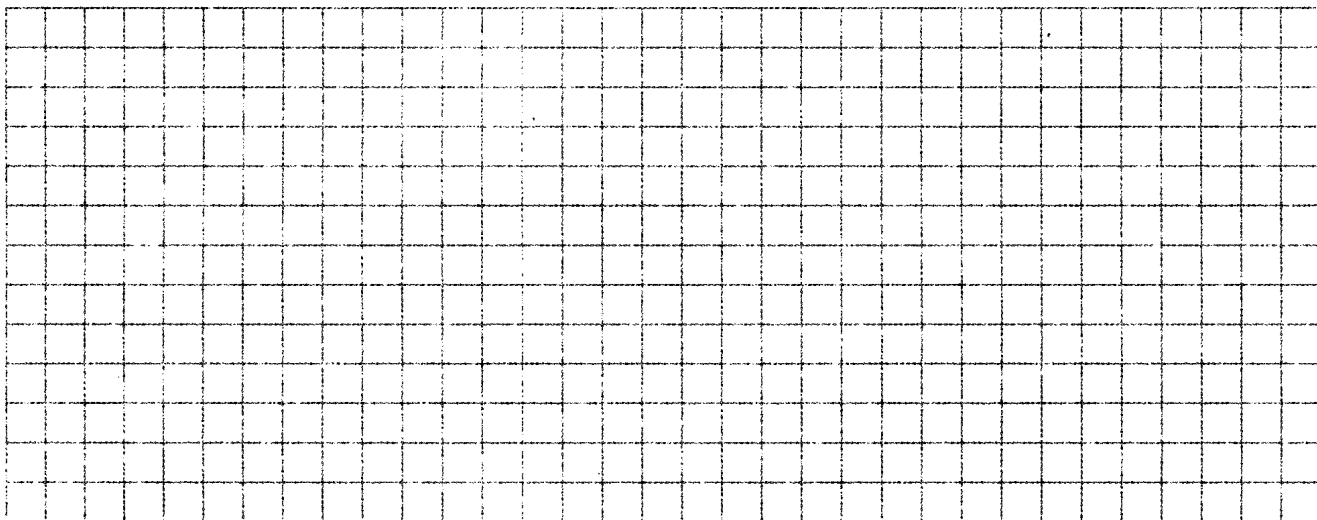


**5.6.** Составить и записать на естественном языке алгоритм, выполняющий:

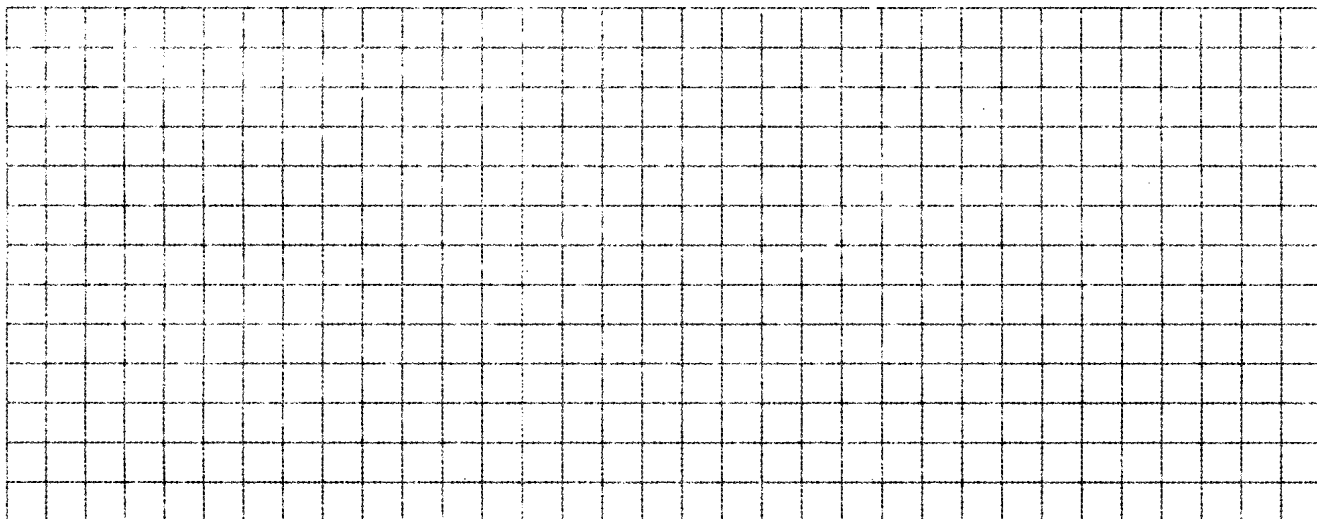
а) решение уравнения  $ax = b$ , где  $a$  и  $b$  — числовые исходные данные;



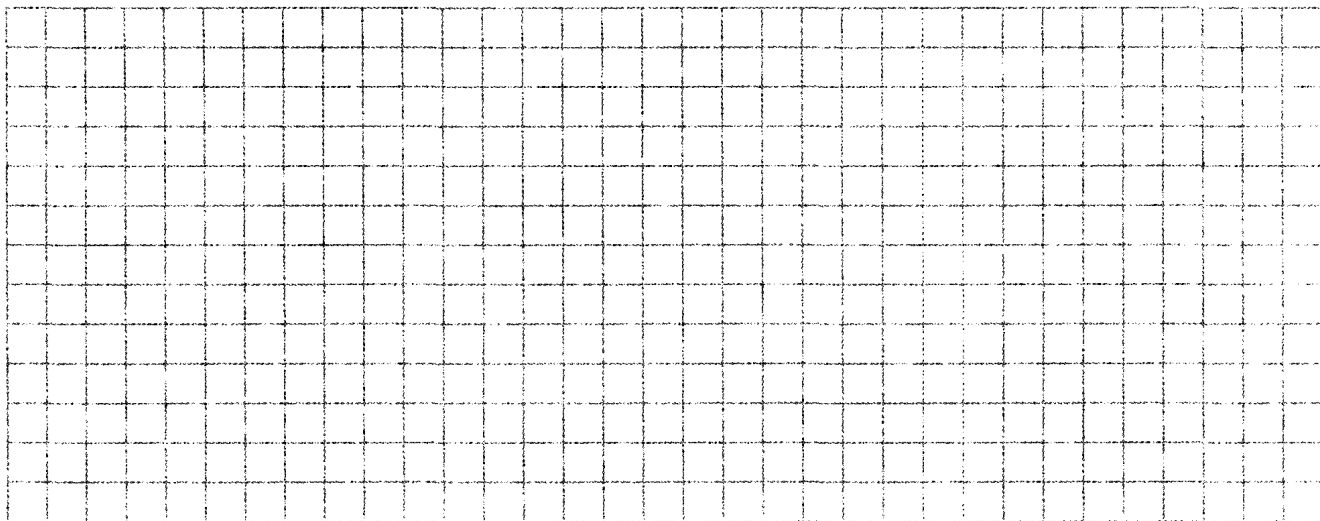
б) поиск количества минимальных элементов в заданном целочисленном массиве из 50 элементов;



в) поиск наибольшего общего делителя двух натуральных чисел;

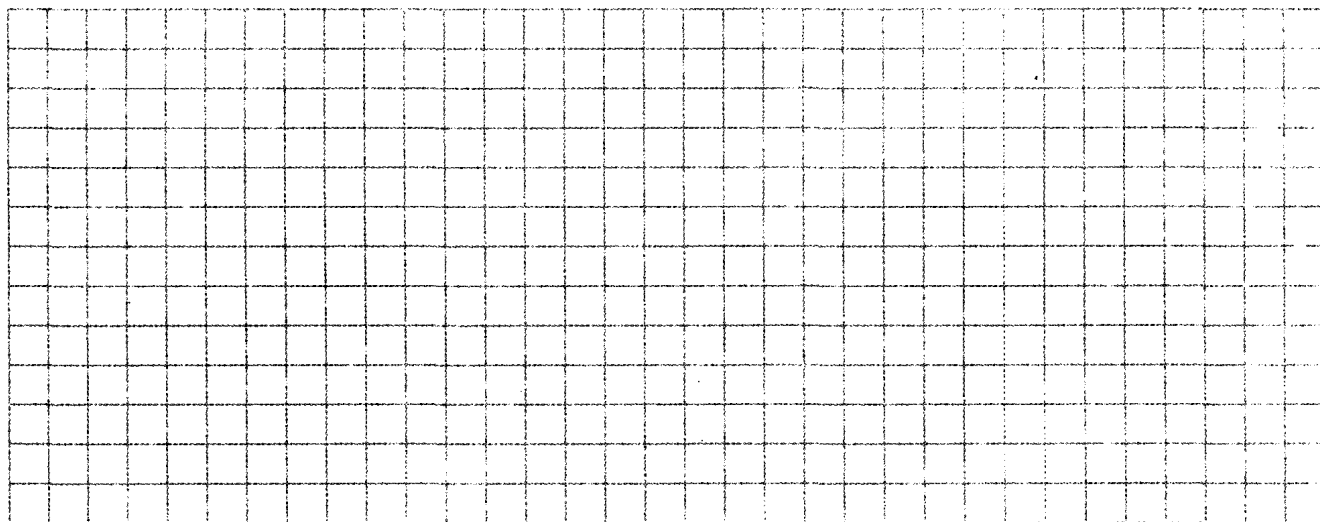


г) поиск количества простых чисел в заданном целочисленном массиве из 50 элементов.

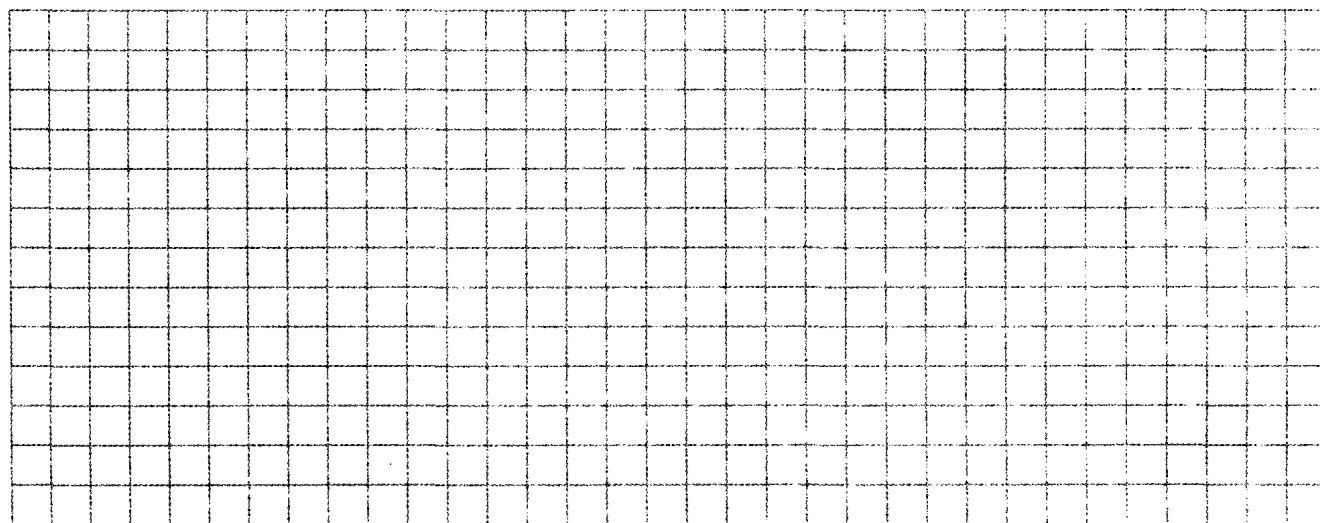


**5.7.** Составить и записать в виде блок-схемы алгоритм, выполняющий:

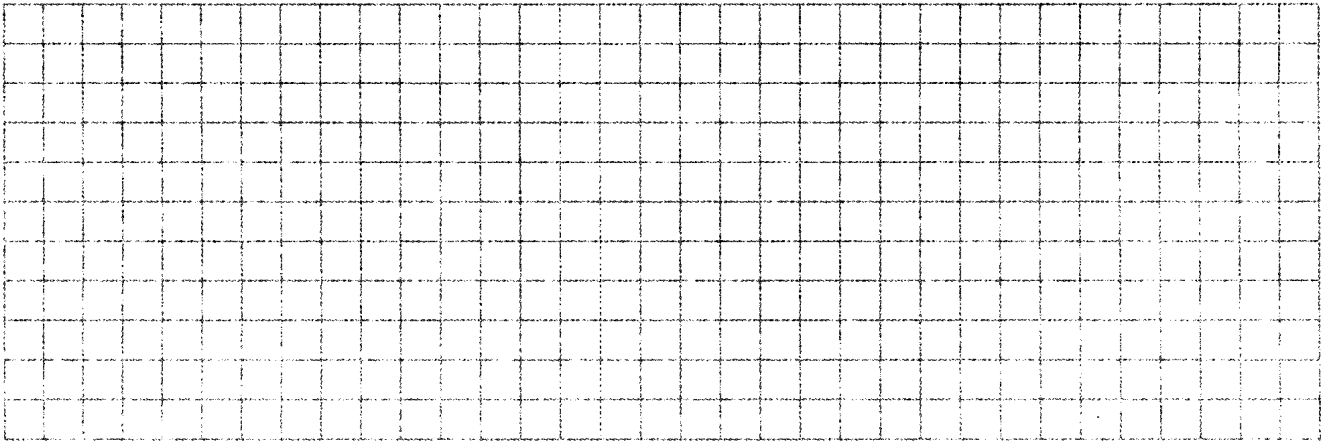
а) решение уравнения  $ax = b$ , где  $a$  и  $b$  — числовые исходные данные;



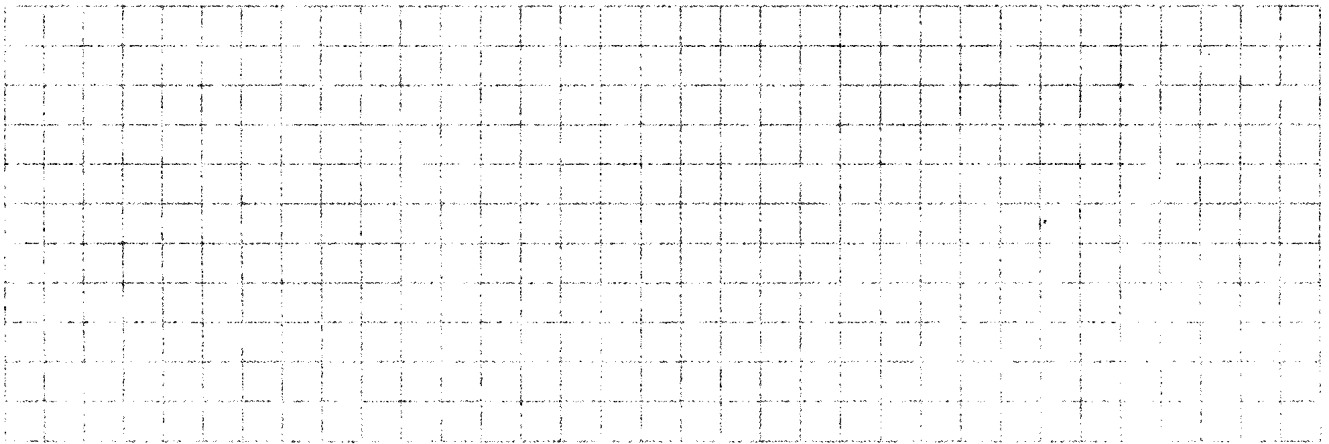
б) поиск количества минимальных элементов в заданном целочисленном массиве из 50 элементов;



в) поиск наибольшего общего делителя двух натуральных чисел;



г) поиск количества простых чисел в заданном целочисленном массиве из 50 элементов.



В задачах на определение результатов выполнения программы для одномерных и двумерных массивов необходимо определить закономерность обработки массива, поскольку его размерность может быть задана переменной величиной, а не конкретным значением.

Однако для того, чтобы понять эту закономерность, иногда требуется выполнить несколько итераций цикла «вручную», используя табличное представление массива.

*Пример:* Значения двумерного массива размера  $n \times n$  задаются с помощью вложенного оператора цикла в представленном фрагменте программы.

Бейсик	Паскаль	Алгоритмический
FOR i=1 TO n FOR k=1 TO n IF i=k THEN A(i,k) = 2 ELSE A(i,k) = 0 END IF NEXT k NEXT n	for i:=1 to n do for k:=1 to n do if i=k then A[i,k]:=2 else A[i,k]:=0	<u>нц</u> для i от 1 до n <u>нц</u> для k от 1 до n <u>если</u> i=k то A[i,k]:=2 <u>иначе</u> A[i,k]:=0 <u>все</u> <u>кц</u> <u>кц</u>

Как будет зависеть от  $n$  сумма элементов массива  $A$  после выполнения алгоритма?

**Решение**

Изобразим двумерный массив в виде таблицы. На диагонали таблицы, там, где  $k = i$ , будут находиться двойки. В остальных ячейках — нули.

$k \backslash i$	1	2	3	...	$n-1$	$n$
1	2	0	0	0	0	0
2	0	2	0	0	0	0
3	0	0	2	0	0	0
...	0	0	0	2	0	0
$n-1$	0	0	0	0	2	0
$n$	0	0	0	0	0	2

Количество элементов диагонали —  $n$ , поэтому сумма элементов массива будет равна  $2n$ .

Ответ:  $2n$ .

**5.8.** Выполнить фрагмент программы и найти итоговые значения переменных:

а)

Бейсик	Паскаль	Алгоритмический
$a=1234$ $b=(a \text{ MOD } 1000) \cdot 10$ $a=a \backslash 1000 + b$ \ и MOD — операции, вычисляющие результат деления нацело первого аргумента на второй и остаток от деления соответственно	$a:=1234;$ $b:=(a \text{ mod } 1000) \cdot 10;$ $a:=a \text{ div } 1000 + b;$ {div и mod — операции, вычисляющие результат деления нацело первого аргумента на второй и остаток от деления соответственно}	$a:=1234$ $b:=\text{mod}(a, 1000) \cdot 10$ $a:=\text{div}(a, 1000) + b$ div и mod — функции, вычисляющие результат деления нацело первого аргумента на второй и остаток от деления соответственно

$a =$  \_\_\_\_\_

$b =$  \_\_\_\_\_

б)

Бейсик	Паскаль	Алгоритмический
$a=1234$ $b=(a \text{ MOD } 10) \cdot 1000$ $a=a \backslash 10 + b$	$a:=1234;$ $b:=(a \text{ mod } 10) \cdot 1000;$ $a:=a \text{ div } 10 + b;$	$a:=1234$ $b:=\text{mod}(a, 10) \cdot 1000$ $a:=\text{div}(a, 10) + b$

$a =$  \_\_\_\_\_

$b =$  \_\_\_\_\_

в)

Бейсик	Паскаль	Алгоритмический
$a=(7-5) \cdot 4$ $b=(a \text{ MOD } 3) + 15$ $a=(b \backslash 4) + 3$	$a:=(7-5) \cdot 4;$ $b:=(a \text{ mod } 3) + 15;$ $a:=(b \text{ div } 4) + 3$	$a:=(7-5) \cdot 4$ $b:=\text{mod}(a, 3) + 15$ $a:=\text{div}(b, 4) + 3$

$a =$  \_\_\_\_\_

$b =$  \_\_\_\_\_

г)	Бейсик	Паскаль	Алгоритмический
	<pre>a=(6+2) . 4 b=(a MOD 5)+1 a=(b\6)-2</pre>	<pre>a:=(6+2) . 4; b:=(a mod 5)+1; a:=(b div 6)-2</pre>	<pre>a:=(6+2) . 4 b:=mod(a,5)+1 a:=div(b,6)-2</pre>

$a =$  \_\_\_\_\_

$b =$  \_\_\_\_\_

### 5.9.

а) Дан фрагмент программы, обрабатывающей массив  $A$  из 10 элементов:

Бейсик	Паскаль	Алгоритмический
<pre>n=10 FOR i = 1 TO n A(i)=i NEXT I j = 1 FOR i = 1 TO n-1 IF A(i) &lt;A(i+1) THEN j=j+1 NEXT I</pre>	<pre>n=10; for i:=1 to n do A[i]:=i; j:=1; for i:=1 to n-1 do     if A[i]&lt;A[i+1]         then j:=j+1;</pre>	<pre>n:=10 <u>НЦ</u> <u>ДЛЯ</u> i <u>ОТ</u> 1 <u>ДО</u> n A[i]= i <u>КЦ</u> j:=1 <u>НЦ</u> <u>ДЛЯ</u> i <u>ОТ</u> 1 <u>ДО</u> n-1     <u>ЕСЛИ</u> A[i]&lt;A[i+1] <u>ТО</u>         j=j+1     <u>ВСЕ</u> <u>КЦ</u></pre>

Чему будет равно значение переменной  $j$  после выполнения данного алгоритма?

Ответ: \_\_\_\_\_

б) Дан фрагмент программы, обрабатывающей массив  $A$  из 10 элементов:

Бейсик	Паскаль	Алгоритмический
<pre>n=10 FOR i = 1 TO n A(i)=11-i NEXT I s = 0 FOR i = 1 TO n IF i&gt;5 THEN s=s+A(i) NEXT I</pre>	<pre>n=10; for i:=1 to n do     A[i]:=11-i; s:=0; for i:=1 to n do     if i&gt;5 then s:=s+A[i];</pre>	<pre>n:=10 <u>НЦ</u> <u>ДЛЯ</u> i <u>ОТ</u> 1 <u>ДО</u> n A[i]= 11-i <u>КЦ</u> s:=0 <u>НЦ</u> <u>ДЛЯ</u> i <u>ОТ</u> 1 <u>ДО</u> n     <u>ЕСЛИ</u> i&gt;5 <u>ТО</u>         s=s+A[i]     <u>ВСЕ</u> <u>КЦ</u></pre>

Чему будет равно значение переменной  $s$  после выполнения данного алгоритма?

Ответ: \_\_\_\_\_

в) Дан фрагмент программы, обрабатывающей массив  $A$  из  $n$  элементов:

Бейсик	Паскаль	Алгоритмический
<pre> j = A(1) FOR i = 2 TO n IF j &lt; A(i) THEN j=A(i) NEXT I </pre>	<pre> j:=A[1]; for i:=2 to n do   if j&lt;A[i]     then j:=A[i]; </pre>	<pre> j:=A[1] нц для i от 2 до n   если j&lt;A[i] то     j:=A[i]   все кц </pre>

Как зависит итоговое значение переменной  $j$  от значений массива  $A$ ?

Ответ: \_\_\_\_\_.

г) Дан фрагмент программы, обрабатывающей массив  $A$  из  $n$  элементов:

Бейсик	Паскаль	Алгоритмический
<pre> j = 0 FOR i = 1 TO n IF A(i)&lt;0 THEN j=j+1 NEXT I </pre>	<pre> j:=0; for i:=1 to n do   if A[i]&lt;0     then j:=j+1; </pre>	<pre> j:=0 нц для i от 1 до n   если A[i]&lt;0 то     j:=j+1   все кц </pre>

Как зависит итоговое значение переменной  $j$  от значений массива  $A$ ?

Ответ: \_\_\_\_\_.

## 5.10.

а) Значения двумерного массива размера  $n \times n$  задаются с помощью вложенного оператора цикла в представленном фрагменте программы:

Бейсик	Паскаль	Алгоритмический
<pre> FOR i=1 TO n FOR k=1 TO n IF i&gt;k THEN   A(i,k) = 1 ELSE   A(i,k) = 0 END IF NEXT k NEXT n </pre>	<pre> for i:=1 to n do for k:=1 to n do if i&gt;k then   A[i,k]:=1 else   A[i,k]:=0 </pre>	<pre> нц для i от 1 до n нц для k от 1 до n   если i&gt;k то     A[i,k]:=1   иначе     A[i,k]:=0   все кц кц </pre>

Как будет зависеть от  $n$  сумма элементов массива  $A$  после выполнения алгоритма? Напишите формулу вычисления суммы элементов массива  $A$ , в зависимости от  $n$ .

Ответ: \_\_\_\_\_.

- б) Значения двумерного массива размера  $n \times n$  задаются с помощью вложенного оператора цикла в представленном фрагменте программы:

Бейсик	Паскаль	Алгоритмический
<pre>FOR i=1 TO n FOR k=1 TO n IF i=k THEN   A(i,k) = 1 ELSE   A(i,k) = -1 END IF NEXT k NEXT n</pre>	<pre>for i:=1 to n do for k:=1 to n do if i=k then   A[i,k]:=1 else   A[i,k]:=-1</pre>	<pre><u>нц</u> для i от 1 до n <u>нц</u> для k от 1 до n   <u>если</u> i=k то     A[i,k]:=1   <u>иначе</u>     A[i,k]:=-1   <u>все</u> <u>кц</u> <u>кц</u></pre>

Как будет зависеть от  $n$  сумма элементов массива  $A$  после выполнения алгоритма? Напишите формулу вычисления суммы элементов массива  $A$ , в зависимости от  $n$ .

Ответ: \_\_\_\_\_.

- в) Значения двумерного массива размера  $n \times n$  задаются с помощью вложенного оператора цикла в представленном фрагменте программы:

Бейсик	Паскаль	Алгоритмический
<pre>FOR i=1 TO n FOR k=1 TO n IF i&gt;k THEN   A(i,k) = 1 ELSE   A(i,k) = -1 END IF NEXT k NEXT n</pre>	<pre>for i:=1 to n do for k:=1 to n do if i&gt;k then   A[i,k]:=1 else   A[i,k]:=-1</pre>	<pre><u>нц</u> для i от 1 до n <u>нц</u> для k от 1 до n   <u>если</u> i&gt;k то     A[i,k]:=1   <u>иначе</u>     A[i,k]:=-1   <u>все</u> <u>кц</u> <u>кц</u></pre>

Как будет зависеть от  $n$  сумма элементов массива  $A$  после выполнения алгоритма? Напишите формулу вычисления суммы элементов массива  $A$ , в зависимости от  $n$ .

Ответ: \_\_\_\_\_.

- г) Значения двумерного массива размера  $n \times n$  задаются с помощью вложенного оператора цикла в представленном фрагменте программы:

Бейсик	Паскаль	Алгоритмический
<pre>FOR i=1 TO n FOR k=1 TO n IF i&gt;k THEN   A(i,k) = i ELSE   A(i,k) = -k END IF NEXT k NEXT n</pre>	<pre>for i:=1 to n do for k:=1 to n do if i&gt;k then   A[i,k]:=i else   A[i,k]:=-k</pre>	<pre><u>нц</u> для i от 1 до n <u>нц</u> для k от 1 до n   <u>если</u> i&gt;k то     A[i,k]:=i   <u>иначе</u>     A[i,k]:=-k   <u>все</u> <u>кц</u> <u>кц</u></pre>

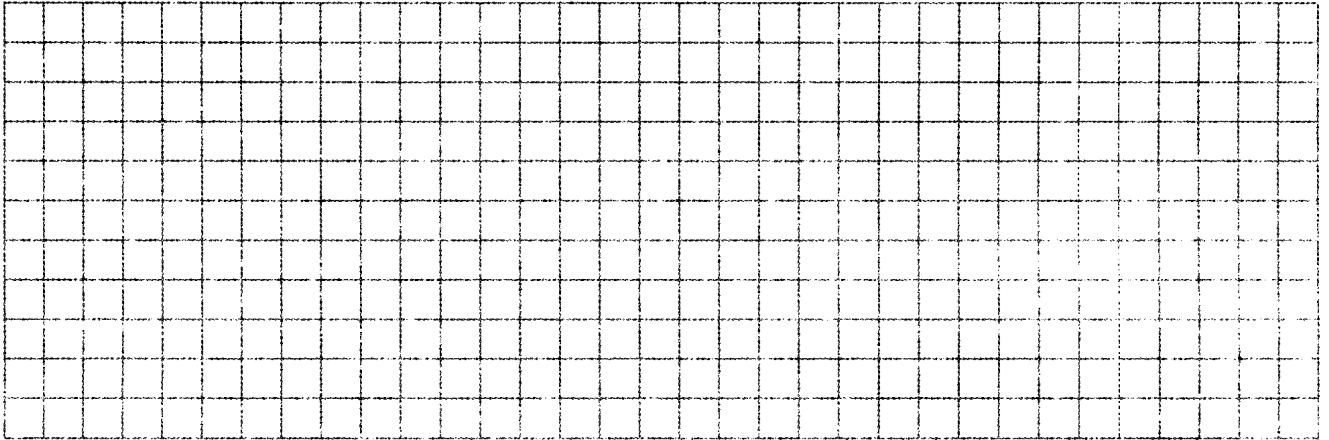
Как будет зависеть от  $n$  сумма элементов массива  $A$  после выполнения алгоритма? Напишите формулу вычисления суммы элементов массива  $A$ , в зависимости от  $n$ .

Ответ: \_\_\_\_\_.

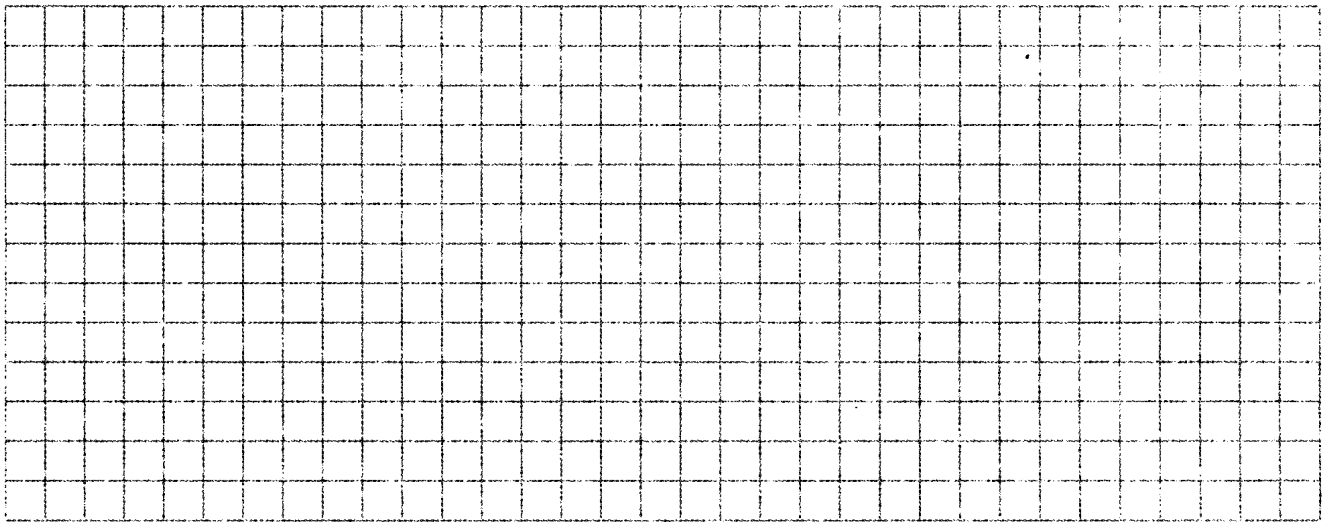


**5.11.** Составить и записать в виде программы на любом языке программирования алгоритм, выполняющий:

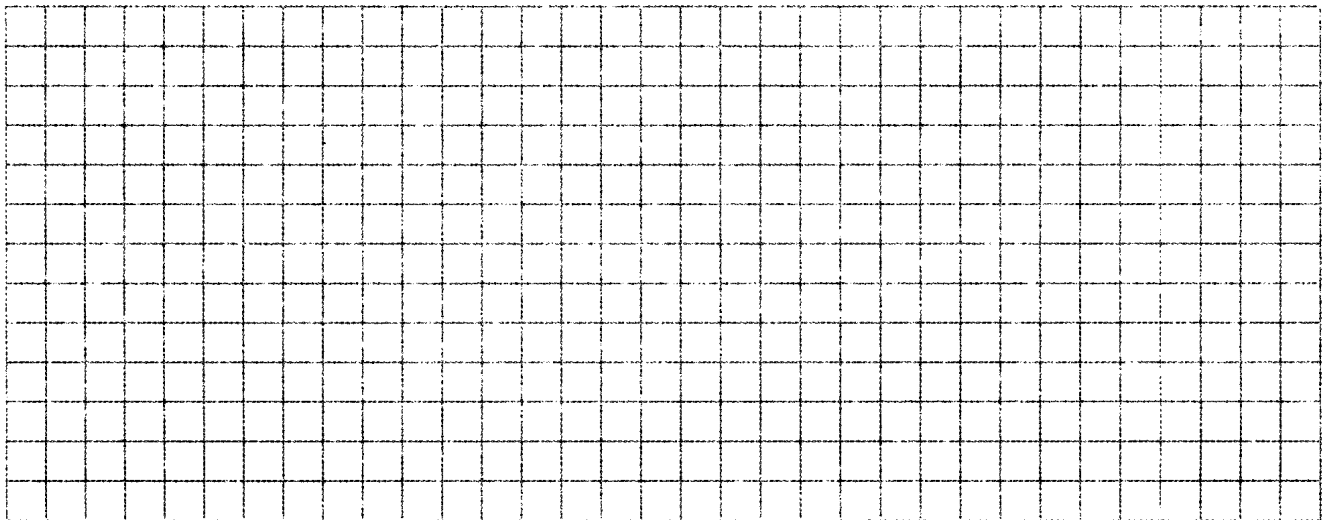
а) Решение уравнения  $a|x| = b$ , где  $a$  и  $b$  — исходные числовые данные;



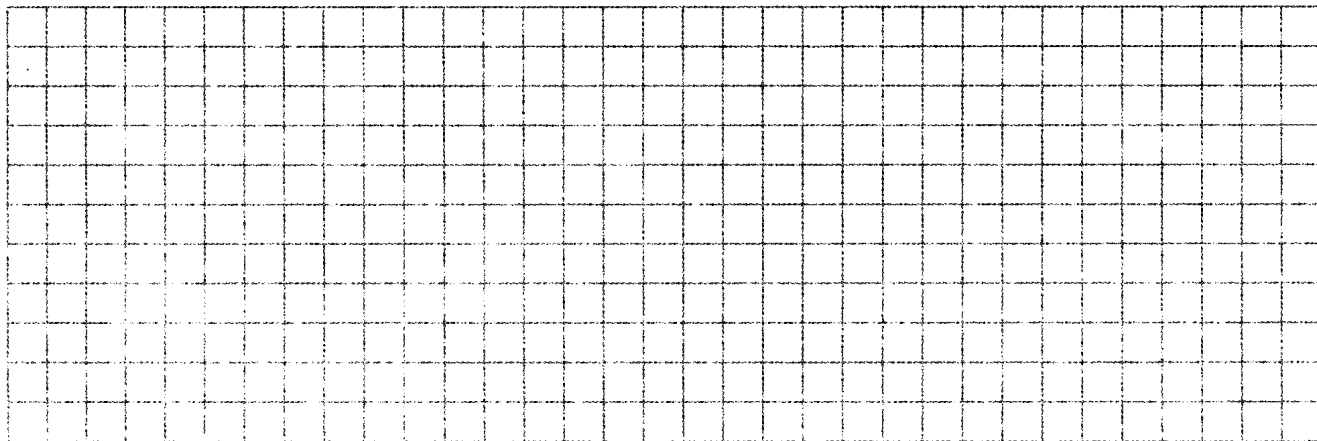
б) Поиск количества минимальных элементов в заданном целочисленном массиве из 50 элементов;



в) Поиск наибольшего общего делителя двух натуральных чисел;



г) Поиск количества простых чисел в заданном целочисленном массиве из 50 элементов;



### Важные рекомендации

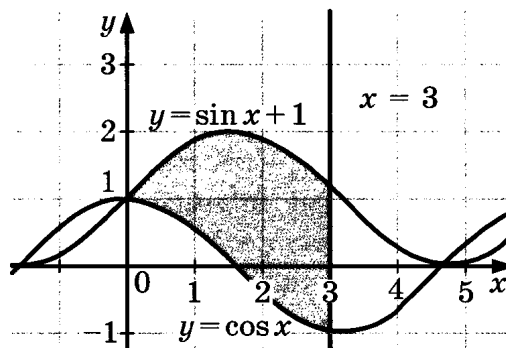
Практика показывает, что наибольшие затруднения при выполнении заданий по теме «Алгоритмизация и программирование» вызывает составление программ и алгоритмов (задачи С1, С2 и С4 демоверсии ЕГЭ 2009 г.). Данное пособие не является учебником программирования, поэтому мы приведем лишь общие рекомендации, не зависящие от языка, на котором записываются программы:

- Перед составлением программы надо тщательно проанализировать условие задания, уяснить постановку задачи, формат входных и выходных данных.
- Приступать к написанию программы можно лишь тогда, когда детально продуман алгоритм решения задачи, не следует начинать писать программу, не представляя алгоритмического решения. Для простой задачи алгоритмическое решение можно держать и в голове, но лучше его зафиксировать на бумаге в виде пояснения или схемы.
- Для написания программы нужно пользоваться тем языком программирования, которым хорошо владеешь и который подходит для решения поставленной задачи. Не следует специально стремиться поразить экзаменаторов знанием экзотического или недавно появившегося языка программирования, поскольку дополнительные баллы за это не предусмотрены. Хорошая программа должна быть максимально простой и эффективной. Не нужно специально использовать сложные приемы или редко применяемые конструкции языка программирования, если в этом нет необходимости.
- Текст программы нужно писать аккуратно и разборчиво даже на черновике, структурировано, выделяя отступами уровень вложенности операторов, не скупясь на комментарии и пояснения. Отсутствие комментариев и пояснений вовсе не является признаком высокой квалификации программиста, скорее наоборот.
- При написании текста программы следует контролировать соответствие типов переменных и применяемых к ним операций, не допускать использования неопределенных (неинициализированных) значений переменных.
- Если текст программы получается, на ваш взгляд, очень сложным и запутанным, приостановите на время ее написание и подумайте, в чем причина этой сложности, нельзя ли найти более простое и изящное решение задачи.
- После того как программа написана, необходимо выполнить ее тестирование, то есть проверку правильности работы для различных исходных данных. Постарайтесь найти ошибку в своей программе, помните, что цель тестирования — поиск ошибок, а не демонстрация правильности программы. Гораздо лучше, если ошибку найдете и исправите вы сами, а не экзаменаторы при проверке.

- При тестировании следует подбирать набор тестовых данных так, чтобы все операторы, все «ветки» программы выполнились на совокупности тестов. Особое внимание следует уделить особым случаям исходных данных, пограничным, критическим точкам. Пусть, например, написана программа сортировки массива. Особыми здесь будут случаи, когда входной массив уже отсортирован в нужном порядке, когда он отсортирован в порядке обратном требуемому, а также когда все элементы сортируемого массива равны. Следует выявить такие критические точки и убедиться, что программа правильно работает на этих тестах.
- Если в задании приведен пример входных данных, его тоже надо включить в состав тестов, но он не должен быть единственным тестом!
- При тестировании следует еще раз сверить структуру исходных и выходных данных с заданием, убедиться, что составленная программа точно соответствует условию задачи.
- При тестировании необходимо внимательно контролировать синтаксис программы (должен быть явно указан тип каждой переменной, если этого требует язык программирования, переменные должны использоваться в соответствии со своим типом, не должно быть неинициализированных и неиспользуемых переменных, должны быть правильно расставлены операторные скобки и разделители операторов и т.д.)
- Если после тестирования программы осталось достаточно времени, то можно поработать над повышением ее эффективности, подумать, нельзя ли упростить ее, уменьшить количество циклов или требуемой для ее работы памяти. В случае внесения изменений в готовую программу процедуру тестирования придется повторить заново.

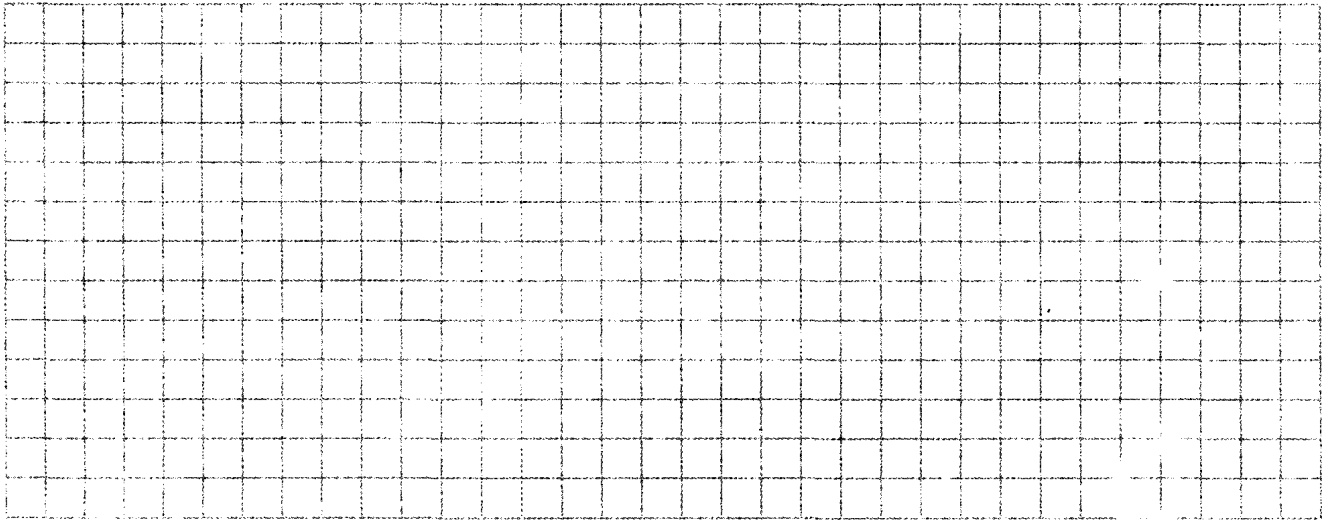
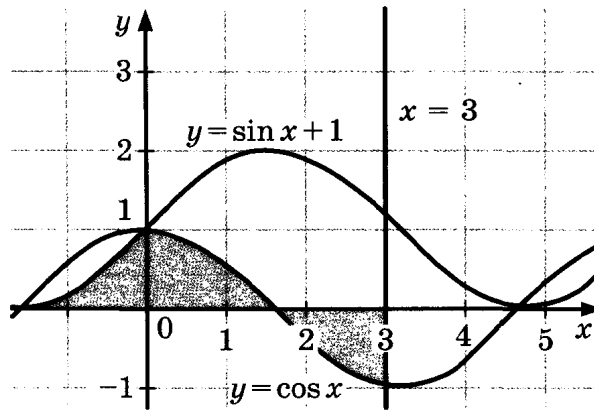
## 5.12.

- a) Найдите и исправьте ошибки в приведенной программе проверки принадлежности точки закрашенной области. Приведите пример, когда исходная программа работала неправильно.

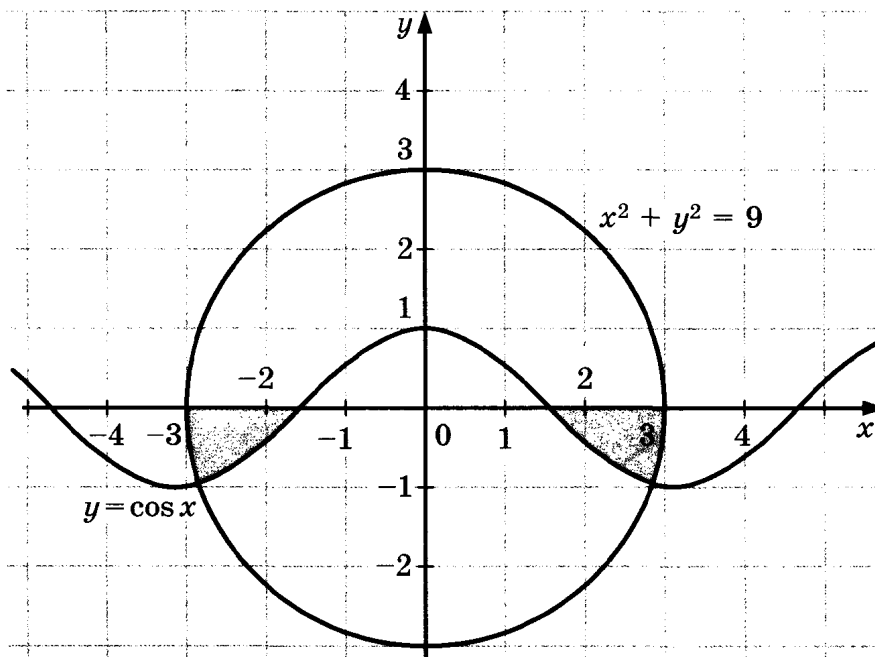


Программа на Паскале	Программа на Бейсике	Программа на Си
<pre>var x,y: real; begin   readln(x,y);   if y&lt;=sin (x)+1 then     if y&gt;= cos (x) then       if x&lt;=3 then         write('принадлежит')       else         write('не принадлежит')     end. end.</pre>	<pre>INPUT x, y IF y&lt;=sin(x)+1 THEN   IF y&gt;= cos(x) THEN     IF x&lt;=3 THEN       PRINT "принадлежит"     ELSE       PRINT "не принадлежит"     ENDIF   ENDIF ENDIF END</pre>	<pre>void main(void) { float x,y;   scanf("%f%f",&amp;x,&amp;y);   if (y&lt;=sin(x)+1)     if (y&gt;=cos(x))       if (x&lt;=3)         printf("принадлежит");       else         printf("не принадлежит");     } }</pre>

- б) Составьте программу проверки принадлежности точки закрашенной области.

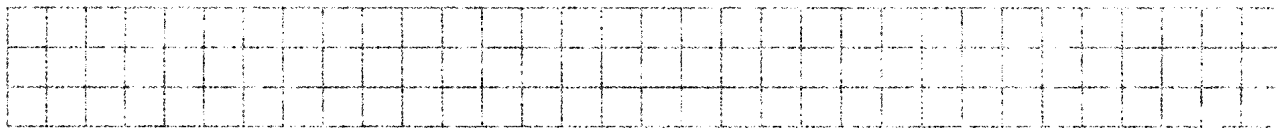
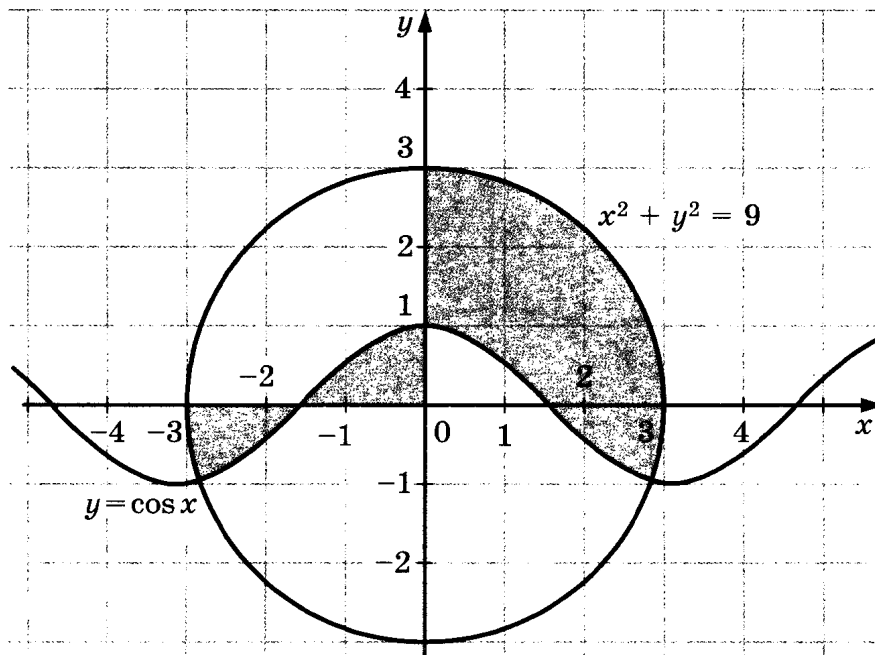


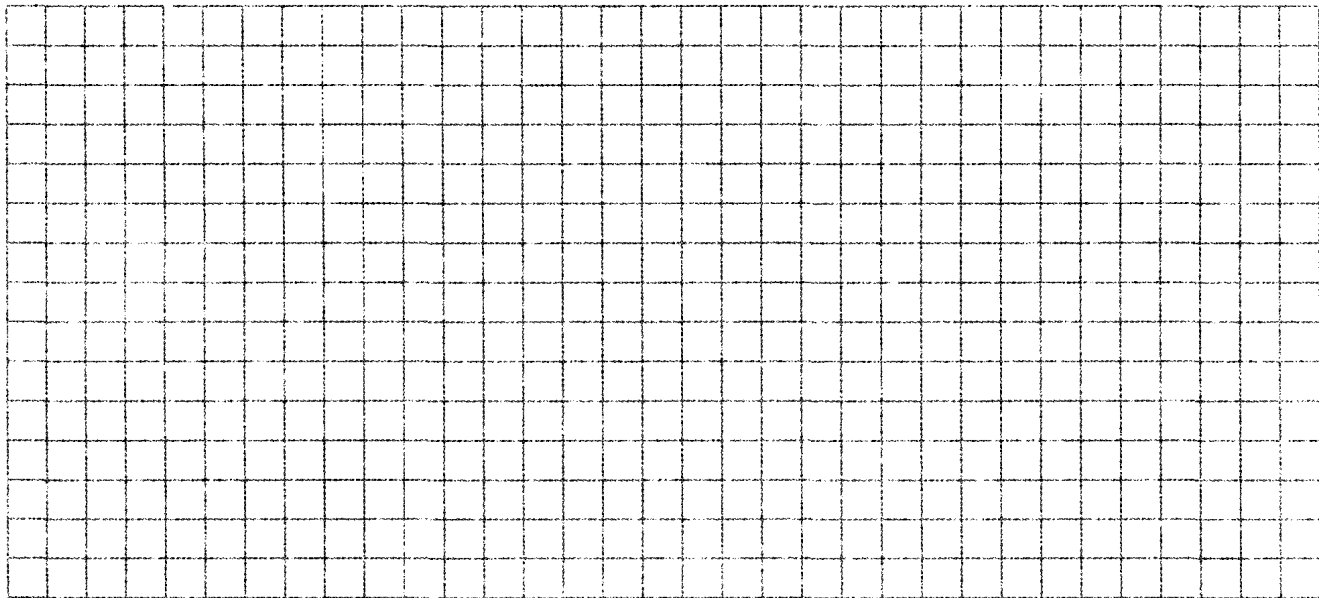
- в) Найдите и исправьте ошибки в приведенной программе проверки принадлежности точки закрашенной области. Приведите пример, когда исходная программа работала неправильно.



<b>Программа на Паскале</b>	<pre> var x,y: real; begin   readln(x,y);   if (sqr(x) + sqr(y) &lt;=3 ) or (y&gt;=cos(x)) or (y&lt;=0))then     write('принадлежит')   else     write('не принадлежит') end. </pre>
<b>Программа на Бейсике</b>	<pre> INPUT x, y IF (x * x+ y * y &lt;= 3) or (y&gt;=cos(x) )or (y&lt;=0) THEN   PRINT "принадлежит" ELSE   PRINT "не принадлежит" ENDIF END </pre>
<b>Программа на Си</b>	<pre> void main(void) { float x,y;   scanf("%f%f",&amp;x,&amp;y);   if ((x * x+ y * y &lt;= 3)    (y&gt;=cos(x) )   (y&lt;=0))     printf("принадлежит");   else     printf("не принадлежит"); } </pre>

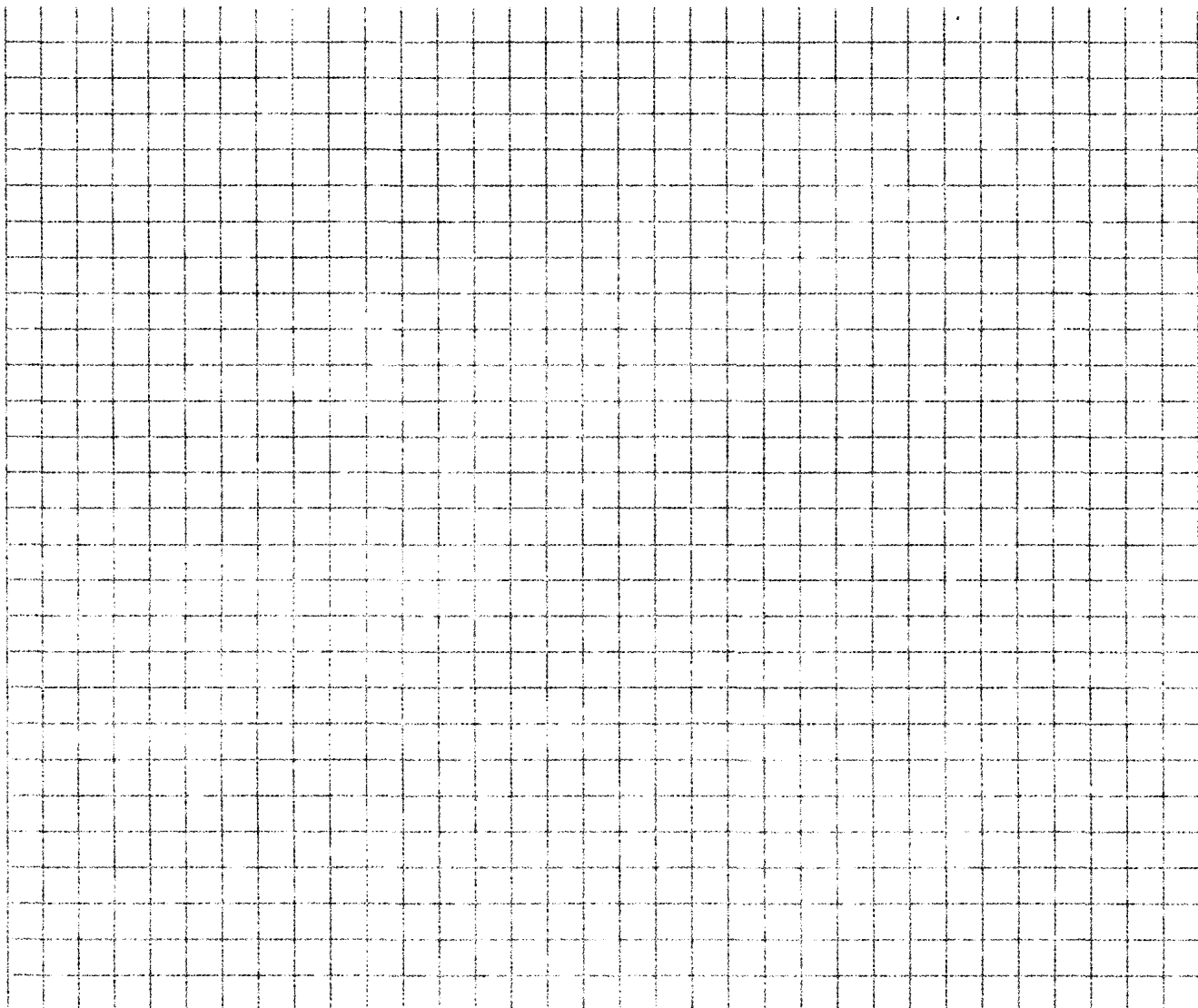
Составьте программу проверки принадлежности точки закрашенной области.



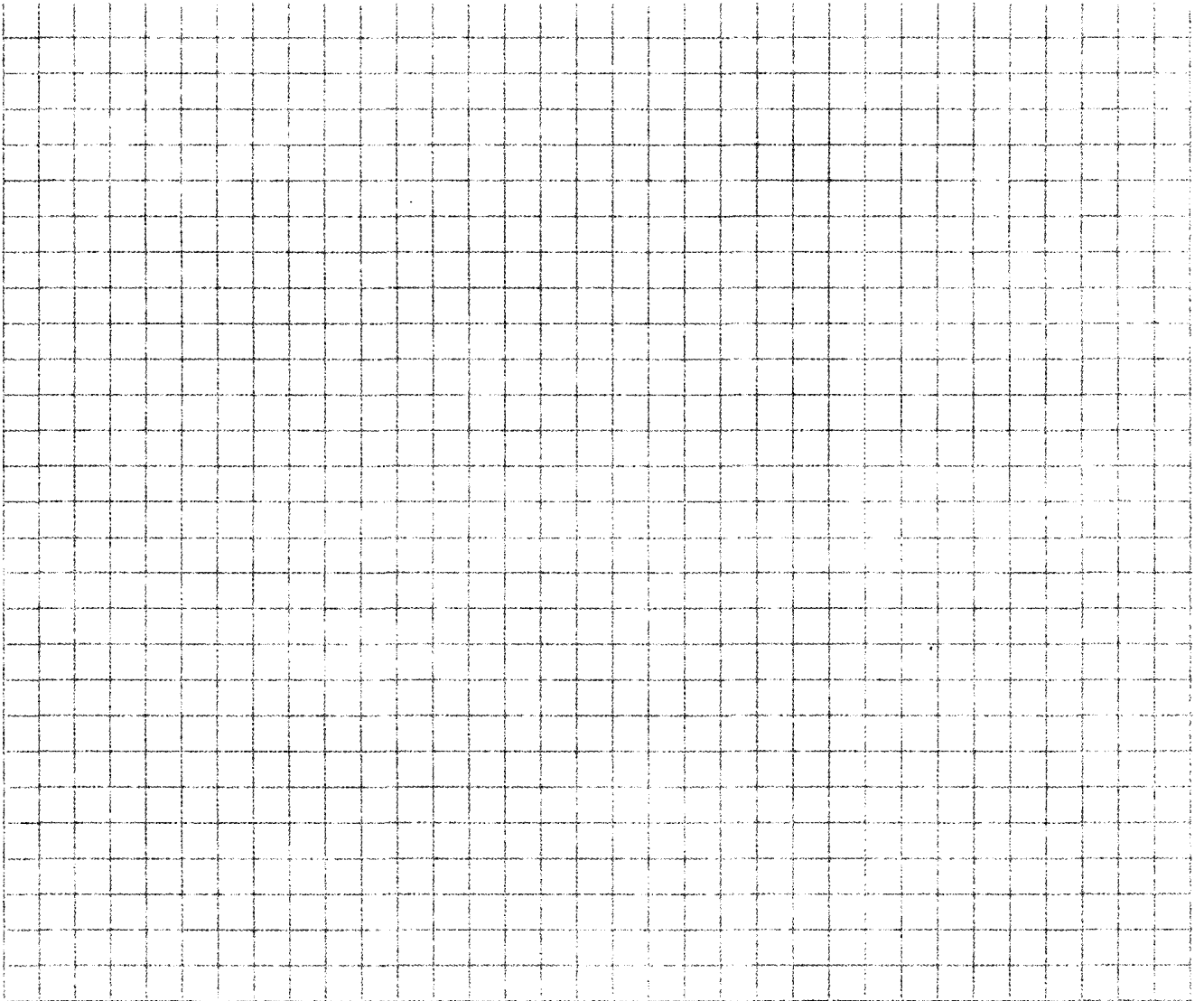


**5.13.**

- а) Составить программу, выводящую на экран все натуральные трехзначные числа, делящиеся без остатка на 17, сумма цифр которых равняется 11.

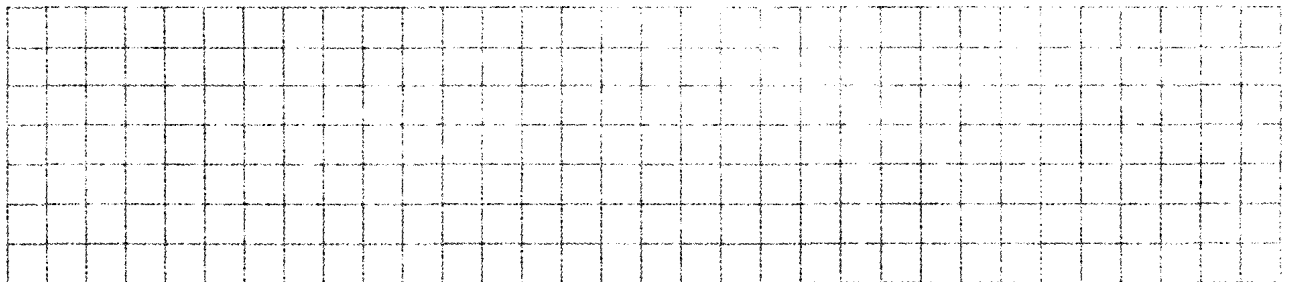


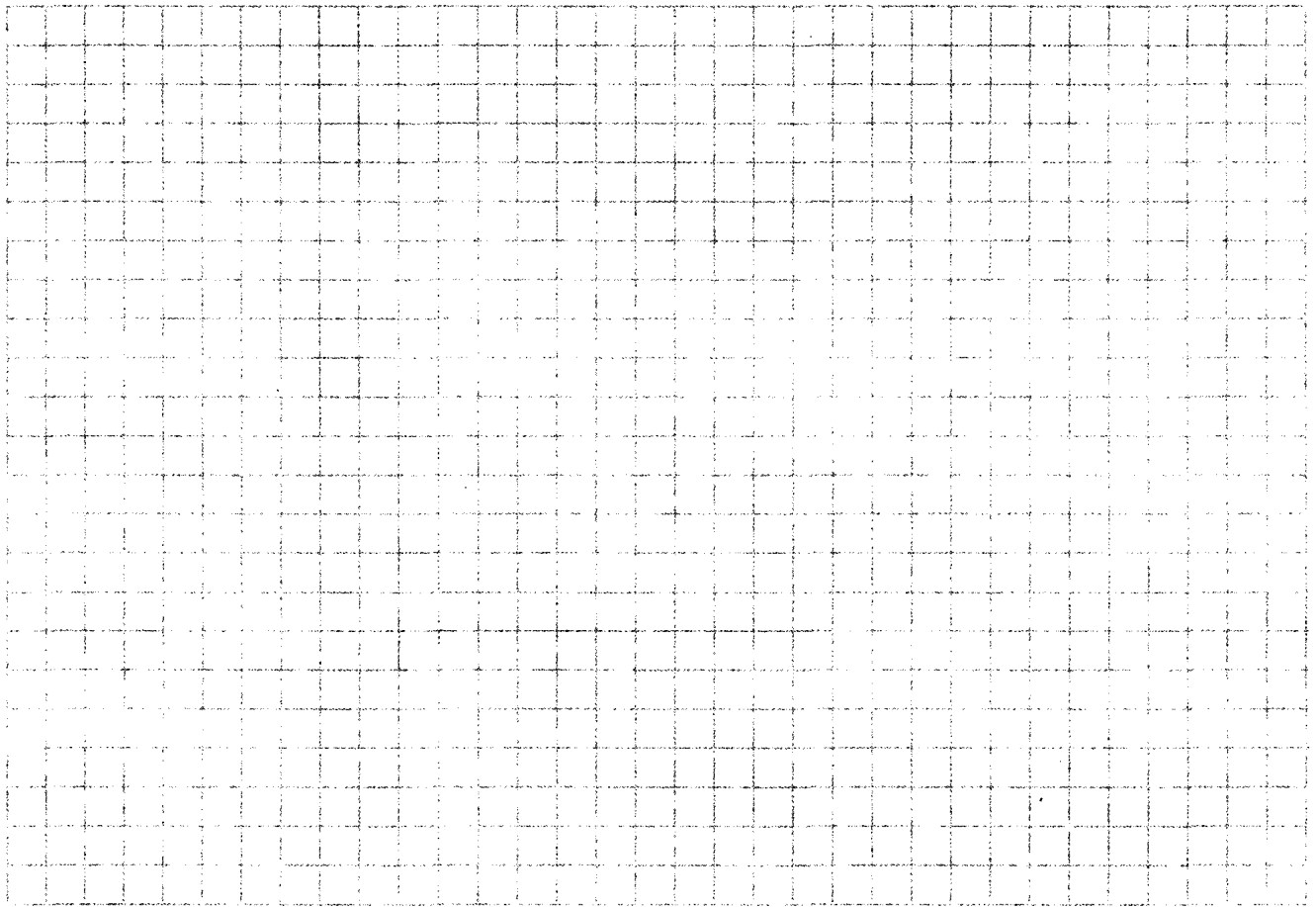
- б) Составить программу, выводящую на экран все натуральные трехзначные числа, делящиеся на 19 с остатком 3, сумма цифр которых равняется 12.



- в) На продовольственном складе имеется 10 000 наименований различных продуктов, информация о которых хранится в текстовом файле. Каждому продукту отведена 1 строка файла. В ней указан числовой артикул продукта, название (не содержащее пробелов), категория продукта (молочный, мясной или рыбный) и его стоимость. Вышеперечисленные поля разделены одним пробелом.

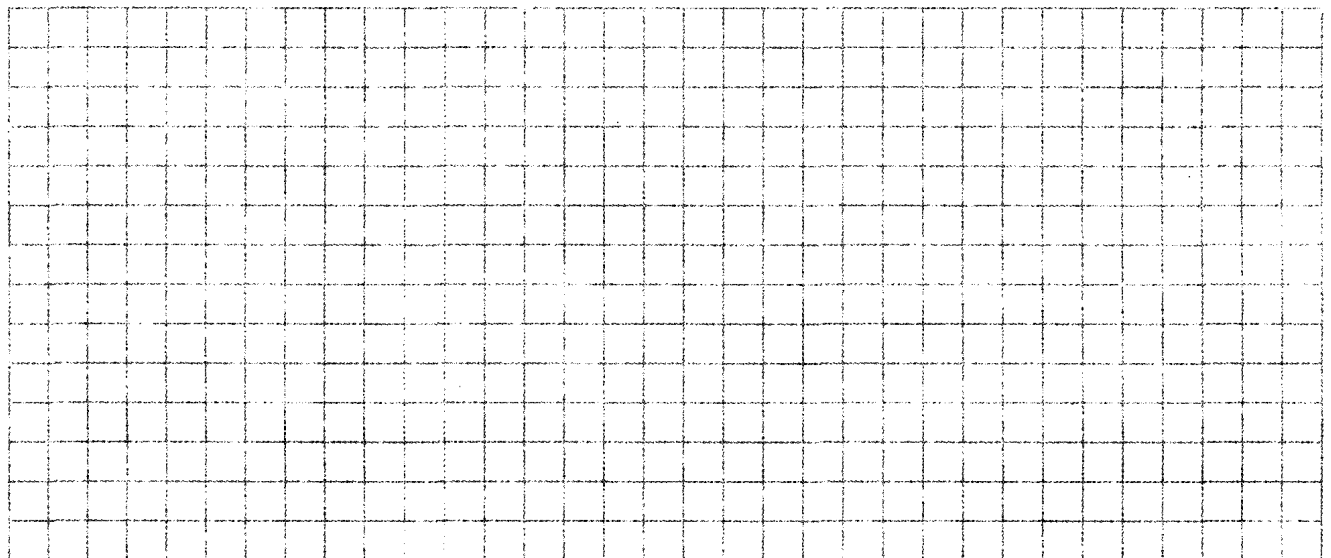
Написать программу, загружающую в память компьютера информацию о продуктах из текстового файла и выдающую на экран монитора наименования и артикулы всех рыбных продуктов, стоимость которых отличается от максимальной по данной категории не более чем на 10%.



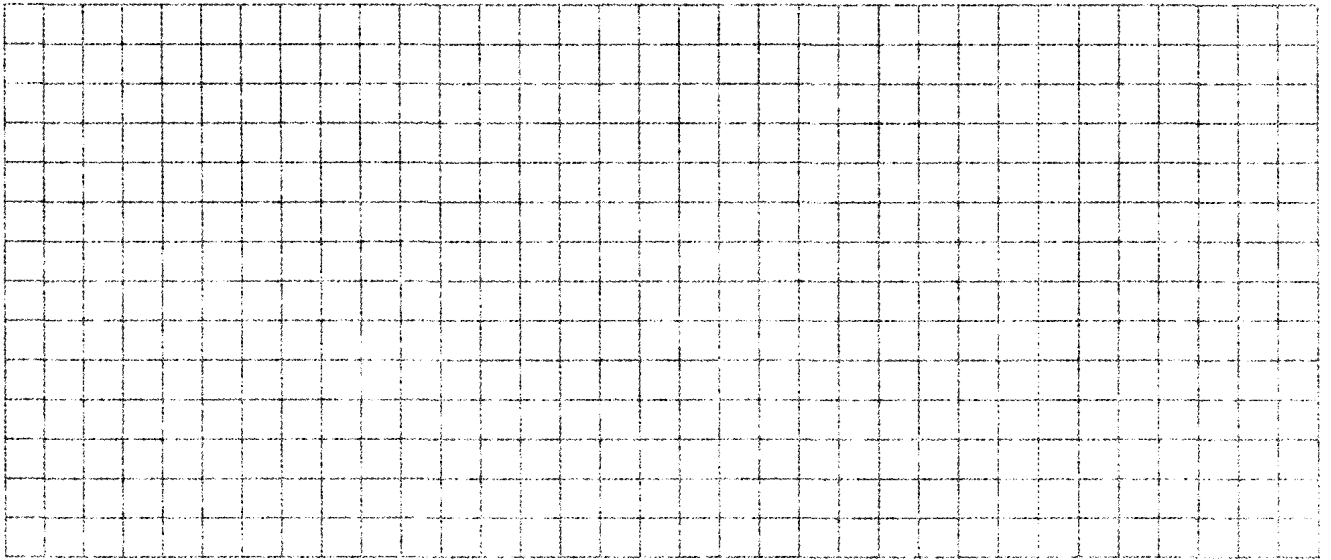


На предприятии работает 1000 сотрудников, информация о которых хранится в текстовом файле. Каждому сотруднику отведена 1 строка файла. В ней указаны целочисленный индивидуальный табельный номер, фамилия сотрудника (не содержащая пробелов), стаж (целое число лет) и зарплата. Вышеперечисленные поля разделены одним пробелом.

Написать программу, загружающую в память компьютера информацию о сотрудниках из текстового файла и выдающую на экран фамилии и табельные номера всех сотрудников, проработавших не менее 8 лет, зарплата которых меньше средней зарплаты по предприятию более чем на 25%.







### Задания в формате ЕГЭ

#### 5.14.

- а) Для составления цепочек используются бусины, помеченные буквами I, M, N, O, P. Цепочка формируется по следующему правилу: в середине цепочки стоит одна из бусин I, M, O. На третьем месте — бусина, помеченная любой гласной, если первая бусина — согласной, или любой согласной, если первая — гласной. На первом месте — одна из бусин I, O, P, не стоящая в цепочке посередине.

Какая из перечисленных цепочек создана по этому правилу?

- 1) OPN                      2) P I I                      3) POM                      4) I I O

- б) Для составления цепочек используются бусины, помеченные буквами A, M, N, O, P. Цепочка формируется по следующему правилу: в середине цепочки стоит одна из бусин M, O. На третьем месте — бусина, помеченная любой гласной, если первая бусина — согласной, или любой согласной, если первая — гласной. На первом месте — одна из бусин N, O, P, не стоящая в цепочке посередине.

Какая из перечисленных цепочек создана по этому правилу?

- 1) NMA                      2) NOM                      3) OON                      4) NAO

- в) Для составления цепочек используются бусины, помеченные буквами A, B, G, E, O. Цепочка формируется по следующему правилу: бусины с гласной и согласной буквой не могут стоять рядом. В середине цепочки может стоять одна из бусин B или E, не стоящая на первом месте. На третьем месте — бусина, помеченная любой гласной.

Какая из перечисленных цепочек создана по этому правилу?

- 1) AGE                      2) AEA                      3) GBA                      4) ABG

- г) Дешифровщику необходимо восстановить поврежденный фрагмент сообщения, состоящий из 4 символов. Имеется достоверная информация, что использовано не более пяти букв (A, B, E, B, G), причем на втором месте стоит один из символов A, G, E, B. На первом — одна из букв B, B, G, E, которой нет на третьем месте. На третьем мес-





Запишите порядок команд в программе получения из числа 5 числа 34 для данного исполнителя, содержащей не более 5 команд, указывая лишь номера команд.

Например, программа 21211 — это программа:

умножь на 3

вычти 1

умножь на 3

вычти 1

вычти 1

которая преобразует число 1 в 4.

Ответ: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

б) Запишите порядок команд в программе получения из числа 4 числа 31 для исполнителя Утроитель, содержащей не более 5 команд, указывая лишь номера команд.

Ответ: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

в) У исполнителя Квадратор две команды, которым присвоены номера:

1. прибавь 1

2. возведи в квадрат

Первая из них увеличивает число на экране на 1, вторая — возводит его в квадрат.

Запишите порядок команд в программе получения из числа 6 числа 66 для исполнителя Квадратор, содержащей не более 5 команд, указывая лишь номера команд.

Ответ: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

г) Запишите порядок команд в программе получения из числа 2 числа 50 для исполнителя Квадратор, содержащей не более 6 команд, указывая лишь номера команд.

Ответ: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

**5.18.** Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости:

вверх	вниз	влево	вправо
-------	------	-------	--------

При выполнении этой команды РОБОТ перемещается на соответствующую клетку. Команды проверки истинности условия на наличие стены у той клетки, где он находится:

сверху свободно	снизу свободно	слева свободно	справа свободно
-----------------	----------------	----------------	-----------------

Если РОБОТ начнет движение в сторону стены, то он разрушится.

- а) Сколько клеток данного лабиринта соответствуют требованию, что, выполнив предложенную программу, РОБОТ остановится в той же клетке, с которой он начал движение?

**НАЧАЛО**

**ПОКА справа свободно**

**ДЕЛАТЬ вправо**

**ПОКА снизу свободно**

**ДЕЛАТЬ вниз**

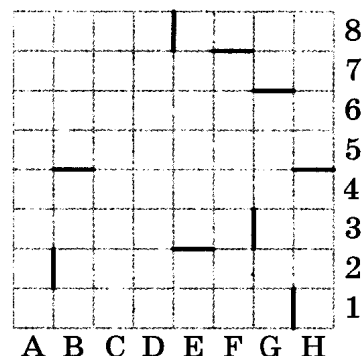
**ПОКА слева свободно**

**ДЕЛАТЬ влево**

**ПОКА сверху свободно**

**ДЕЛАТЬ вверх**

**КОНЕЦ**



В ответе запишите число — количество таких клеток, а далее, через запятые, их адреса (сначала идет латинская буква столбца, а затем цифра строки).

Например, нижний левый угол лабиринта имеет адрес A1.

Ответ: \_\_\_\_\_.

- б) Сколько клеток данного лабиринта соответствуют требованию, что, выполнив предложенную программу, РОБОТ остановится в той же клетке, с которой он начал движение?

**НАЧАЛО**

**ПОКА сверху свободно**

**ДЕЛАТЬ вверх**

**ПОКА слева свободно**

**ДЕЛАТЬ влево**

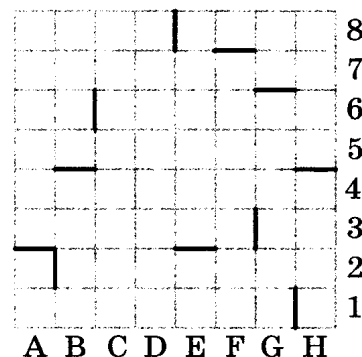
**ПОКА снизу свободно**

**ДЕЛАТЬ вниз**

**ПОКА справа свободно**

**ДЕЛАТЬ вправо**

**КОНЕЦ**



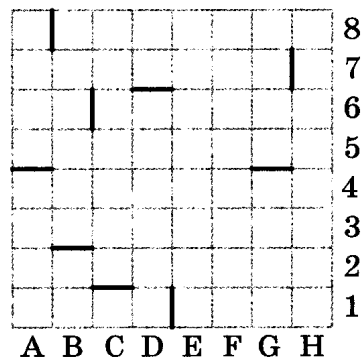
В ответе запишите число — количество таких клеток, а далее, через запятые, их адреса (сначала идет латинская буква столбца, а затем цифра строки).

Например, нижний левый угол лабиринта имеет адрес A1.

Ответ: \_\_\_\_\_.

- в) Сколько клеток данного лабиринта соответствуют требованию, что выполнив предложенную программу, РОБОТ остановится в той же клетке, с которой он начал движение?

ПОКА сверху свободно  
 ДЕЛАТЬ вверх  
 ПОКА слева свободно  
 ДЕЛАТЬ влево  
 ПОКА снизу свободно  
 ДЕЛАТЬ вниз  
 ПОКА справа свободно  
 ДЕЛАТЬ вправо  
 КОНЕЦ



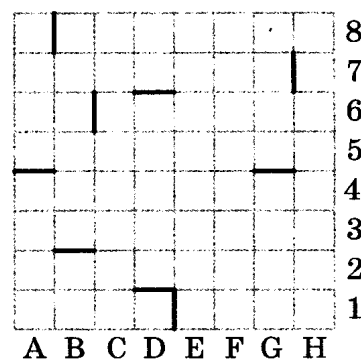
В ответе запишите число — количество таких клеток, а далее, через запятые, их адреса (сначала идет латинская буква столбца, а затем цифра строки).

Например, нижний левый угол лабиринта имеет адрес A1.

Ответ: \_\_\_\_\_.

- г) Сколько клеток данного лабиринта соответствуют требованию, что выполнив предложенную программу, РОБОТ остановится в той же клетке, с которой он начал движение?

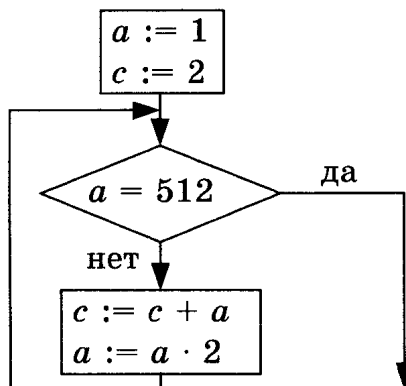
НАЧАЛО  
 ПОКА справа свободно  
 ДЕЛАТЬ вправо  
 ПОКА сверху свободно  
 ДЕЛАТЬ вверх  
 ПОКА слева свободно  
 ДЕЛАТЬ влево  
 ПОКА снизу свободно  
 ДЕЛАТЬ вниз  
 КОНЕЦ



Ответ: \_\_\_\_\_.

**5.19.** Определите значение переменной  $c$  после выполнения фрагмента алгоритма:

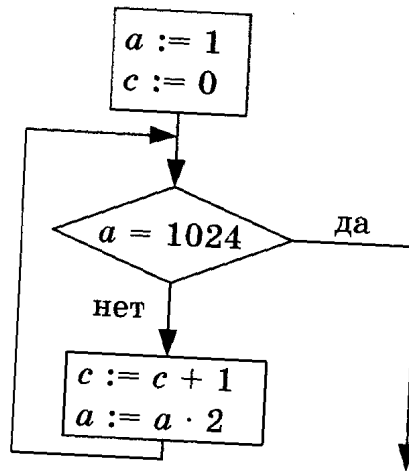
- а)



Примечание: знаком := обозначена операция присваивания.  
 знаком \* обозначена операция умножения.

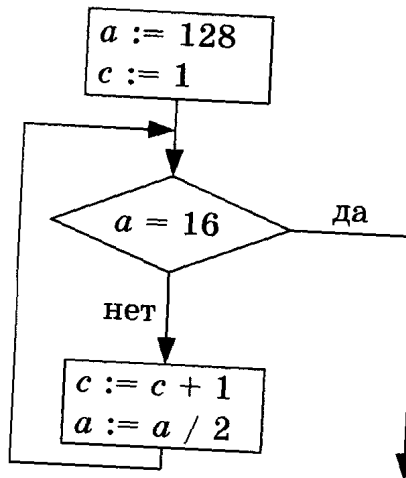
Ответ: \_\_\_\_\_.

б)

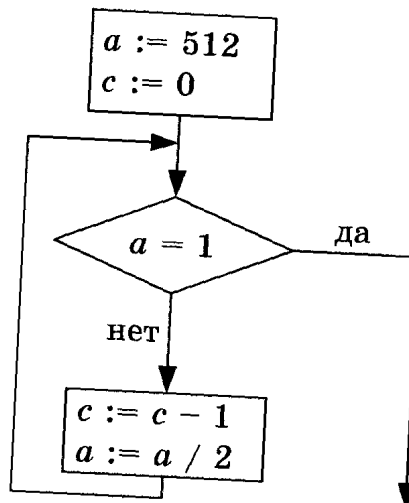


Ответ: \_\_\_\_\_

в)



Ответ: \_\_\_\_\_



Ответ: \_\_\_\_\_

**5.20.** Определите значение целочисленных переменных  $a$  и  $b$  после выполнения фрагмента программы:

а)

Бейсик	Паскаль	Алгоритмический
$a=3+2 \cdot 4$ $b=(a \text{ MOD } 10)+24$ $a=(b \setminus 10)+1$ \ и MOD — операции, вычисляющие результат деления нацело первого аргумента на второй и остаток от деления соответственно	$a:=3+2 \cdot 4;$ $b:=(a \text{ mod } 10)+24;$ $a:=(b \text{ div } 10)+1$ {div и mod — операции, вычисляющие результат деления нацело первого аргумента на второй и остаток от деления соответственно}	$a:=3+2 \cdot 4$ $b:=\text{mod}(a,10)+24$ $a:=\text{div}(b, 10)+1$  div и mod — функции, вычисляющие результат деления нацело первого аргумента на второй и остаток от деления соответственно

- 1)  $a = 3, b = 24$
- 2)  $a = 3, b = 25$
- 3)  $a = 4, b = 25$
- 4)  $a = 4, b = 35$

б)

Бейсик	Паскаль	Алгоритмический
$a=(3+2) \cdot 4$ $b=(a \text{ MOD } 5)+11$ $a=(b \setminus 3)+1$	$a:=(3+2) \cdot 4;$ $b:=(a \text{ mod } 5)+11;$ $a:=(b \text{ div } 3)+1$	$a:=(3+2) \cdot 4$ $b:=\text{mod}(a,5)+11$ $a:=\text{div}(b, 3)+1$

- 1)  $a = 4, b = 11$
- 2)  $a = 4, b = 12$
- 3)  $a = 5, b = 11$
- 4)  $a = 6, b = 15$

в)

Бейсик	Паскаль	Алгоритмический
$a=(3-2) \cdot 4$ $b=(a \text{ MOD } 5)+12$ $a=(b \setminus 3)+1$	$a:=(3-2) \cdot 4;$ $b:=(a \text{ mod } 5)+12;$ $a:=(b \text{ div } 3)+1$	$a:=(3-2) \cdot 4$ $b:=\text{mod}(a,5)+12$ $a:=\text{div}(b, 3)+1$

- 1)  $a = 2, b = 12$
- 2)  $a = 2, b = 16$
- 3)  $a = 3, b = 12$
- 4)  $a = 6, b = 16$

г)

Бейсик	Паскаль	Алгоритмический
$a=24-2 \cdot 4$ $b=(a \text{ MOD } 6)+12$ $a=(b \setminus 7)+1$	$a:=24-2 \cdot 4;$ $b:=(a \text{ mod } 6)+12;$ $a:=(b \text{ div } 7)+1$	$a:=24-2 \cdot 4$ $b:=\text{mod}(a,6)+12$ $a:=\text{div}(b, 7)+1$

- 1)  $a = 2, b = 22$
- 2)  $a = 4, b = 22$
- 3)  $a = 3, b = 16$
- 4)  $a = 6, b = 16$



## 5.21.

а) Дан фрагмент программы, обрабатывающей массив  $A$  из  $n$  элементов:

Бейсик	Паскаль	Алгоритмический
<pre> j=1 FOR i=1 TO n IF A(i) &lt; A(j) THEN j = i NEXT i s=j                     </pre>	<pre> j:=1; for i:=1 to n do begin     if A[i]&lt;A[j] then    j:=i end s:=j                     </pre>	<pre> j:=1 <u>нц</u> <u>для</u> i <u>от</u> 1 <u>до</u> n     <u>если</u> A[i]&lt;A[j] <u>то</u>         j:=i     <u>все</u> <u>кц</u> s:=j                     </pre>

Чему будет равно значение переменной  $s$  после выполнения данного алгоритма при любых значениях элементов массива  $A$ ?

- 1) Индексу минимального элемента в массиве  $A$  (первому из них, если минимальных элементов несколько)
- 2) Индексу минимального элемента в массиве  $A$  (последнему из них, если минимальных элементов несколько)
- 3) Индексу максимального элемента в массиве  $A$  (последнему из них, если максимальных элементов несколько)
- 4) Индексу минимального элемента в массиве  $A$  (последнему из них, если максимальных элементов несколько)

б) Дан фрагмент программы, обрабатывающей массив  $A$  из  $n$  элементов:

Бейсик	Паскаль	Алгоритмический
<pre> j = 1 FOR i = 1 TO n IF A(i) = A(j) THEN j = i NEXT i s=j                     </pre>	<pre> j:=1; for i:=1 to n do begin     if A[i]=A[j] then    j:=i end; s:=j                     </pre>	<pre> j:=1 <u>нц</u> <u>для</u> i <u>от</u> 1 <u>до</u> n     <u>если</u> A[i]=A[j] <u>то</u>         j:=i     <u>все</u> <u>кц</u> s:=j                     </pre>

Чему будет равно значение переменной  $s$  после выполнения данного алгоритма при любых значениях элементов массива  $A$ ?

- 1) Индексу элемента, равного первому и имеющему наибольший индекс
- 2) Индексу элемента, равного последнему и имеющему наименьший индекс
- 3) Значению  $n$
- 4) Единице

в) Дан фрагмент программы, обрабатывающей двумерный массив  $A$  размера  $n \times n$ .

Бейсик	Паскаль	Алгоритмический
<pre> k = 1 FOR i = 1 TO n c = A(i,k+2) A(i,k+2) = A(i,k) A(i,k) = c NEXT i                     </pre>	<pre> k:=1; for i:=1 to n do begin     c:=A[i,k+2];     A[i,k+2]:=A[i,k];     A[i,k]:=c end                     </pre>	<pre> k:=1 <u>нц</u> <u>для</u> i <u>от</u> 1 <u>до</u> n     c:=A[i,k+2]     A[i,k+2]:=A[i,k]     A[i,k]:=c <u>кц</u>                     </pre>

Представим массив в виде квадратной таблицы, в которой для элемента массива  $A[i, j]$  величина  $i$  является номером строки, а величина  $j$  — номером столбца, в котором расположен элемент. Тогда данный алгоритм меняет местами:

- 1) два столбца в таблице
- 2) две строки в таблице
- 3) элементы диагонали и  $k$ -й строки таблицы
- 4) элементы диагонали и  $k$ -го столбца таблицы

г) Дан фрагмент программы, обрабатывающей двумерный массив  $A$  размера  $n \times n$ .

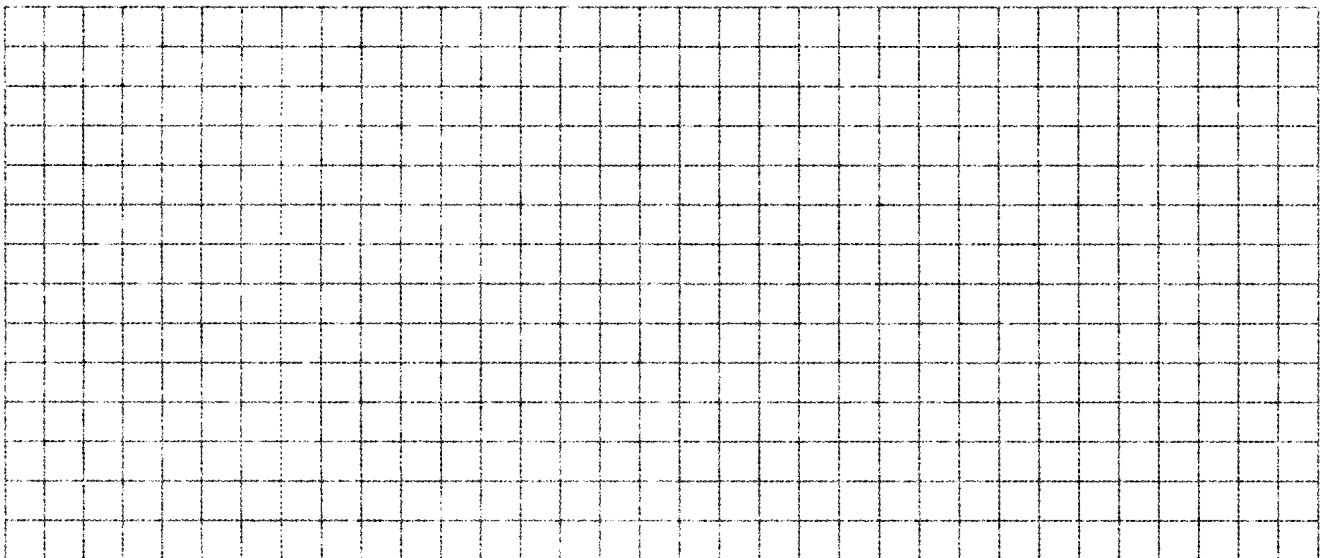
Бейсик	Паскаль	Алгоритмический
<pre>k = 3 FOR i = 1 TO n   c = A(i,i)   A(i,i) = A(i,k)   A(i,k) = c NEXT i</pre>	<pre>k:=3; for i:=1 to n do begin   c:=A[i,i];   A[i,i]:=A[i,k];   A[i,k]:=c end</pre>	<pre>k:=3 <u>НЦ</u> <u>для i от 1 до n</u> c:=A[i,i]   A[i,i]:=A[i,k]   A[i,k]:=c <u>КЦ</u></pre>

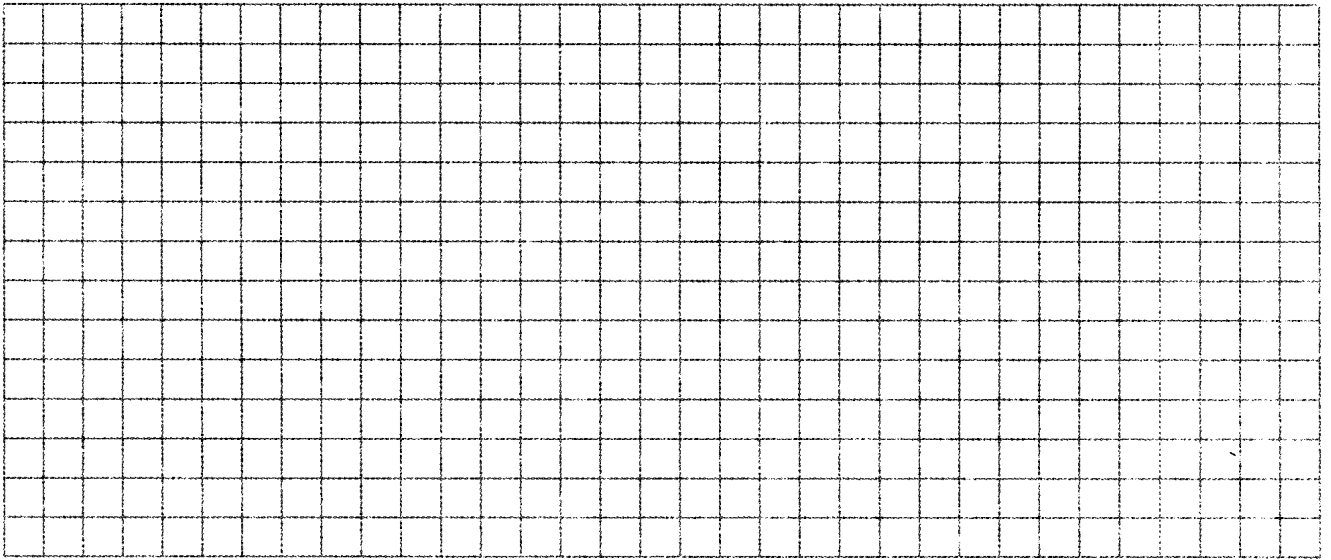
Представим массив в виде квадратной таблицы, в которой для элемента массива  $A[i, j]$  величина  $i$  является номером строки, а величина  $j$  — номером столбца, в котором расположен элемент. Тогда данный алгоритм меняет местами:

- 1) два столбца в таблице
- 2) две строки в таблице
- 3) элементы диагонали и  $k$ -й строки таблицы
- 4) элементы диагонали и  $k$ -го столбца таблицы

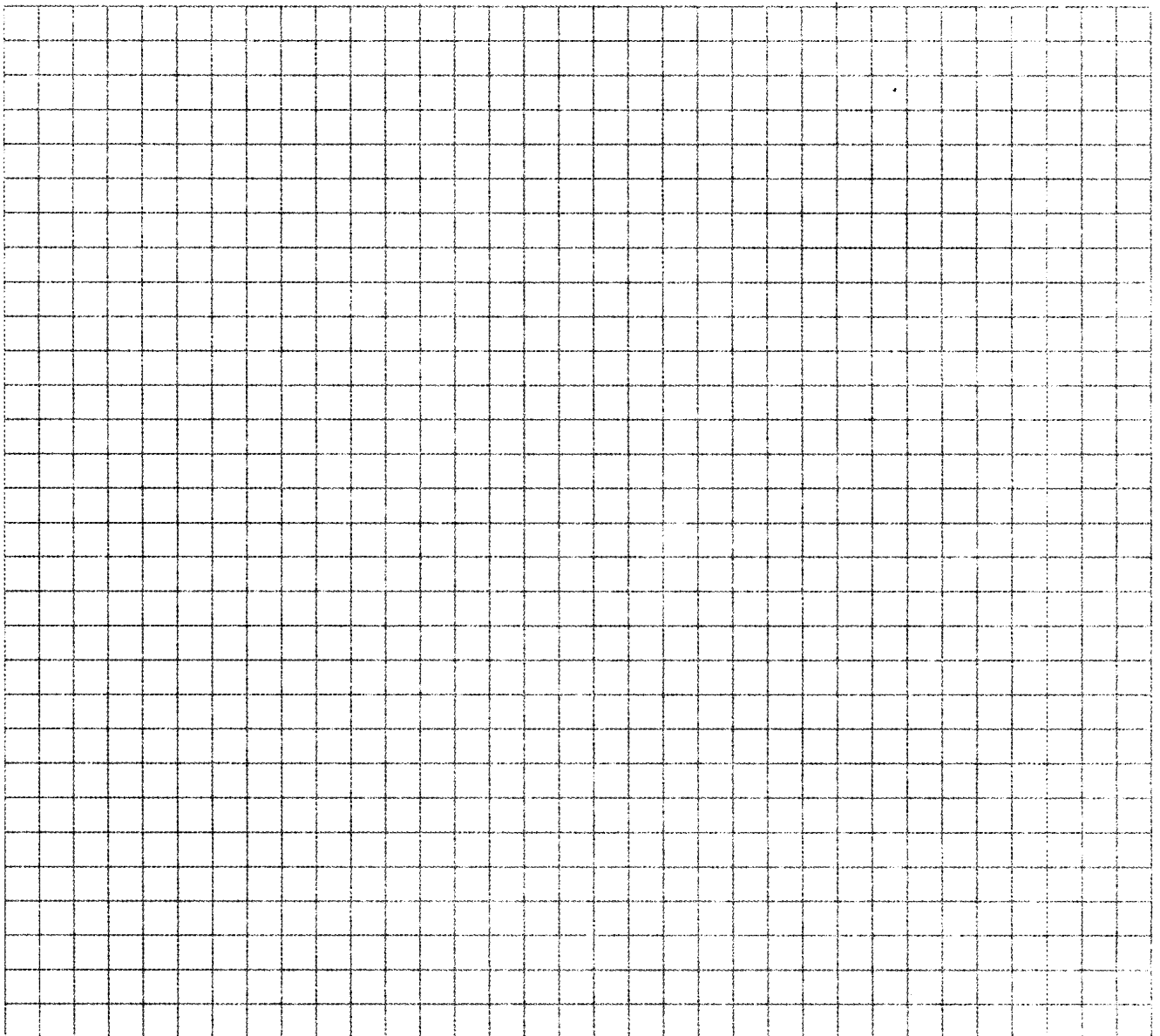
**5.22.** Составить и записать в виде программы на любом языке программирования или в виде блок-схемы или на естественном языке алгоритм, выполняющий:

а) поиск минимального нечетного элемента в заданном целочисленном массиве из 50 элементов. Известно, что в массиве есть хотя бы один нечетный элемент.

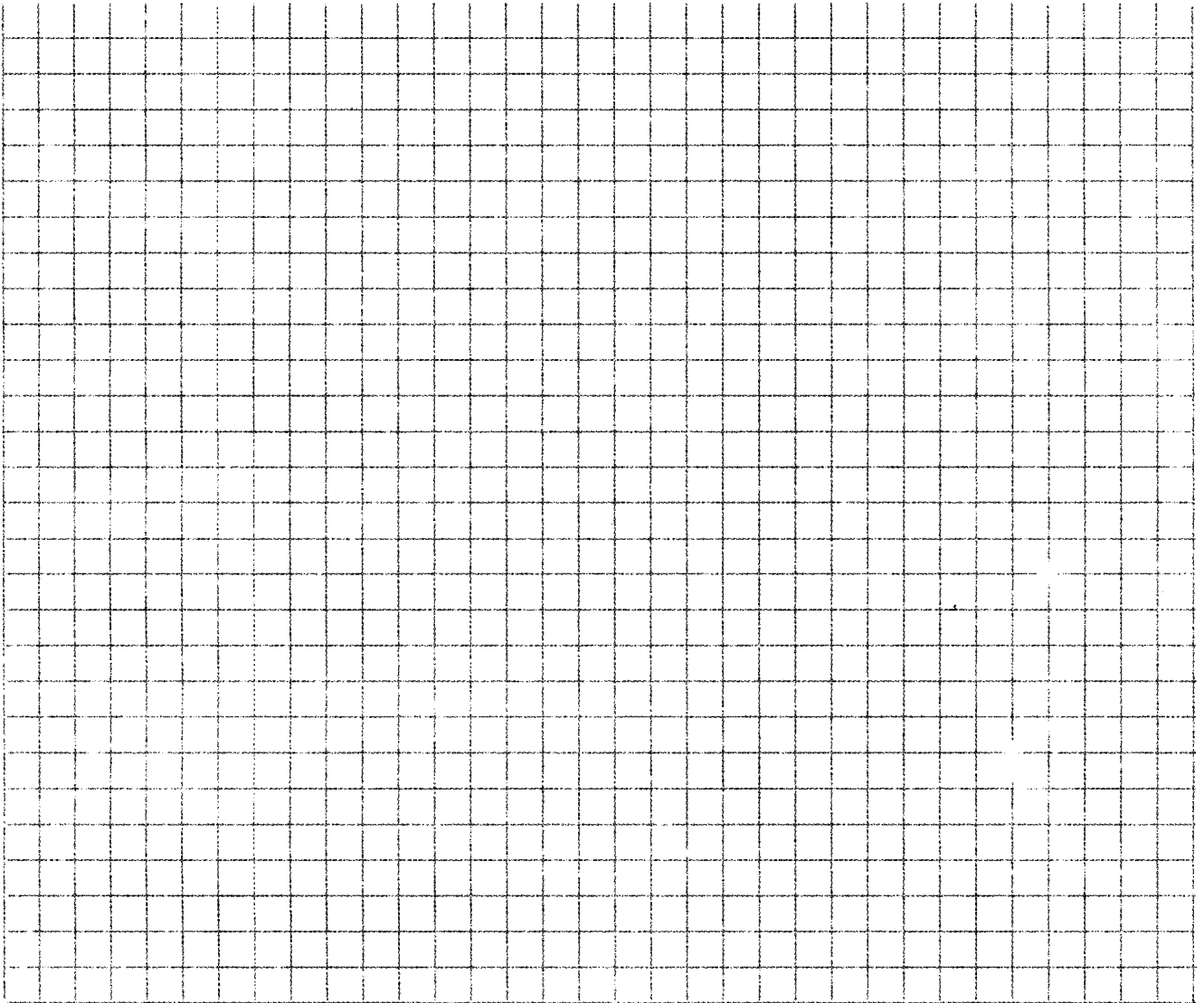




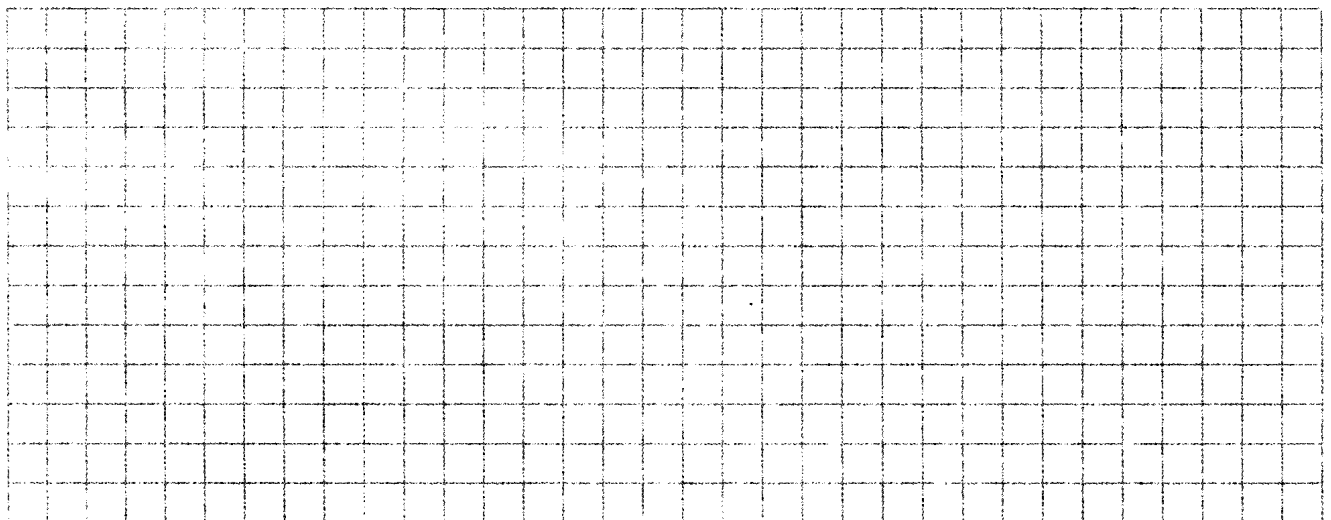
**поиск среднего арифметического четных элементов в заданном целочисленном массиве из 50 элементов. Известно, что в массиве есть хотя бы один четный элемент.**

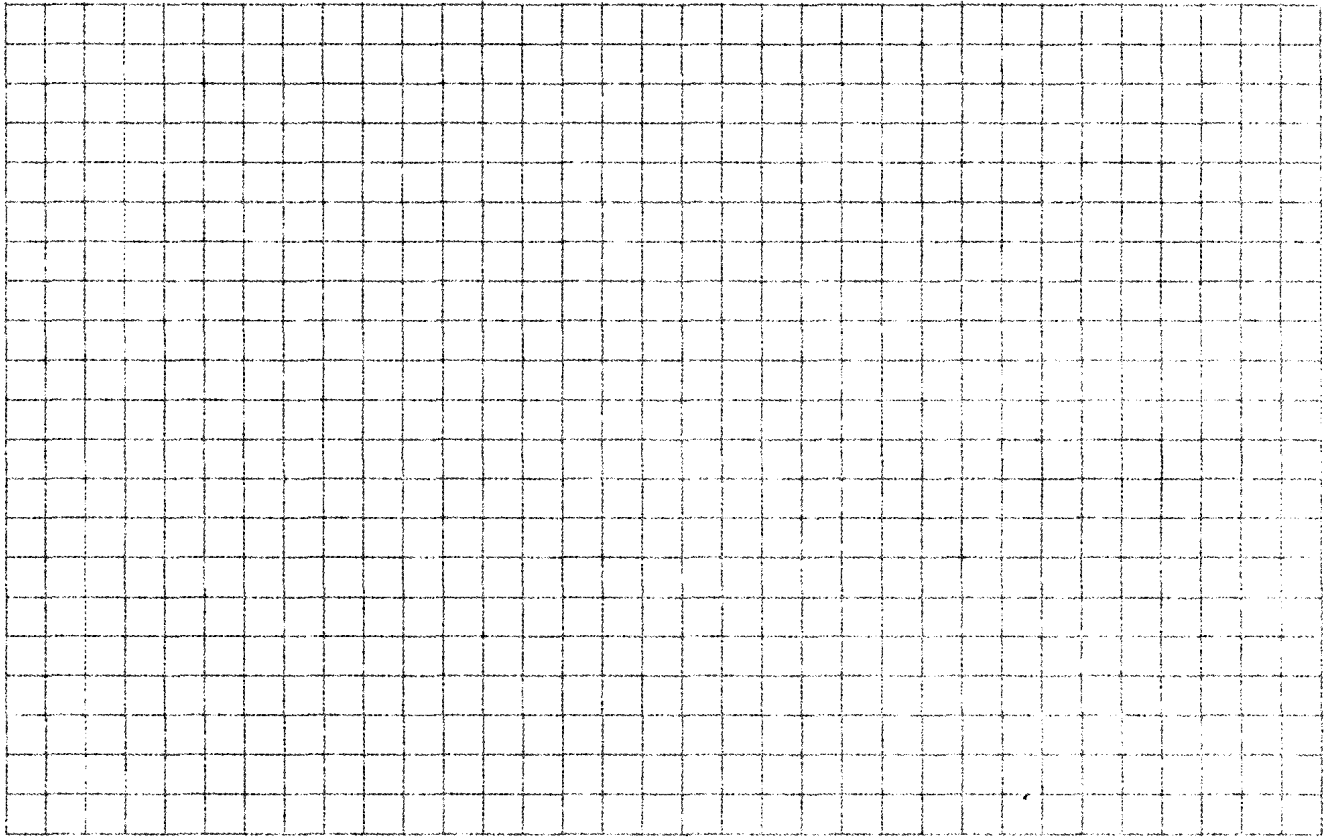


- в) поиск разности максимально четного и минимального нечетного элементов в заданного целочисленного массиве из 50 элементов. Известно, что в массиве есть и четные, и нечетные элементы.



- г) поиск второго после максимального по величине элемента в заданном целочисленном массиве из 50 различных элементов.





### 5.23.

- a) Требовалось написать программу, которая решает неравенство  $ax - b < 0$  относительно  $x$  для любых чисел  $a$  и  $b$ , введенных с клавиатуры. Все числа считаются действительными. Программист торопился и написал программу неправильно.

Программа на Паскале	Программа на Бейсике	Программа на Си
<pre>var a,b,x: real; begin readln(a,b); if a = 0 then write('любое число') else if a &lt; 0 then write('x &gt;',b/a) else write('x &lt;', b/a); end.</pre>	<pre>INPUT a, b IF a = 0 THEN PRINT "любое число" ELSE IF a &lt; 0 THEN PRINT "x &gt;",b/a ELSE PRINT "x &lt;",b/a END</pre>	<pre>void main(void) { float a,b,x; scanf("%f%f", &amp;a,&amp;b); if (a==0) printf("любое число"); else if (a&lt;0) printf("x&gt;%f",b/a); else printf("x&lt;%f",b/a); }</pre>

Последовательно выполните два задания:

- 1) приведите пример таких чисел  $a$ ,  $b$ ,  $x$ , при которых программа неверно решает поставленную задачу;
- 2) укажите, как нужно доработать программу, чтобы не было случаев ее неправильной работы. (Это можно сделать несколькими способами, поэтому можно указать любой способ доработки исходной программы.)

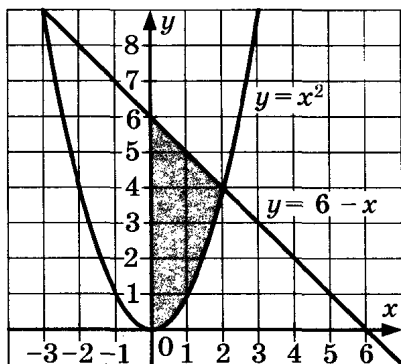
- б) требовалось написать программу, которая решает уравнение « $ax = b$ » относительно  $x$  для любых чисел  $a$  и  $b$ , введенных с клавиатуры. Все числа считаются действительными. Программист торопился и написал программу неправильно.

Программа на Паскале	Программа на Бейсике	Программа на Си
<pre>var a, b: real; begin   readln(a,b);   if b = 0 then     write('x = 0')   else     if a = 0 then       write('нет решений')     else       write('x = ',b/a);     end. end.</pre>	<pre>INPUT a, b IF b = 0 THEN   PRINT "x = 0" ELSE   IF a = 0 THEN     PRINT "нет решений"   ELSE     PRINT "x=",b/a   END END</pre>	<pre>void main(void) { float a,b;   scanf("%f%f",     &amp;a,&amp;b);   if (b==0)     printf("x=0");   else     if (a==0)       printf("нет решений");     else       printf("x=%f",b/a); }</pre>

Последовательно выполните два задания:

- 1) приведите пример таких чисел  $a$ ,  $b$ ,  $x$ , при которых программа работает неправильно;
- 2) укажите, как нужно доработать программу, чтобы не было случаев ее неправильной работы. (Это можно сделать несколькими способами, поэтому можно указать любой способ доработки исходной программы.)

в)



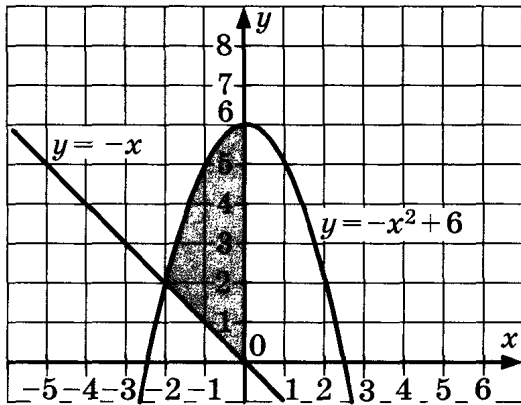
Требовалось написать программу, которая вводит с клавиатуры координаты точки на плоскости ( $x$ ,  $y$  — действительные числа) и определяет принадлежность точки закрашенной области, включая ее границы. Программист торопился и написал программу неправильно.

Программа на Паскале	Программа на Бейсике	Программа на Си
<pre>var x,y: real; begin   readln(x,y);   if y&gt;=sqr(x) then     if y&lt;=6-x then       write('принадлежит')     else       write('не принадлежит')   end. end.</pre>	<pre>INPUT x, y IF y&gt;=x * x THEN   IF y&lt;= 6-x THEN     PRINT "принадлежит"   ELSE     PRINT "не принадлежит"   ENDIF ENDIF END</pre>	<pre>void main(void) { float x,y;   scanf("%f%f",&amp;x,&amp;y);   if (y&gt;=x * x)     if (y&lt;=6-x)       printf("принадлежит");     else       printf("не принадлежит"); }</pre>

Последовательно выполните следующее:

- 1) приведите пример таких чисел  $x$ ,  $y$ , при которых программа работает неправильно;
- 2) укажите, как нужно доработать программу, чтобы не было случаев ее неправильной работы. (Это можно сделать несколькими способами, поэтому можно указать любой способ доработки исходной программы.)

г)



Требовалось написать программу, которая вводит с клавиатуры координаты точки на плоскости ( $x, y$  — действительные числа) и определяет принадлежность точки закрашенной области, включая ее границы. Программист торопился и написал программу неправильно.

Программа на Паскале	Программа на Бейсике	Программа на Си
<pre>var x,y: real; begin   readln(x,y);   if y&lt;=6-sqr(x) then     if y&gt;=-x then       write('принадлежит')     else       write('не принадлежит')   end.</pre>	<pre>INPUT x, y IF y&lt;=6-x * x THEN   IF y&gt;= 6-x THEN     PRINT "принадлежит"   ELSE     PRINT "не принадлежит"   ENDIF ENDIF END</pre>	<pre>void main(void) { float x,y;   scanf("%f%f",&amp;x,&amp;y);   if (y&lt;=6-x * x)     if (y&gt;=6-x)       printf("принадлежит");   else     printf("не принадлежит"); }</pre>

Последовательно выполните следующее:

- 1) приведите пример таких чисел  $x, y$ , при которых программа работает неправильно;
- 2) укажите, как нужно доработать программу, чтобы не было случаев ее неправильной работы. (Это можно сделать несколькими способами, поэтому можно указать любой способ доработки исходной программы.)

## 5.24.

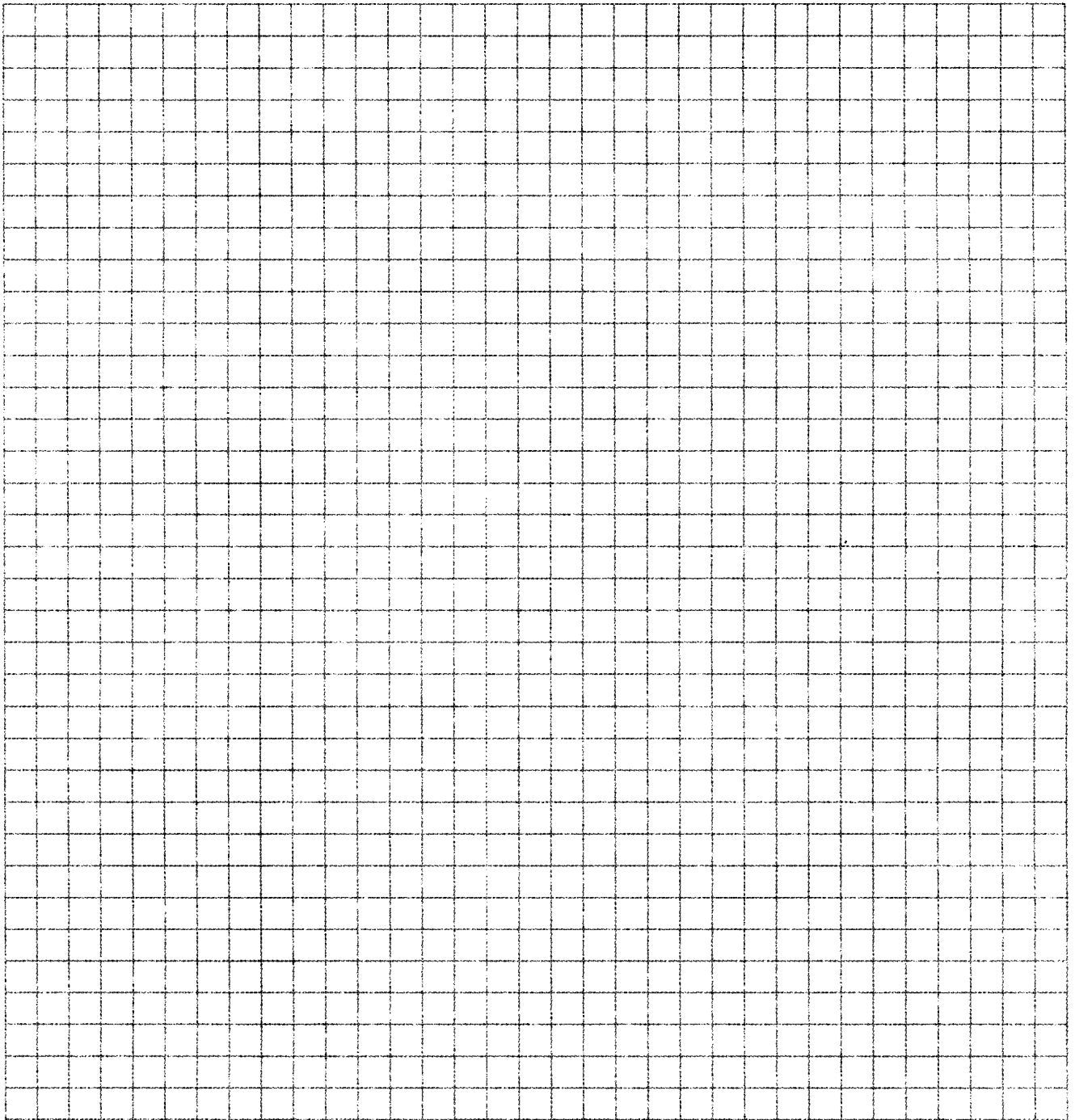
- а) На вход программе подаются сведения о сотрудниках некоторой организации. В первой строке сообщается количество сотрудников  $N$ , которое не меньше 5, но не превосходит 1000, каждая из следующих  $N$  строк имеет следующий формат: <Фамилия> <Должность> <Стаж> <Оклад> <Премия>, где
- <Фамилия> — строка, состоящая не более чем из 20 символов;
  - <Должность> — строка, состоящая не более чем из 15 символов;
  - <Стаж> <Оклад> <Премия> — целые числа. Под стажем понимается количество полных лет, отработанных сотрудником в данной организации.
- Оклад и премия — целые числа, не превышающие 1 000 000. Эти элементы входной строки отделены друг от друга одним пробелом. Пример входной строки:

Иванов механик 15 25500 5000

Требуется написать программу (укажите язык программирования, который вы используете), которая будет выводить на экран фамилии, должности и стаж трех сотрудников с наибольшей зарплатой. Если среди остальных есть сотрудники, с такой же зар-

платой, что и один из этих трех, то следует также вывести и их фамилии, должности и стаж.

Зарплата сотрудника равна сумме его оклада и премии.



На вход программе подаются сведения о сотрудниках некоторой организации. В первой строке сообщается количество сотрудников  $N$ , которое не меньше 5, но не превосходит 1000, каждая из следующих  $N$  строк имеет следующий формат: <Фамилия> <Должность> <Стаж> <Оклад> <Премия>, где  
<Фамилия> — строка, состоящая не более чем из 20 символов;  
<Должность> — строка, состоящая не более чем из 15 символов;  
<Стаж> <Оклад> <Премия> — целые числа. Под стажем понимается количество полных лет, отработанных сотрудником в данной организации.



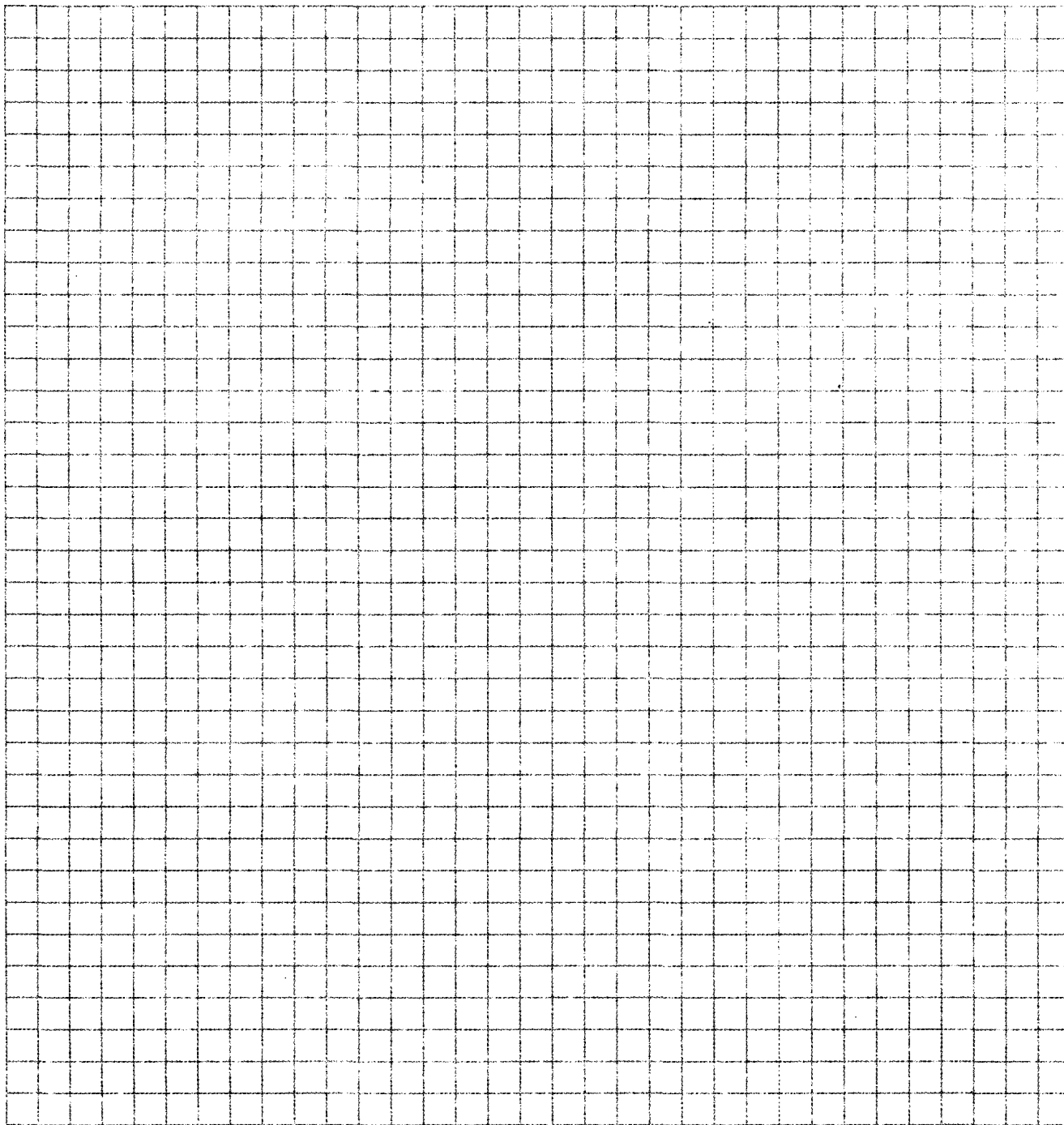
Оклад и премия — целые числа, не превышающие 1 000 000.

Эти элементы входной строки отделены друг от друга одним пробелом. Пример входной строки:

Иванов механик 15 25500 5000

Требуется написать программу (укажите язык программирования, который вы используете), подсчитывающую среднюю зарплату сотрудников, стаж которых больше среднего по организации.

Зарплата сотрудника равна сумме его оклада и премии.



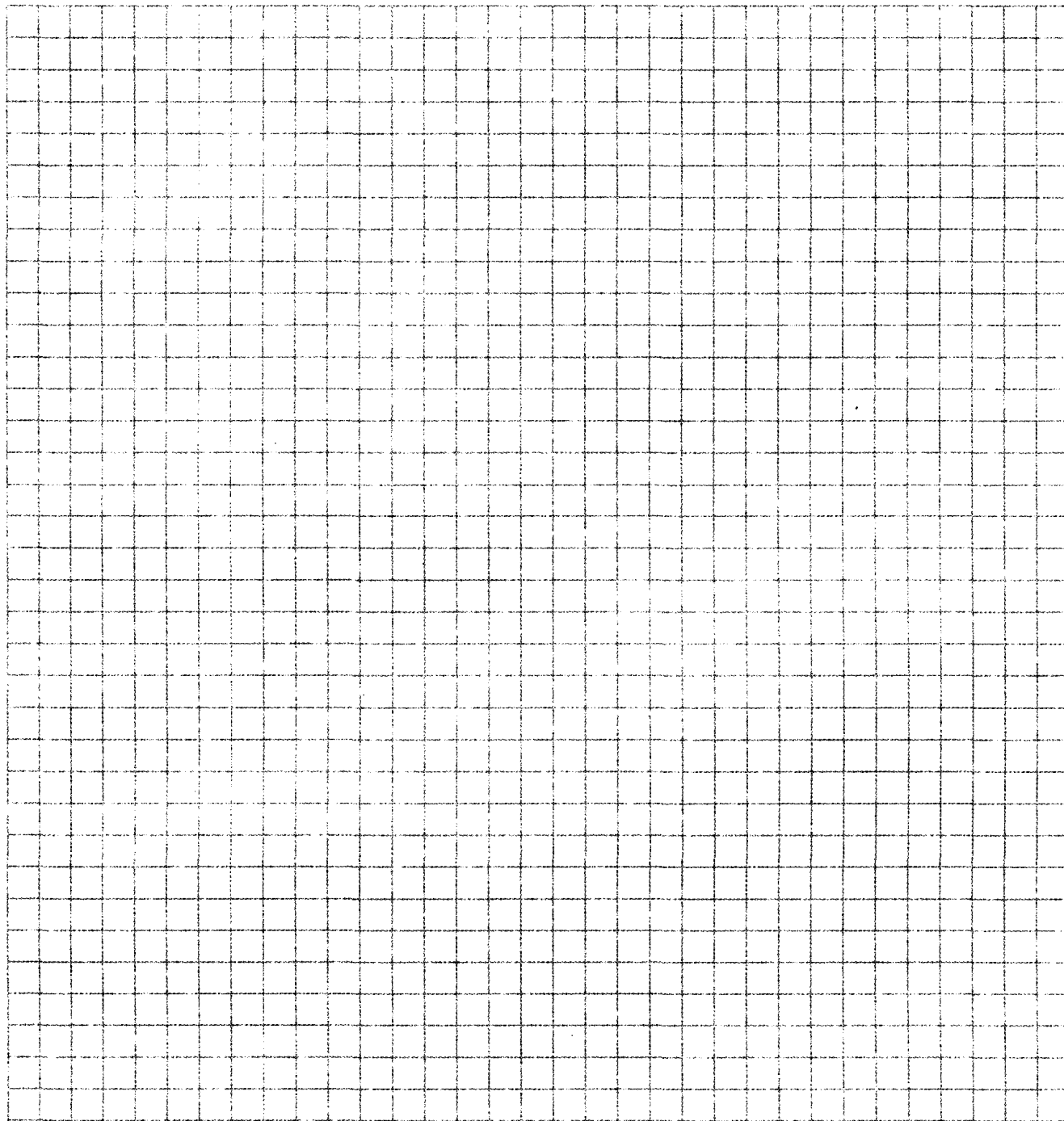
На вход программе подаются строчные английские буквы. Ввод этих символов заканчивается точкой (другие символы, отличные от «.» и букв «a» ... «z», во входных данных отсутствуют; в программе на языке Бейсик символы можно вводить по одному в строке, пока не будет введена точка).

Требуется написать эффективную программу (укажите язык программирования, который вы используете), которая будет печатать в алфавитном порядке только те буквы, которые ровно три раза подряд встретились во входной последовательности. Каждая буква должна быть распечатана только один раз. Точка при этом не учитывается.

Например, пусть на вход подаются следующие символы: `btfgbbffjrtatbama`.

В данном случае программа должна вывести:

`Aft`



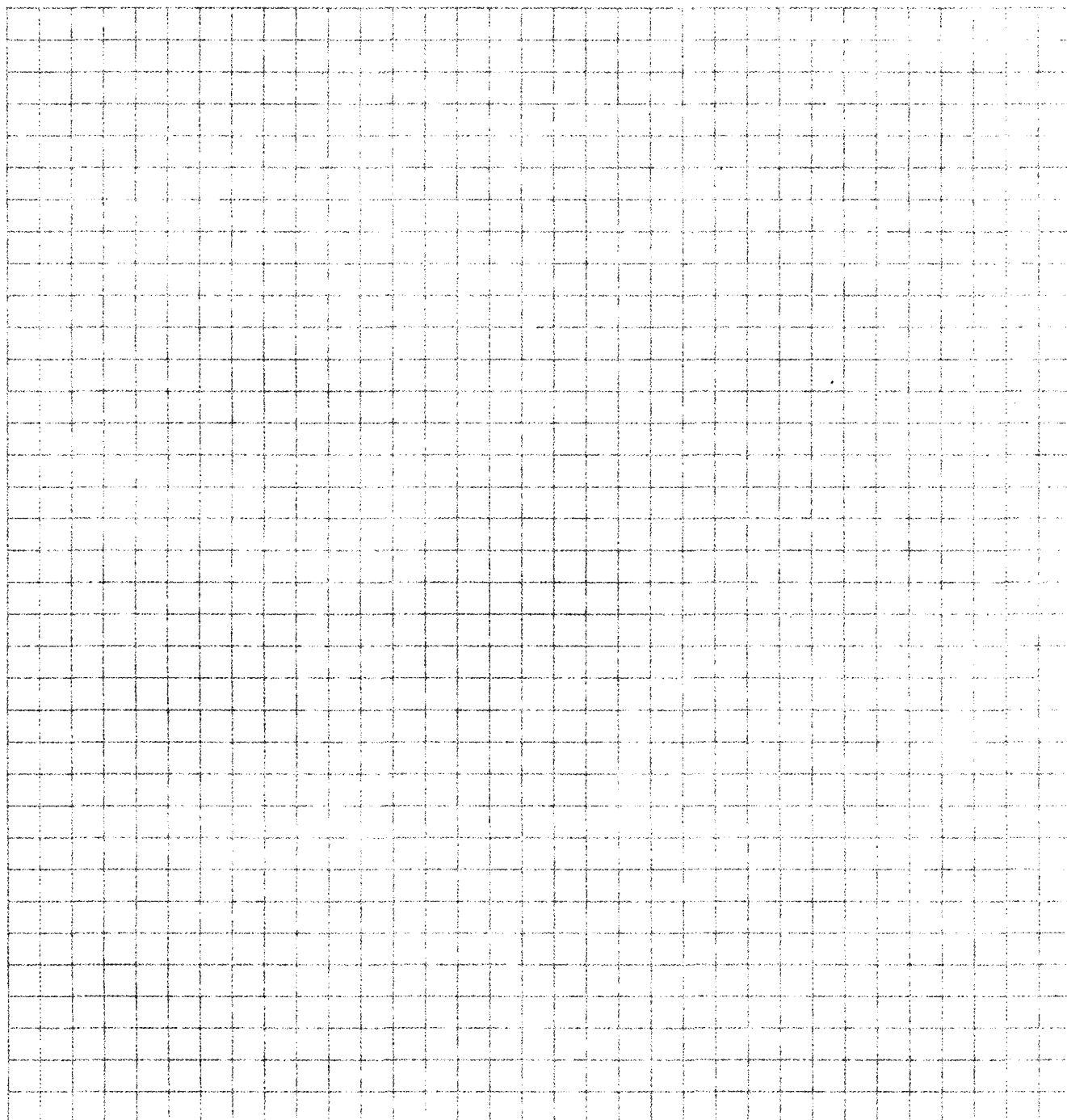
На вход программе подаются строчные английские буквы. Ввод этих символов заканчивается точкой (другие символы, отличные от «.» и букв «a» ... «z», во входных данных отсутствуют; в программе на языке Бейсик символы можно вводить по одному в строке, пока не будет введена точка).

Требуется написать эффективную программу (укажите язык программирования, который вы используете), которая будет печатать в алфавитном порядке только те буквы, которые хотя бы однократно ровно три раза подряд встретились во входной последовательности. Каждая буква должна быть распечатана только один раз. Точка при этом не учитывается.

Например, пусть на вход подаются следующие символы: `bbbffjrtatbbbamaaa`.

В данном случае программа должна вывести:

`ab`



## 6. Информационные модели

### Основные понятия

В задачах ЕГЭ на анализ информационных моделей обычно используются две основные *информационные модели* — таблицы и схемы. Как правило, умения понимать эти модели достаточно для решения данного рода задач.

Информация в *таблице* построена по следующему принципу: на пересечении строки и столбца находится информация, характеризующая комбинацию этой строки и столбца.

*Пример.* Таблица умножения. На пересечении строки номер  $X$  и столбца  $Y$  находится результат умножения  $X$  на  $Y$ .

Информация на схеме построена по следующему принципу: если между объектами схемы имеется связь, она отображается линией, соединяющей названия этих объектов на схеме.

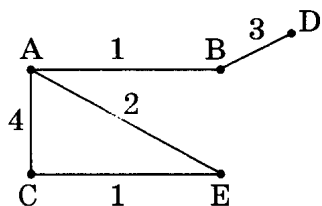
*Пример.* Схема железнодорожных станций (или метрополитена). Между двумя станциями можно проехать, если на схеме нарисована линия, соединяющая эти станции.

### Практическая часть

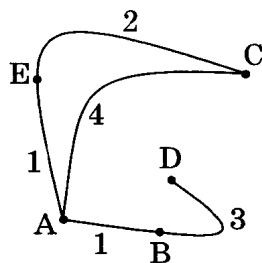
**6.1.** В таблице приведена стоимость перевозок между соседними железнодорожными станциями. Укажите схему, соответствующую таблице.

	A	B	C	D	E
A		1	4		1
B	1			3	
C	4				2
D		3			
E	1		2		

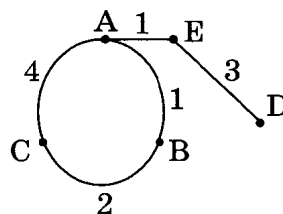
1)



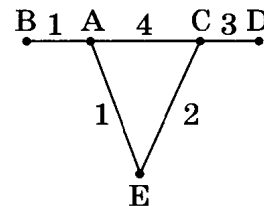
2)



3)



4)



*Рекомендации.* Анализируем представленную информацию. Таблица отображает стоимость для всех комбинаций станций. То есть на пересечении строки  $X$  и столбца  $Y$  находится число, соответствующее стоимости проезда от станции  $X$  до станции  $Y$ . Отсутствие числа в ячейке соответствует отсутствию дороги между этими станциями.

На схеме названия станций соединены линиями с числами. Судя по всему, линия, соединяющая станцию  $X$  со станцией  $Y$  отражает наличие пути между станциями  $X$  и  $Y$ , а число на этой линии — стоимости проезда по этому пути.

Задача нахождения соответствия между схемой и таблицей, таким образом, сводится к задаче идентичности связей и их стоимостей.

Так, по таблице из станции  $A$  можно добраться до станции  $B$  за 1, до станции  $C$  за 4, и до станции  $E$  за 1.

Схема 1 не соответствует этому (путь из  $A$  в  $E$  «стоит» 2). Дальше не будем ее рассматривать.

Схемы 2, 3 и 4 соответствуют.

По таблице от станции  $B$  можно добраться до станции  $B$  за 1, а до станции  $D$  за 3.

Схема 2 соответствует этому. Схемы 3 и 4 не соответствуют этому (путь из  $B$  в  $D$  на них отсутствует).

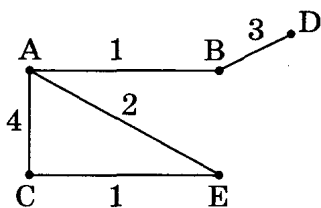
Значит, единственным возможным кандидатом остается только схема 2.

Для подстраховки можно проверить на соответствие остальные строки таблицы. Если обнаружится несоответствие, значит, либо ошиблись составители задачи (что очень маловероятно, ибо все задания по несколько раз проверяются), либо мы были невнимательны в одной из наших предыдущих проверок и отсеяли что-то правильное.

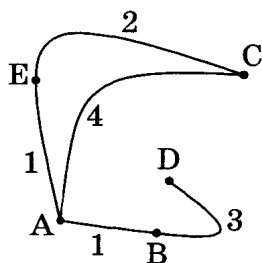
**6.2.** В таблице приведена стоимость перевозок между соседними железнодорожными станциями. Укажите схему, соответствующую таблице.

	A	B	C	D	E
A		1	4		1
B	1		2		
C	4	2			
D					3
E	1			3	

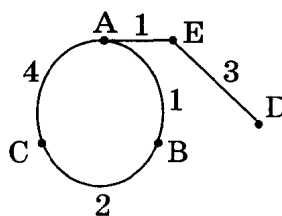
1)



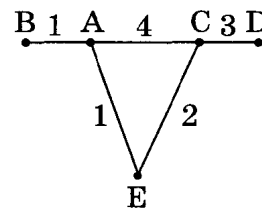
2)



3)



4)



**6.3.** Таблица стоимости перевозок устроена следующим образом: числа, стоящие на пересечениях строк и столбцов таблиц, означают стоимость проезда между соответствующими соседними станциями. Если пересечение строки и столбца пусто, то станции не являются соседними.

Укажите таблицу, для которой выполняется условие: «Минимальная стоимость проезда из  $A$  в  $B$  не больше 6».

Стоимость проезда по маршруту складывается из стоимостей проезда между соответствующими соседними станциями.

1)

	A	B	C	D	E
A			3	1	
B			4		2
C	3	4			2
D	1				
E		2	2		

2)

	A	B	C	D	E
A			3	1	1
B			4		
C	3	4			2
D	1				
E	1		2		

3)

	A	B	C	D	E
A			3	1	
B			4		1
C	3	4			2
D	1				
E		1	2		

4)

	A	B	C	D	E
A				1	
B			4		1
C		4		4	2
D	1		4		
E		1	2		

**Рекомендации.** Если таблица большая и плотно заполнена числами, то для решения такого рода задач требуется, вообще говоря, сложный алгоритм (например, алгоритм Дейкстры). Его изучение не входит в стандартный курс школьной информатики. Но в данном случае таблица небольшая и далеко не все ее клетки заполнены. Значит, можно ее достаточно эффективно решить, осуществив осмысленный перебор возможных вариантов. При решении рекомендуется воспользоваться двумя соображениями:

- 1) вариант пути, который становится больше или равен 6, но еще не достигает конечной точки (B) можно дальше не рассматривать;
- 2) возвращение в точку, через которую уже прошел путь (петля), не приведет к эффективному решению, поэтому такие ветки тоже можно не рассматривать.

Рассмотрим пример такого «осмысленного перебора»:

Таблица 1:

Из точки A два пути — в точку C (равен 3) и в точку D (равен 1).

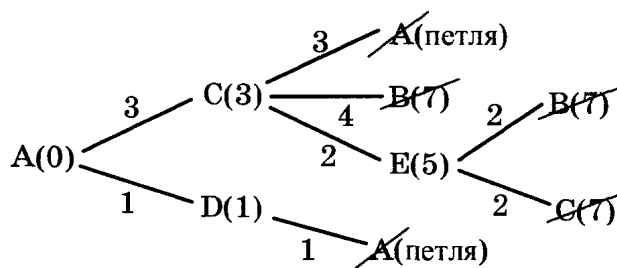
Если пошли в точку C (путь A—C): из нее три пути — в точку A (петля, не рассматриваем), в точку B (равен 4, в сумме  $3 + 4 = 7$ , больше 6, не рассматриваем) и в точку E (равен 2, итого путь A—C—E равен  $3 + 2 = 5$ ).

Из точки E только два пути, оба равны 2, то есть сумма  $5 + 2$  будет больше 6, дальше можно не рассматривать.

Если пошли в точку D (путь A—D): из нее только путь в точку A (петля).

Все возможные пути мы отбросили, до точки B за 6 добраться не удалось, значит, таблица варианта 1 нам не подходит.

Все это решение гораздо удобнее и эффективнее рисовать при помощи небольшой схемки, на которой рисуются возможные пути из стартовой точки и для каждой точки, в которую приходит путь, указывается текущая стоимость проезда:



Далее нужно таким же образом рассмотреть остальные 3 варианта таблицы. Один из них должен соответствовать условию.

Возможно решение этой задачи в обратную сторону, «с конца». В этом случае сначала рассматриваются все точки, из которых можно добраться до конечной точки (B). Потом — все точки, из которых можно добраться до этих, и т.д. Смысл решения точно такой же, нужно отбрасывать варианты с «петлями» и варианты, стоимость проезда в которых становится больше или равна 6.

Эти два способа решения («с начала» и «с конца») в обычном случае одинаковы. Мы рекомендуем предпочесть решение «с конца», если количество путей, входящих в конечную точку, меньше, чем количество путей, выходящих из стартовой (как, например, в таблице варианта 2).

- 6.4.** Грунтовая дорога проходит последовательно через населенные пункты А, В, С и D. При этом длина дороги между А и В равна 80 км, между В и С — 50 км, и между С и D — 10 км.

Между А и С построили новое асфальтовое шоссе длиной 40 км. Оцените минимально возможное время движения велосипедиста (в часах) из пункта А в пункт В, если его скорость по грунтовой дороге — 20 км/час, по шоссе — 40 км/час?

Ответ:

*Рекомендация.* Хотя внешне задача выглядит совсем не как только что рассмотренный класс задач, решается она таким же образом. Мы рекомендуем нарисовать схему возможных путей из стартовой точки (А) и рассчитать время проезда до точек, связанных с ней. При этом нужно не забыть, что время рассчитывается как длина пути, деленная на скорость. Потом примените тот же способ перебора, что и в предыдущей задаче (только теперь числа на линиях схемы будут показывать не стоимость проезда, а время в пути). Когда вы рассмотрите все возможные способы проезда (которые, конечно, не содержат «петель»), выберите среди них минимальное время.

- 6.5.** Между четырьмя местными аэропортами: ОКТЯБРЬ, БЕРЕГ, КРАСНЫЙ и СОСНОВО, ежедневно выполняются авиарейсы. Приведен фрагмент расписания перелетов между ними:

Аэропорт вылета	Аэропорт прилета	Время вылета	Время прилета
СОСНОВО	КРАСНЫЙ	06:20	08:35
КРАСНЫЙ	ОКТЯБРЬ	10:25	12:35
ОКТЯБРЬ	КРАСНЫЙ	11:45	13:30
БЕРЕГ	СОСНОВО	12:15	14:25
СОСНОВО	ОКТЯБРЬ	12:45	16:35
КРАСНЫЙ	СОСНОВО	13:15	15:40
ОКТЯБРЬ	СОСНОВО	13:40	17:25
ОКТЯБРЬ	БЕРЕГ	15:30	17:15
СОСНОВО	БЕРЕГ	17:35	19:30
БЕРЕГ	ОКТЯБРЬ	19:40	21:55







Аэропорт вылета	Аэропорт прилета	Время вылета	Время прилета
ВОСТОРГ	ГОРКА	13:10	17:15
ОЗЕРНЫЙ	ЗАРЯ	13:00	14:30
ОЗЕРНЫЙ	ВОСТОРГ	12:10	14:20
ГОРКА	ОЗЕРНЫЙ	11:15	15:30
ВОСТОРГ	ОЗЕРНЫЙ	12:35	14:50
ЗАРЯ	ОЗЕРНЫЙ	12:30	14:20
ВОСТОРГ	ЗАРЯ	10:30	12:15
ЗАРЯ	ГОРКА	14:40	16:45
ГОРКА	ЗАРЯ	15:15	17:20
ОЗЕРНЫЙ	ГОРКА	14:30	16:20

Ответ:

**6.9.** Между четырьмя местными аэропортами: ВОСТОРГ, ЗАРЯ, ОЗЕРНЫЙ и ГОРКА — ежедневно выполняются авиарейсы. Приведен фрагмент расписания перелетов между ними. Путешественник оказался в аэропорту ВОСТОРГ в полночь (0:00). Определите самое раннее время, когда он может попасть в аэропорт ГОРКА.

Аэропорт вылета	Аэропорт прилета	Время вылета	Время прилета
ВОСТОРГ	ГОРКА	10:15	14:30
ОЗЕРНЫЙ	ЗАРЯ	10:20	12:30
ОЗЕРНЫЙ	ВОСТОРГ	14:14	16:30
ГОРКА	ОЗЕРНЫЙ	15:40	17:35
ВОСТОРГ	ОЗЕРНЫЙ	10:05	12:20
ЗАРЯ	ОЗЕРНЫЙ	11:40	13:55
ВОСТОРГ	ЗАРЯ	9:40	11:50
ЗАРЯ	ГОРКА	12:20	14:10
ГОРКА	ЗАРЯ	13:15	15:40
ОЗЕРНЫЙ	ГОРКА	12:30	14:25

Ответ:

## 7. Определение выигрышной стратегии игры (Анализ и построение дерева игры)

### Основные понятия

При решении заданий данного типа предполагается, что вы проанализируете возможные ходы обоих игроков и определите, кто из них выигрывает и как он для этого должен ходить.

Сначала рассмотрим пример решения задачи.

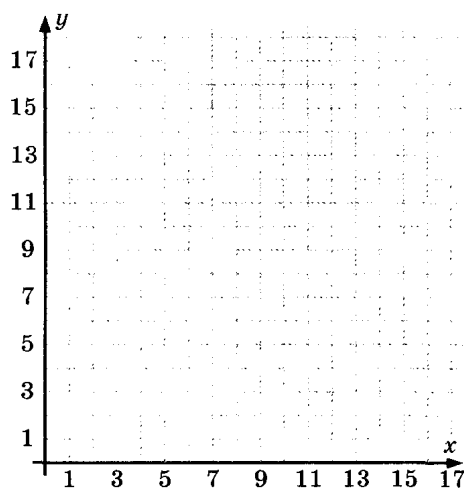
Мы предлагаем решать задачу «с конца».

Для тех позиций, про которые точно известно, что они ведут к выигрышу, определим «предыдущие» позиции как проигрышные, для них снова определим выигрышные (только теперь уже за два хода), для них снова определим проигрышные. И так до тех пор, пока не получим, что стартовая (начальная) позиция проигрышная или выигрышная. Тогда будет ясно, кто выигрывает, и останется проанализировать, как он должен ходить, чтобы выиграть.

Рассмотрим *пример*.

**7.1.** Два игрока играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. В начале игры фишка находится в точке с координатами  $(5, 2)$ . Ход состоит в том, что игрок перемещает фишку из точки с координатами  $(x, y)$  в одну из трех точек: или в точку с координатами  $(x + 3, y)$ , или в точку с координатами  $(x, y + 3)$ , или в точку с координатами  $(x, y + 4)$ . Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами  $(0, 0)$  не меньше 13 единиц. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.

Нарисуем координатную плоскость  $(x, y)$ , на которой будем отмечать возможные позиции фишки. Нам понадобится 1-я четверть координатной плоскости, причем по обеим осям нас будут интересовать точки примерно до 17-й координаты.



В данной задаче «концом» является положение, когда «расстояние по прямой от фишки до точки с координатами  $(0, 0)$  не меньше 13 единиц».

Нам нужно точно знать, какие точки плоскости этой находятся на расстоянии меньше 13 единиц от начала координат, а какие — нет. Если у вас на экзамене будет циркуль и бумага

в клетку (и если циркулем вам разрешат пользоваться) — отлично, просто нарисуйте четверть окружности с центром в точке  $(0, 0)$  и радиусом 13.

Если у вас чего-нибудь из этого нет, придется для начала вычислить те целочисленные координаты, которые лежат на границе этой окружности.

Для точек  $(0, 13)$  и  $(13, 0)$  очевидно, что они лежат на окружности.

Если одна координата равна 12, то (по теореме Пифагора) вторая координата равна

$$\sqrt{13^2 - 12^2} = \sqrt{169 - 144} = \sqrt{25} = 5.$$

То есть точки  $(12, 5)$  и  $(5, 12)$  тоже лежат на окружности.

Для координаты 11 вторая координата получается

$$\sqrt{13^2 - 11^2} = \sqrt{169 - 121} = \sqrt{48} < \sqrt{49} = 7.$$

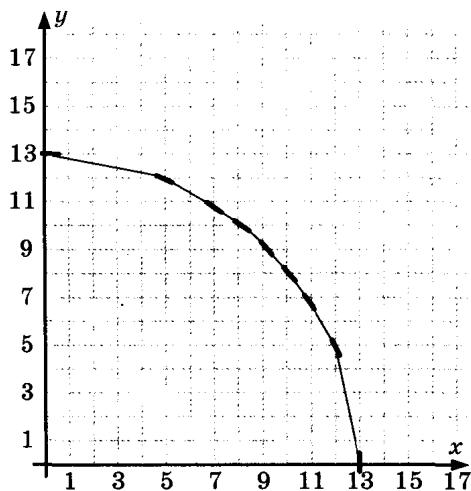
То есть точки  $(11, 7)$  и  $(7, 11)$  лежат за окружностью.

Для координаты 10 вторая координата —  $\sqrt{13^2 - 10^2} = \sqrt{169 - 100} = \sqrt{69} > \sqrt{64} = 8$ .

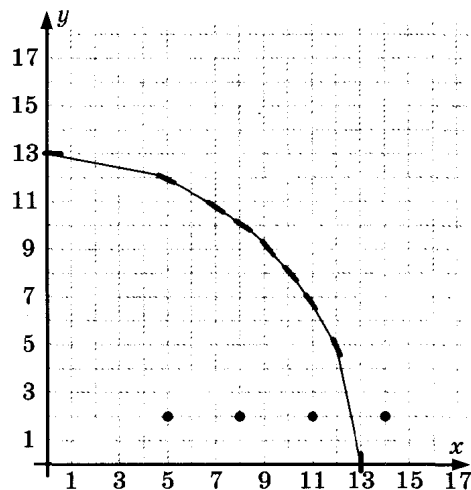
То есть точки  $(10, 8)$  и  $(8, 10)$  лежат внутри окружности.

Для координаты 9 вторая координата —  $\sqrt{13^2 - 9^2} = \sqrt{169 - 81} = \sqrt{88} > \sqrt{81} = 9$ . То есть точка  $(9, 9)$  лежит внутри окружности.

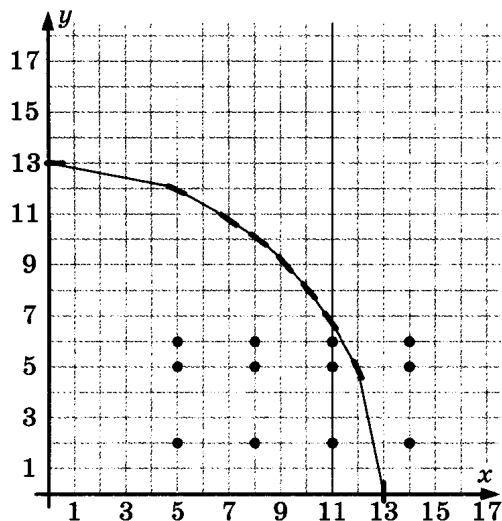
По полученным точкам нарисуем окружность (радиуса 13):



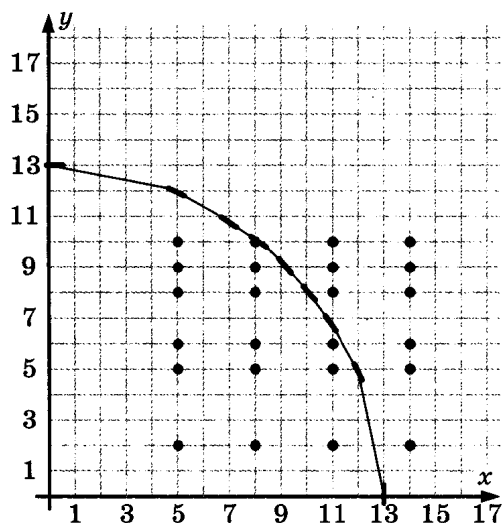
Теперь обозначим (на рисунке — кружком) все позиции, начиная от стартовой  $(5, 2)$ , в которых может находиться фишка в процессе игры, пока не выйдет за пределы нарисованной окружности. Для этого от стартовой позиции обозначим точки вправо с шагом 3 (так как есть ход  $(x + 3, y)$ ):



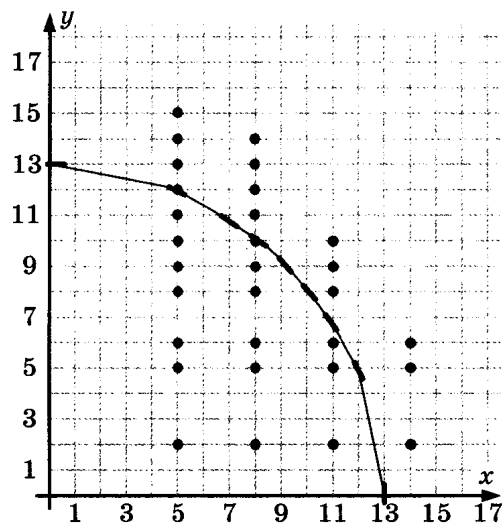
Затем для всех нарисованных точек обозначим точки, находящиеся выше этих на 3 и на 4 единицы (так как есть ходы  $(x, y + 3)$  и  $(x, y + 4)$ ):



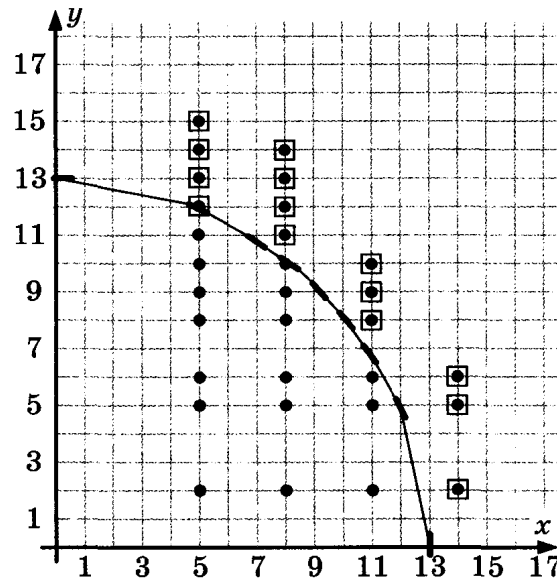
От каждой из этих точек тоже можно пойти на 3 или на 4 единицы вверх:



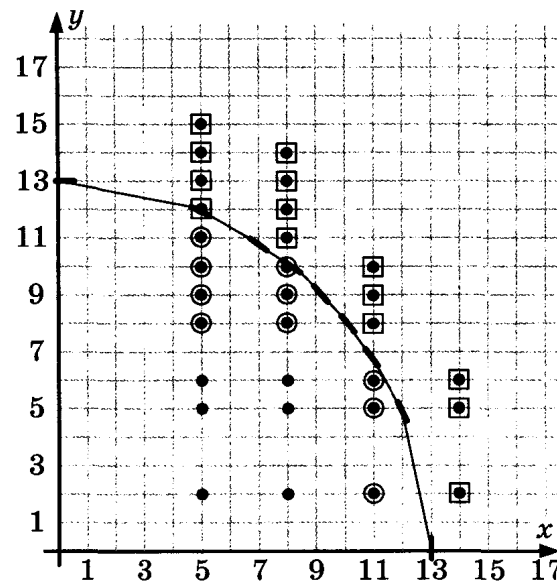
И так как от этих точек тоже можно пойти на 3 или 4 единицы вверх, получаем все точки, которые достижимы при данных ходах из стартовой позиции  $(5, 2)$  и которые возможны в игре (остальные нет смысла рисовать, потому что игра закончится раньше):



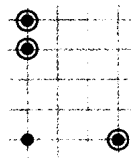
Теперь про все обозначенные точки, которые лежат на окружности или за окружностью (расстояние от них до точки  $(0, 0)$  не меньше 13 единиц) можем сказать, что они являются проигрышными. Это значит, что если игроку достается ход с фишкой, стоящей в одной из этих позиций, то его противник только что выиграл на своем ходу. Обведем эти точки в квадратики (на рисунке они будут обозначены как  $\square$ ):



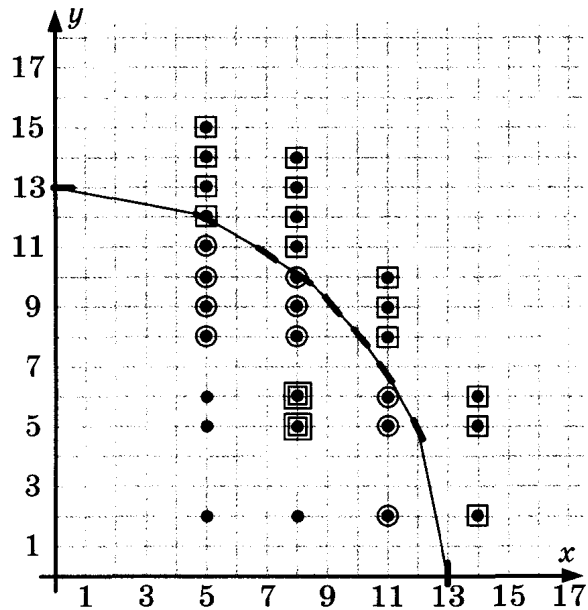
Для всех этих точек теперь можем найти точки, выигрышные за один ход (из которых имеется ход, который приводит фишку в одну из обозначенных точек). Обведем эти точки кружками (на рисунке они будут обозначены как  $\odot$ ):



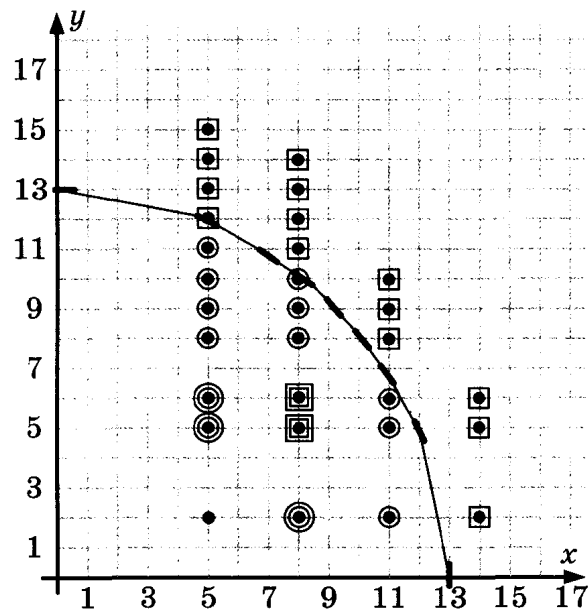
Теперь более сложный шаг — среди оставшихся необведенных точек найти те, из которых куда бы не пошел игрок, его соперник окажется в выигрышной позиции. Это те точки, у которых справа на 3 единицы и сверху на 3 и на 4 единицы находятся позиции, обозначенные кружком ( $\odot$ ). Обозначим это на схеме:



Таких точек оказалось всего две. Это проигрышные позиции в два хода. Обведем их на рисунке двойным квадратиком ( $\square$ ):



Для этих точек найдем теперь позиции, выигрышные за два хода (из которых имеется ход, приводящий фишку в данную проигрышную позицию). Таких точек в данном случае будет три. Обведем их двойным кружком ( $\odot$ ):



Теперь становится видно, что стартовая позиция является проигрышной — куда бы ни пошел из нее игрок, его соперник оказывается в выигрышной позиции и за два хода выигрывает.

Таким образом, мы решили задачу.

Как вы думаете, сколько баллов (из трех возможных) мы заработали за это задание?

Надеемся, вы уже прочитали критерии оценивания этой задачи и поняли, что пока мы заработали ноль баллов. Если вы еще не прочитали критерии оценивания, внимательно прочитайте их сейчас:

Указания по оцениванию	Баллы
Правильное указание выигрывающего игрока и его ходов со строгим доказательством правильности (с помощью или без помощи дерева игры).	3
Правильное указание выигрывающего игрока, стратегии игры, приводящей к победе, но при отсутствии доказательства ее правильности.	2
При наличии в представленном решении одного из пунктов: 1. Правильно указан выигрывающий игрок и его первый ход, рассмотрены все возможные ответы второго игрока, но неверно определены дальнейшие действия. 2. Правильно указан выигрывающий игрок и его первый ход, но описание выигрышной стратегии неполно и рассмотрены несколько (больше одного, но не все) вариантов ответов второго игрока.	1
Задание не выполнено или в представленном решении полностью отсутствует описание элементов выигрышной стратегии, и отсутствует анализ вариантов первого-второго ходов играющих (даже при наличии правильного указания выигрывающего игрока).	0

Мало решить задачу. Нужно еще правильным образом оформить ее решение и ответ.

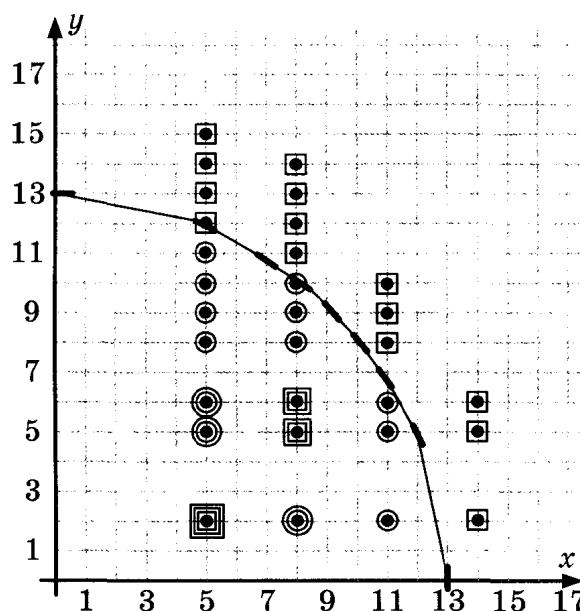
Вы уже прочитали, что все три балла можно получить только в том случае, если приведено доказательство правильности выбранного решения.

Начнем оформлять задачу именно с доказательства правильности решения.

Для этого опишем всю последовательность только что рассмотренных действий с теми рассуждениями, которые мы при этом делали.

Обратите внимание — мы рекомендуем вам не полениться и подробно расписать свои действия и рассуждения. Иначе эксперт может не понять ваше решение и не засчитает вам драгоценный балл за доказательство правильности.

### Пример оформления задания:



Нарисуем на координатной плоскости все возможные положения фишки (обозначены точками). Нарисуем дугу окружности (радиуса 13), обозначающую границу выигрыша. Обо-



значим  $\blacksquare$  проигрышные позиции (если игроку достался ход в этой позиции, значит его противник только что выиграл). Из них найдем выигрышные позиции, из которых существует ход, приводящий в  $\blacksquare$ . Их обведем кружком  $\odot$ . Из них найдем проигрышные позиции, из которых куда бы ни пошел, попадешь в  $\odot$ . Обозначим их  $\blacksquare$ . Из них найдем выигрышные из которых есть ход в  $\blacksquare$ . Обозначим их  $\odot$ . Из них получаем, что стартовая позиция — проигрышная (из нее куда ни пойдешь, противник оказывается в  $\odot$  — выигрышной позиции).

Мы рассмотрели все возможные ходы обоих игроков. На основании приведенных рассуждений делаем вывод, что выиграет игрок, делающий второй ход.

*Ответ:* выигрывает второй игрок.

Стратегия игры:

I ход I-го игрока	I ход II-го игрока (выигрышный)	Дальнейшие ходы
(8, 2)	(8, 5)	При любом последующем ходе I игрока второй игрок должен передвинуть фишку на $(x + 3, y)$ и тем самым выиграть
(5, 5)	(8, 5)	
(5, 6)	(8, 6)	

Хотя этот способ оформления ответа на задачу соответствует тому методу решения, которое мы предлагаем, предпочтительнее все же будет способ, предложенный в Демо-версии.

Это связано с тем, что у эксперта при проверке в образце задание будет оформлено именно так. И если вы приведете решение, полностью соответствующее этому образцу, у эксперта будет гораздо меньше сомнений, должен ли он ваш ответ засчитать как верное доказательство правильности. Он просто убедится, что вы слово в слово привели образцовый ответ и поставит вам ваши заслуженные 3 балла.

**Рекомендуемый образец оформления ответа:**

Выигрывает второй игрок.

Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры.

	1-й ход	2-й ход	3-й ход	4-й ход
Стартовая позиция	I-й игрок (все варианты хода)	II-й игрок (выигрышный ход)	I-й игрок (все варианты хода)	II-й игрок (выигрышный ход, один из вариантов)
5, 2	8, 2	<u>8, 5</u>	11, 5	14, 5
			8, 8	<u>11, 8</u>
			8, 9	<u>11, 9</u>
	5, 5	<u>8, 5</u>	11, 5	14, 5
			8, 8	<u>11, 8</u>
			8, 9	<u>11, 9</u>
	5, 6	<u>8, 6</u>	11, 6	14, 6
			8, 9	<u>11, 9</u>
			8, 10	<u>11, 10</u>

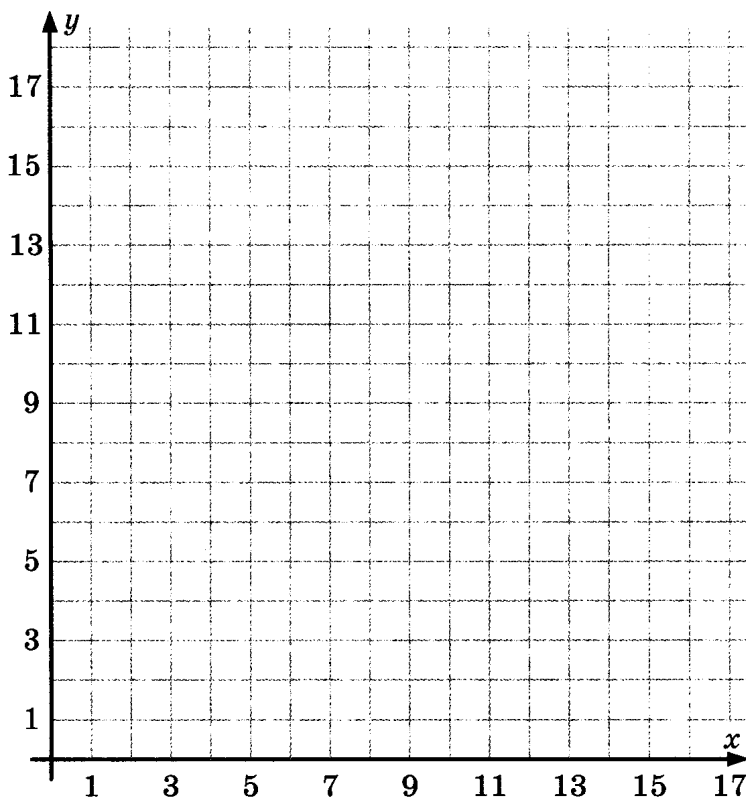
Таблица содержит все возможные варианты ходов первого игрока. Из нее видно, что при любом ходе первого игрока у второго имеется ход, приводящий к победе.

**Обратите внимание!** Мы рекомендуем выучить текст наизусть и привести именно эти фразы. В таблице обязательно подписать столбцы, указав для проигрывающего игрока «все варианты хода», а для выигрывающего игрока «выигрышный ход».

После таблицы ОБЯЗАТЕЛЬНО приведите эти две фразы, которые позволяют считать, что вы действительно доказали правильность решения.

## Практическая часть

**7.2.** Два игрока играют в следующую игру. На координатной плоскости в точке  $(1, 3)$  стоит фишка. Игроки ходят по очереди. Ход состоит в том, что игрок перемещает фишку из точки с координатами  $(x, y)$  в одну из трех точек: в точку с координатами  $(x + 4, y)$ , в точку  $(x, y + 3)$ , или в точку  $(x, y + 4)$ . Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами  $(0, 0)$  не меньше 13 единиц. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Какими должны быть ходы выигрывающего игрока? Ответ обоснуйте.

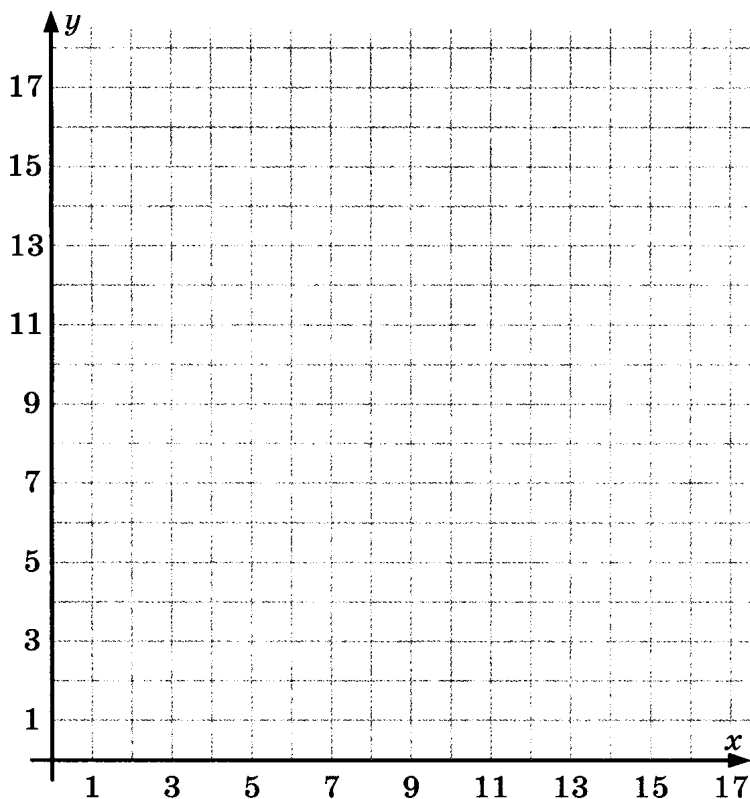


Ответ: \_\_\_\_\_.

		Дальнейшие ходы

**7.3.** Два игрока играют в следующую игру. На координатной плоскости в точке  $(1, 2)$  стоит фишка. Игроки ходят по очереди. Ход состоит в том, что игрок перемещает фишку из

точки с координатами  $(x, y)$  в одну из трех точек: в точку с координатами  $(x + 3, y)$ , в точку  $(x, y + 3)$  или в точку  $(x + 3, y + 3)$ . Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами  $(0, 0)$  не меньше 13 единиц. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Какими должны быть ходы выигрывающего игрока? Ответ обоснуйте.



Ответ: \_\_\_\_\_.

		Дальнейшие ходы

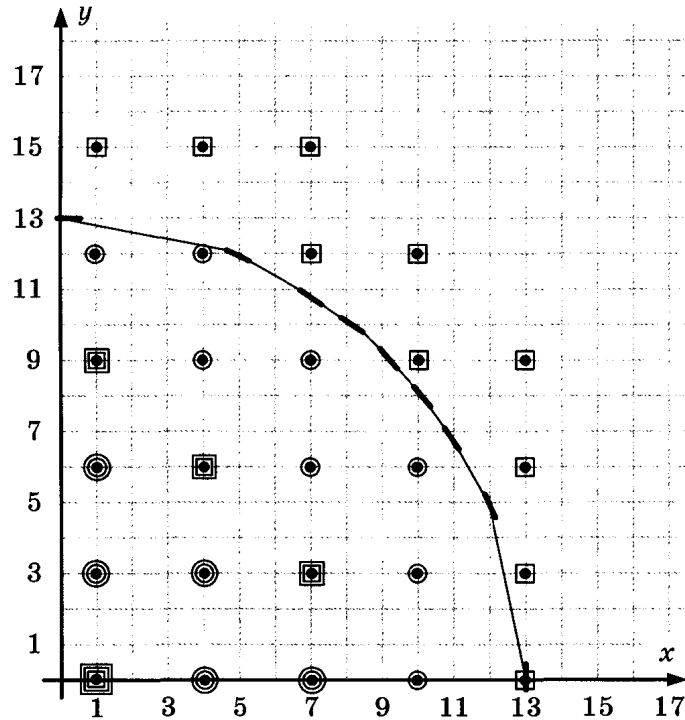
**7.4.** Два игрока играют в следующую игру. На координатной плоскости в точке  $(1, 0)$  стоит фишка. Игроки ходят по очереди. Ход состоит в том, что игрок перемещает фишку из точки с координатами  $(x, y)$  в одну из трех точек: в точку с координатами  $(x + 3, y)$ , в точку  $(x, y + 3)$  или в точку  $(x + 3, y + 3)$ . Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами  $(0, 0)$  не меньше 13 единиц. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Какими должны быть ходы выигрывающего игрока? Ответ обоснуйте.

*Рекомендация.* Рассмотренный ранее способ решения задачи работает по принципу «снизу вверх» — отталкиваясь от позиций, про которые известно, что они проигрышные, мы определяем выигрышность / проигрышность все более предыдущих позиций, пока не вернемся к начальной позиции.

Существует второй способ решения задачи — последовательно рассматривать все возможные ходы от стартовой позиции, пока не придем к выигрышу одного из игроков. Затем можно будет выбрать те ходы, которые приводят к выигрышу.

Но решение этой задачи «в лоб» будет трудоемким. Ведь от начальной позиции до финала игроки должны сделать не менее четырех ходов. В каждой позиции этих ходов можно пойти одним из трех способов. Таким образом, количество комбинаций ходов, которые нужно будет выписать, не меньше чем  $3^4$  (81). Это достаточно трудоемко. Рассмотренный ранее графический метод нагляден, перебирает все возможные варианты и позволяет быстро получить результат.

*Решение:*



*Ответ:* Выигрывает второй игрок.

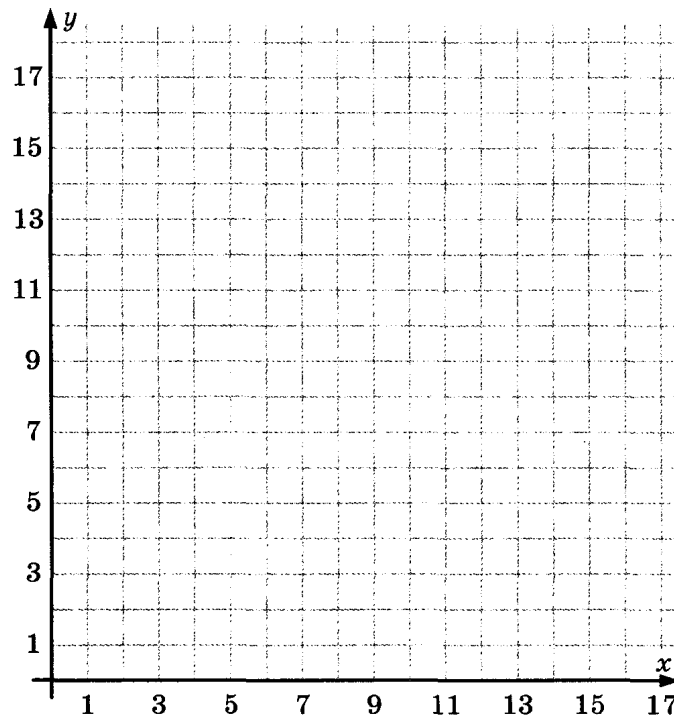
Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры.

	1-й ход	2-й ход	3-й ход	4-й ход
Стартовая позиция	I-й игрок (все варианты хода)	II-й игрок (выигрышный ход)	I-й игрок (все варианты хода)	II-й игрок (выигрышный ход, один из вариантов)
1, 0	4, 0	<u>7, 3</u>	10, 3	<u>13, 6</u>
			10, 6	<u>13, 6</u>
			7, 6	<u>10, 9</u>
	4, 3	<u>7, 3</u>	10, 3	<u>13, 6</u>
			10, 6	<u>13, 6</u>
			7, 6	<u>10, 9</u>
	1, 3	<u>4, 6</u>	7, 6	<u>10, 9</u>
			7, 9	<u>10, 9</u>
			4, 9	<u>7, 12</u>

Таблица содержит все возможные варианты ходов первого игрока. Из нее видно, что при любом ходе первого игрока у второго имеется ход, приводящий к победе.

*Замечание.* Очень важно написать текст, который приведен до таблицы и после таблицы. Повторяем, мы рекомендуем выучить эти «волшебные» фразы и записать их в ответ именно в таком виде. Это нужно, чтобы вам засчитали полностью ваши 3 балла за задачу. Формулируя ответ ровно в таких же выражениях, как приведено в Демо-версии ЕГЭ (читай «в образце»), вы страхуете себя от возможных недоразумений при оценивании.

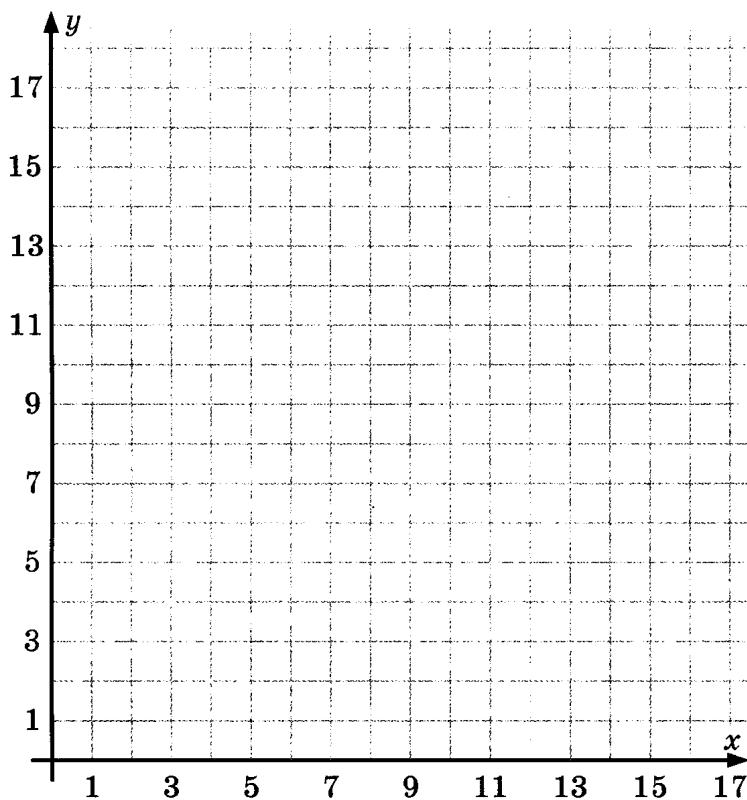
**7.5.** Два игрока играют в следующую игру. На координатной плоскости в точке  $(1, 0)$  стоит фишка. Игроки ходят по очереди. Ход состоит в том, что игрок перемещает фишку из точки с координатами  $(x, y)$  в одну из трех точек: в точку с координатами  $(x + 3, y)$ , в точку  $(x, y + 3)$  или в точку  $(x, y + 4)$ . Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами  $(0, 0)$  не меньше 13 единиц. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Какими должны быть ходы выигрывающего игрока? Ответ обоснуйте.



Ответ: \_\_\_\_\_.

	1-й ход	2-й ход	3-й ход	4-й ход
Стартовая позиция				

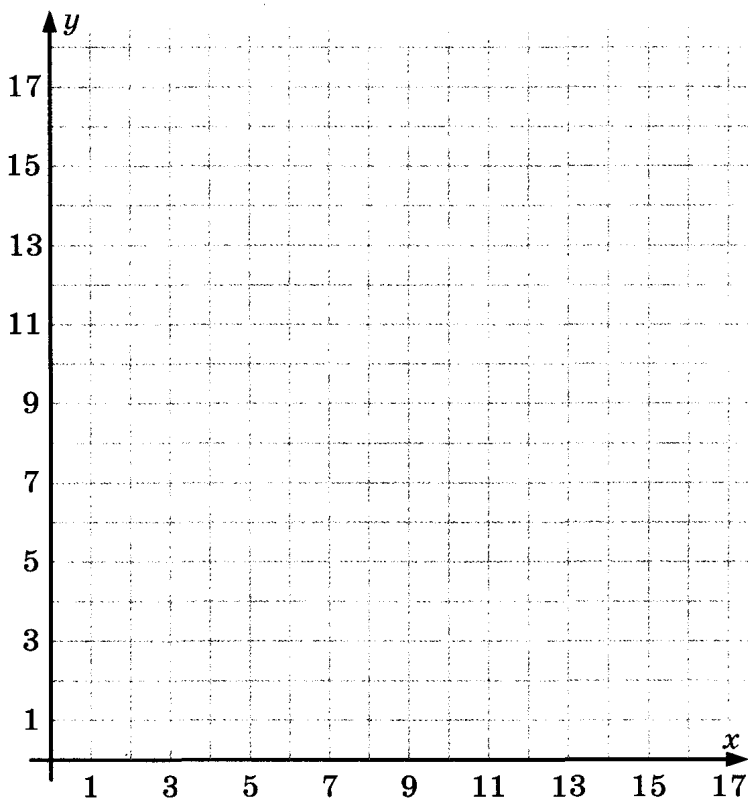
**7.6.** Два игрока играют в следующую игру. На координатной плоскости в точке  $(1, 2)$  стоит фишка. Игроки ходят по очереди. Ход состоит в том, что игрок перемещает фишку из точки с координатами  $(x, y)$  в одну из трех точек: в точку с координатами  $(x + 3, y)$ , в точку  $(x, y + 3)$  или в точку  $(x, y + 4)$ . Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами  $(0, 0)$  больше 13 единиц. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Какими должны быть ходы выигрывающего игрока? Ответ обоснуйте.



Ответ: \_\_\_\_\_.

	1-й ход	2-й ход	3-й ход	4-й ход
Стартовая позиция				

**7.7.** Два игрока играют в следующую игру. На координатной плоскости в точке  $(1, 0)$  стоит фишка. Игроки ходят по очереди. Ход состоит в том, что игрок перемещает фишку из точки с координатами  $(x, y)$  в одну из трех точек: в точку с координатами  $(x + 4, y)$ , в точку  $(x, y + 4)$  или в точку  $(x + 4, y + 4)$ . Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами  $(0, 0)$  не меньше 13 единиц. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Какими должны быть ходы выигрывающего игрока? Ответ обоснуйте.



Ответ: \_\_\_\_\_.

	1-й ход	2-й ход	3-й ход	4-й ход
Стартовая позиция				

## 8. Файловая система компьютера

### Основные понятия

Для удобства использования данные на компьютере хранятся в виде файлов. *Файлом* называется поименованная совокупность данных, хранящихся на внутреннем (жесткий диск, флэш-диск) или внешнем (CD-диск, DVD-диск, дискета, флэш-карта, карта памяти, магнитная лента и др.) носителе информации.

Имя файла обычно состоит из собственно имени и так называемого *расширения*, служащего для определения типа данных, хранящихся в файле (типа файла). Имя файла и его расширение разделяются точкой.

Кроме имени, файл обычно имеет *атрибуты* (дата создания, дата изменения, пользователь, права доступа и т.д.).

Файловая система позволяет организовывать файлы в иерархические древовидные *каталоги*. Каталог самого верхнего уровня называется *корневым каталогом*. Во многих операционных системах (в том числе семейства Windows) корневой каталог обозначается именем логического источника данных — латинской буквой с двоеточием. Буквы А и В обозначают дисководы для дискет, С — жесткий диск компьютера, остальные буквы — внутренние носители данных или внешние, подключенные к компьютеру, например по интерфейсу USB или по сети.

Остальные каталоги делятся на *системные* и *пользовательские*. Системные каталоги имеют служебные имена, назначенные операционной системой при установке. Имена пользовательских каталогов задаются пользователем по его усмотрению.

Пользователь может передвигаться по каталогам с помощью *программ навигации* (Проводник, FarManager и др.) или с помощью специальных команд операционной системы. Тот каталог, в котором пользователь находится в настоящий момент, называется *текущим каталогом*.

Последовательность имен каталогов, проходимых от некоторого каталога до нужного файла, называется путем к этому файлу. Имена каталогов в этой последовательности принято разделять знаком «\».

Различают абсолютное (полное) и относительное имя файла. *Абсолютное имя* складывается из пути к файлу от корневого каталога и имени файла. *Относительное имя* — из пути к файлу из текущего каталога и имени файла. Если файл находится в текущем каталоге, то его относительное имя есть просто имя файла.

*Пример.* Пусть имя файла E:\SCHOOL\USER\DOC\user.txt

Это полное (абсолютное) имя файла, поскольку оно начинается с имени корневого каталога E:. Путь к файлу — E:\SCHOOL\USER\DOC\. Само имя файла — user.txt, где txt — расширение, указывающее на то, что файл должен содержать текстовые данные.

Для групповых операций с файлами (поиск, удаление, перемещение) используются *маски* имен файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которых также могут встречаться следующие символы:

символ «?» (вопросительный знак) означает ровно один произвольный символ;

символ «·» (звездочка) означает любую последовательность символов произвольной длины, в том числе «·» может задавать и пустую последовательность.



**Примеры.**

Маска для выделения всех файлов в текущем каталоге: \*.\*

Маска для выделения файлов с расширением .txt: \*.txt

Маска для выделения файлов, имена которых начинаются с буквы s и имеют односимвольное расширение: s\*.\*.

## Практическая часть

**8.1.** Выпишите имя корневого каталога, путь к файлу, его расширение:

а) В:\ТХТ\DOC\MY.CPP

Ответ: \_\_\_\_\_.

б) Y:\Y\Z\Z.Y

Ответ: \_\_\_\_\_.

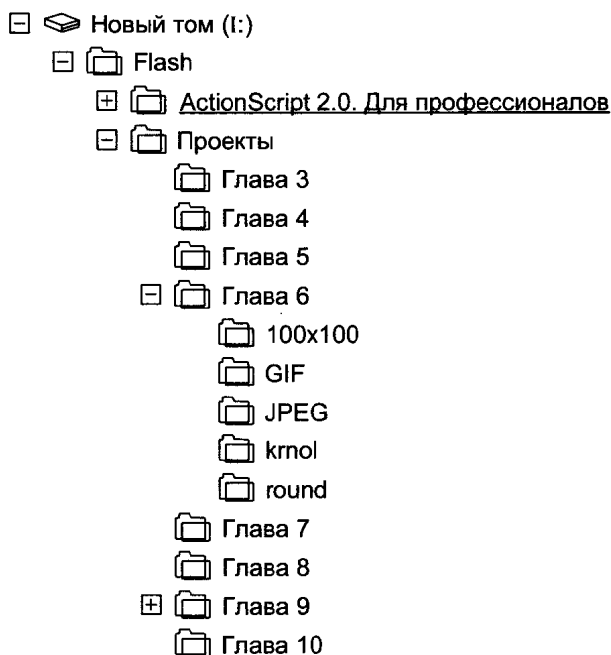
в) C:\Мои документы\фото\2007\портрет.jpg

Ответ: \_\_\_\_\_.

г) C:\текст.doc

Ответ: \_\_\_\_\_.

**8.2.** На рисунке приведен фрагмент файловой системы компьютера.



а) Запишите полный путь к файлу 333.gif, находящемуся в каталоге GIF.

Ответ: \_\_\_\_\_.

б) Запишите путь из каталога Проекты к файлу 333.gif, находящемуся в каталоге GIF.

Ответ: \_\_\_\_\_.

в) Запишите путь из каталога Глава 6 к файлу 333.gif, находящемуся в каталоге GIF.

Ответ: \_\_\_\_\_.

г) Запишите полный путь к файлу test.swf, находящемуся в каталоге Проекты.

Ответ: \_\_\_\_\_.

*Пример.* Какие файлы будут выбраны из текущего каталога по маске d · с.?р

### Решение

По указанной маске будут выбраны из текущего каталога все файлы, имена которых начнутся на d и заканчиваются на с, и одновременно имеют двухбуквенное расширение, заканчивающееся на р.

*Пример.* Составьте маску, по которой из текущего каталога будут выбраны все файлы, имена которых начинаются на букву «F» с расширением, состоящим ровно из трех символов.

### Решение

Составим отдельно маски для расширения и собственно имени файла.

Поскольку указано, что расширение состоит ровно из трех символов, для него необходимо использовать маску «???». Символ «·» в данном случае использовать нельзя, так как он не накладывает ограничений на длину подставляемой последовательности символов. Напротив, поскольку для имени файла ограничения на длину не предусмотрены — маска для имени будет «F·».

Объединяя маски для имени и расширения и не забывая про разделитель имени и расширения — точку, получает ответ: F·.???

*Пример.* Определите, какое из указанных имен файлов удовлетворяет маске

?cr · m.d?c

1) crim.doc

2) acrim.doc

3) ocrm.dtc

4) ocrm.dc

### Решение

Проверим последовательно, какие имена файлов удовлетворяют указанной маске:

1) crim.doc — не подходит, т.к. символу «?» в имени файла ничего не соответствует, а это недопустимо;

2) acrim.doc — не подходит, поскольку в имени файла есть лишний символ «s», не соответствующий маске;

3) ocrm.dtc — подходит, символу «?» маски в имени файла соответствует ровно одна буква «o», а в расширении «t». Символу «·» маски в имени файла соответствует пустая последовательность символов, что допускается.

Для полноты решения проверим четвертый вариант.

4) ocrm.dc — не подходит, так как символу «?» в расширении файла ничего не соответствует, что недопустимо.

Ответ: 3.

**8.3.** В текущем каталоге имеются файлы:

- 1) zzz.cpp            2) zzz.cpprc            3) z.c            4) 1.cpp            5) 123.pc

а) Какие файлы будут выбраны из текущего каталога по маске `*.cpp`

Ответ: \_\_\_\_\_.

б) Какие файлы будут выбраны из текущего каталога по маске `*.c`

Ответ: \_\_\_\_\_.

в) Какие файлы будут выбраны из текущего каталога по маске `*z*.c?`

Ответ: \_\_\_\_\_.

г) Какие файлы будут выбраны из текущего каталога по маске `*.*.c`

Ответ: \_\_\_\_\_.

**8.4.**

а) Составьте маску, по которой из текущего каталога будут выбраны все файлы с расширением `.doc`, в именах которых встречается буква «я».

Ответ: \_\_\_\_\_.

б) Составьте маску, по которой из текущего каталога будут выбраны все файлы с расширением `.doc`, имена которых начинаются и заканчиваются на букву «я».

Ответ: \_\_\_\_\_.

в) Составьте маску, по которой из текущего каталога будут выбраны все файлы, имена которых начинаются на букву «z» с расширением, состоящим ровно из двух символов.

Ответ: \_\_\_\_\_.

г) Составьте маску, по которой из текущего каталога будут выбраны все файлы с расширением `.txt`, имена которых включают идущие подряд цифры 2009.

Ответ: \_\_\_\_\_.

### Задания формата ЕГЭ

**8.5.**

а) В некотором каталоге хранился файл Лермонтов.doc. После того как в этом каталоге создали подкаталог и переместили в созданный подкаталог файл Лермонтов.doc, полное имя файла стало `F:\Россия\Поэты\XIX\Лермонтов.doc`. Каково имя вновь созданного каталога?

- 1) Поэты            2) F:\Россия            3) XIX            4) Россия

б) В некотором каталоге хранился файл Задачи.doc. После того как в этом каталоге создали подкаталог и переместили в созданный подкаталог файл Задачи.doc, полное имя файла стало `G:\Математика\9\Алгебра\Задачи.doc`. Каково имя вновь созданного каталога?

- 1) Алгебра            2) Математика            3) 9            4) G:\Математика

- в) Перемещаясь из одного каталога в другой, пользователь последовательно посетил каталоги Задачи, 9, Алгебра, C:\, Геометрия, 11. Каково полное имя каталога, в котором оказался пользователь?

*Примечание:* при каждом перемещении пользователь либо спускался в каталог на уровень ниже, либо поднимался на уровень выше.

- 1) 11
- 2) C:\ Алгебра\9\Задачи
- 3) C:\Геометрия\11
- 4) Геометрия\11

- г) Перемещаясь из одного каталога в другой, пользователь последовательно посетил каталоги Задачи, Геометрия, В:\, Алгебра, Уравнения, Задачи. Каково полное имя каталога, в котором оказался пользователь?

*Примечание:* при каждом перемещении пользователь либо спускался в каталог на уровень ниже, либо поднимался на уровень выше.

- 1) Алгебра\Уравнения\Задачи
- 2) В:\ Алгебра\Уравнения\Задачи
- 3) В:\Геометрия\Задачи
- 4) Геометрия\Задачи

## 8.6.

- а) Определите, какое из указанных имен файлов удовлетворяет маске: hel?lo.c??
- 1) hello.cpp
  - 2) helolo.c
  - 3) helolo.cnt
  - 4) helsslo.cpp
- б) Определите, какое из указанных имен файлов удовлетворяет маске: he · o.c ·
- 1) helol.cpp
  - 2) he.oc
  - 3) helolo.pcc
  - 4) herro.c
- в) Определите, какое из указанных имен файлов удовлетворяет маске: ?com · .d?c
- 1) com.doc
  - 2) dcoma.dc
  - 3) dcom.dotc
  - 4) dcoma.djc
- г) Определите, какое из указанных имен файлов удовлетворяет маске: · com?.d · c
- 1) com.doc
  - 2) dcoma.dc
  - 3) dcom.dotc
  - 4) dcoma.cdc

## 8.7.

- а) Некий файл хранится по следующему адресу:

D:\VASYA\PICTURES\FACE.JPG.

Укажите заведомо неверное утверждение.

- 1) Из папки VASYA путь к файлу можно указать так: PICTURES\FACE.JPG
- 2) Этот файл содержит изображение черного треугольника на синем фоне
- 3) Полное имя этого файла: FACE.JPG
- 4) Этот файл хранится не в корне диска D:

- б) Некий файл хранится по следующему адресу:  
A:\MARUSYA\HOMEWORK\WINTER.DOC.  
Укажите заведомо неверное утверждение.
- 1) Полное имя этого файла: A:\MARUSYA\HOMEWORK\WINTER.DOC
  - 2) Из папки MARUSYA путь к файлу можно указать так: HOMEWORK\WINTER.DOC
  - 3) Этот файл хранится в корне диска A:
  - 4) Этот файл содержит таблицу расписания поездов
- в) Определите, какое из указанных имен файлов удовлетворяет маске: ??prive · t.p? ·
- 1) Aprivet.pas
  - 2) z1privet.past
  - 3) privert.p3
  - 4) doprivest.p
- г) Определите, какое из указанных имен файлов удовлетворяет маске: ?info · rm.d · ?
- 1) 12info7rm.deep
  - 2) 11inform.doc
  - 3) 4info5rm.d
  - 4) 7inform.dp

## 9. Электронные таблицы

### Основные понятия

Так как одним из самых удобных способов представления информации является таблица, обработке таблиц посвящена отдельная категория программных продуктов — *электронные таблицы*. Электронная таблица состоит из строк и столбцов, на пересечении которых находятся ячейки. Каждая ячейка может хранить один из нескольких видов информации: текстовую, числовую, дату/время или формулу.

В электронных таблицах принято обозначать столбцы буквами латинского алфавита, а строки — числами. Таким образом, для обозначения ячейки нужно указать номер столбца и номер строки этой ячейки. Например, D7. Это называется *адресом ячейки*.

Для обработки информации, содержащейся в ячейках, используют формулы.

Формулы в электронных таблицах начинаются с символа равно (=). В качестве аргументов в формулах можно использовать числа, элементы текста и адреса других ячеек. Эти аргументы можно связывать различными алгебраическими операциями, операциями отношения и другими.

Для удобства в электронных таблицах введены различные функции. Самые широкоиспользуемые — вычисление суммы (СУММ) и вычисление среднего арифметического значения (СРЗНАЧ).

Для перечисления адресов ячеек, над которыми должна быть осуществлена функция, используют знак «;» (точка с запятой). Для указания прямоугольного блока ячеек используется обозначение *диапазона*: адрес верхней левой и правой нижней ячейки блока, с символом «:» (двоеточие) между ними.

При копировании формул из одной ячейки в другую действует *автоиндексация*. Это значит, что адреса ячеек при копировании будут изменены в соответствии с расстоянием, на которое осуществляется копирование. То есть если копируется формула из ячейки на 3 столбца вправо и на 2 строки вверх, то во всех адресах, которые присутствуют в формуле, номера столбцов будут «увеличены» на 3, а номера строк уменьшены на 2.

Чтобы такого эффекта не происходило, перед теми элементами адреса (номером столбца и/или номером строки), которые не должны измениться при копировании, нужно ставить знак доллара (\$). Это называется *абсолютная адресация* ячейки. В противоположность ей, отсутствие знака доллара в адресе называется *относительной адресацией*.

### Практическая часть

**9.1.** Укажите одно или несколько правильных утверждений. Буквы, соответствующие правильным утверждениям, обведите, остальные — зачеркните.

Для электронных таблиц справедливо:

- а) электронная таблица состоит из строк и столбцов, на пересечениях которых находятся ячейки;
- б) строки и столбцы электронной таблицы пронумерованы. Строки обычно нумеруются последовательными натуральными числами. Столбцы обычно нумеруются буквами латинского алфавита, от А и дальше;
- в) каждая ячейка электронной таблицы имеет уникальный адрес, состоящий из номера столбца и номера строки (например, А1, С14, ВD23).

**9.2.** Укажите один или несколько правильных ответов. Буквы, соответствующие правильным ответам, обведите, остальные — зачеркните.

Какая информация может храниться в ячейке электронной таблицы?

- |                            |                      |
|----------------------------|----------------------|
| а) Целое число             | д) Время/дата        |
| б) Вещественное число      | е) Звук              |
| в) Графическое изображение | ж) Вложенная таблица |
| г) Строка символов         | з) Формула           |

**9.3.** Для расчета значения ячейки на основании значения другой ячейки (ячеек) в электронных таблицах используются формулы. Укажите выражения, которые электронная таблица будет считать формулами и вычислит. Буквы, соответствующие правильным ответам обведите, остальные — зачеркните:

- |                |                  |
|----------------|------------------|
| а) $A1 + 3$    | е) $A1 = A3$     |
| б) $= 5$       | ж) $A1 + A3$     |
| в) $= A1 - 2$  | з) $= A1 = A2$   |
| г) $A1 < A3$   | и) $= A(12 + 4)$ |
| д) $= 53 - 10$ | к) $= A3 > A2$   |

**9.4.** Для осуществления операций над константами и значениями ячеек в электронной таблице используются операторы. Сопоставьте каждому оператору (слева) действие, которое этот оператор осуществляет (справа). Впишите рядом с буквой операции число, соответствующее смыслу этой операции:

- |             |  |
|-------------|--|
| а) $>$      | 1) возведение в степень (первый аргумент в степени второго)    |
| б) $-$      | 2) сравнение аргументов (если первый меньше или равен второму) |
| в) $\cdot$  | 3) сравнение аргументов (если первый больше второго)           |
| г) $=$      | 4) сложение аргументов   |
| д) $\wedge$ | 5) умножение аргументов  |
| е) $\&$     | 6) деление первого аргумента на второй                         |
| ж) $+$      | 7) сравнение аргументов (если первый больше или равен второму) |
| з) $>=$     | 8) сравнение аргументов (если аргументы не равны)              |
| и) $/$      | 9) вычитание второго аргумента из первого                      |
| к) $<>$     | 10) склеивание двух текстовых аргументов                       |
| л) $<=$     | 11) сравнение аргументов (если аргументы равны)                |

**9.5.** Для осуществления более сложных действий над аргументами в электронных таблицах используются функции. Укажите, какие из приведенных утверждений верны при записи функций (в электронной таблице). Буквы, соответствующие правильным утверждениям, обведите, остальные — зачеркните:

- а) функции используются в формулах (то есть в выражениях, начинающихся со знака  $=$  (равно));
- б) каждая функция имеет свое уникальное имя;
- в) имя функции должно быть записано обязательно русскими буквами;
- г) после указания имени функции нужно обязательно написать аргументы в круглых скобках;

- д) даже если функция не имеет аргументов, все равно нужно написать пустые круглые скобки;
- е) аргументы функции отделяются друг от друга символом «;» (точка с запятой);
- ж) для указания в качестве аргумента целой области соседних ячеек используется *диапазон* ячеек;
- з) прямоугольный диапазон ячеек записывается указанием адреса левой верхней и правой нижней ячейки прямоугольного диапазона через символ «:» (двоеточие). Например, А1:В3;
- и) для указания в качестве диапазона целиком строки, столбца или нескольких используется только один (нужный) индекс. Например: А:А (весь столбец А); В:D (столбцы от В до D); 6:6 (вся строка 6); 4:12 (строки начиная с 4-й по 12-ю).

**9.6.** В списке наиболее часто используемых функций установите соответствие между именем функции и возвращаемым ею значением. Впишите рядом с буквой функции число, соответствующее смыслу этой функции:

- |           |   |
|-----------|---|
| а) СУММ   | 1) наибольшее значение  |
| б) СРЗНАЧ | 2) если указанное условие ИСТИНА, то первый, второй аргумент, иначе — третий аргумент |
| в) МАКС   | 3) наименьшее значение  |
| г) МИН    | 4) сумма  |
| д) ЕСЛИ   | 5) среднее арифметическое (сумма, деленная на количество)                             |

**9.7.** Во всех ячейках таблицы записано число 1. Вычислите результат функции СУММ для каждого списка аргументов:

- а) СУММ(А2) = \_\_\_\_\_
- б) СУММ(А2:А4) = \_\_\_\_\_
- в) СУММ(С3:F3) = \_\_\_\_\_
- г) СУММ(В2:D5) = \_\_\_\_\_
- д) СУММ(А3:А7;В4:С5) = \_\_\_\_\_
- е) СУММ(А1;В3;А4;В8) = \_\_\_\_\_
- ж) СУММ(А5;В3;Е6;F12) = \_\_\_\_\_
- з) СУММ(3:3) = \_\_\_\_\_

**9.8.** Для данной электронной таблицы вычислите результат функций:

	А	В
1	3	5
2	2	9
3	5	7
4	0	3

- а) МИН(А1:В4) \_\_\_\_\_
- б) МАКС(А3:В4) \_\_\_\_\_
- в) СУММ(А1:А4) \_\_\_\_\_
- г) СРЗНАЧ(А1:А4) \_\_\_\_\_
- д) СРЗНАЧ(В1:В4) \_\_\_\_\_



**9.9.** В электронной таблице значение формулы =СУММ(B1:B2) равно 5. Чему равно значение ячейки B3, если значение формулы =СРЗНАЧ(B1:B3) равно 3?

- 1) 8                                      2) 2                                      3) 3                                      4) 4

*Рекомендация.* Нарисуйте на черновике (или представьте) упоминаемые в задаче ячейки. Вспомните, что символ «:» (двоеточие) между адресами ячеек — это не перечисление, а указание на диапазон. Вспомните, что СРЗНАЧ — это не «сложить и поделить пополам», а сумма указанных ячеек, деленная на их количество. А количество в данном случае вовсе не две. Догадайтесь, что если мы знаем количество и среднее значение, то можно отсюда вычислить сумму. Осталось догадаться, что если известна сумма N чисел и сумма тех же чисел, но без одного из них, то вычитанием легко вычислить оставшееся число.

**9.10.** В электронной таблице значение формулы =СРЗНАЧ(A1:C1) равно 5. Чему равно значение ячейки D1, если значение формулы =СУММ(A1:D1) равно 7?

- 1) 2                                      2) -8                                      3) 8                                      4) -3

**9.11.** В электронной таблице значение формулы =СРЗНАЧ(B1:D1) равно 4. Чему равно значение ячейки A1, если значение формулы =СУММ(A1:D1) равно 9?

- 1) -3                                      2) 5                                      3) 1                                      4) 3

**9.12.** В электронной таблице значение формулы =СРЗНАЧ(A1:B4) равно 3. Чему равно значение ячейки A4, если значение формулы =СУММ(A1:B3) равно 30, а в ячейке B4 лежит 5?

Ответ: \_\_\_\_\_.

*Напоминание.* Обычная запись адреса ячейки в формуле (например, C5) соответствует относительной адресации ячейки. Это значит, что при копировании такой формулы в другую ячейку таблицы произойдет автоматическая индексация. То есть обе части адреса ячейки (номер строки и номер столбца, они называются *индексами*) изменятся на такое значение, на сколько по вертикали и по горизонтали ячейка была скопирована относительно своего исходного положения. Например, формула «=C5» находится в ячейке A4. Эта формула копируется в ячейку B2 — на 2 строки выше и на один столбец правее. То есть новый адрес формулы уменьшается на 2 по вертикали (номер строки) и увеличивается на один по горизонтали (номер столбца). Значит, относительные адреса ячеек, используемых в формуле, изменяются на эти же величины: номер строки уменьшается на 2, номер столбца увеличивается на 1. Адрес C5 превратится в адрес D3 (C + 1, 5 - 2).

**9.13.** Дан фрагмент электронной таблицы:

	A	B	C
1	1	3	=A1+B1+2
2	2	4	

Чему станет равным значение ячейки C2, если в нее скопировать формулу из ячейки C1?

Ответ: \_\_\_\_\_.

**Напоминание.** Если при копировании формулы не требуется автоматическая индексация, используют абсолютную адресацию ячеек. Для этого перед той частью адреса (перед тем индексом), которая не должна измениться при копировании, нужно поставить знак доллара (\$). Если знак абсолютной адресации стоит перед обеими частями адреса (обеими индексами, например, \$C\$5), этот адрес не изменится при копировании в любое место. Если знак «\$» только перед одной частью адреса — будет изменяться та часть адреса (тот индекс), перед которым знак доллара не стоит. Например, адрес \$B4 не будет меняться при копировании влево-вправо, но будет меняться (как обычно) при копировании вверх-вниз.

**9.14.** Дан фрагмент электронной таблицы:

	A	B	C
1	10	20	=A1+B\$1+2
2	30	40	

Чему станет равно значение ячейки C2, если в нее скопировать формулу из ячейки C1?

Ответ: \_\_\_\_\_.

**9.15.** При работе с электронной таблицей в ячейку C3 записана формула: =\$B3+C\$2. Какой вид приобретет формула, после того как ячейку C3 скопируют в ячейку D2?

Ответ: \_\_\_\_\_.

**9.16.** При работе с электронной таблицей в ячейку E5 записана формула: =\$C3-B\$3. Какой вид приобретет формула, после того как ячейку E5 скопируют в ячейку D3?

Ответ: \_\_\_\_\_.

**9.17.** При работе с электронной таблицей в ячейку B4 записана формула: =\$A\$4+C3. Какой вид приобретет формула, после того как ячейку B4 скопируют в ячейку C2?

Ответ: \_\_\_\_\_.

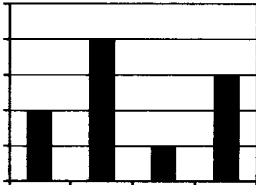
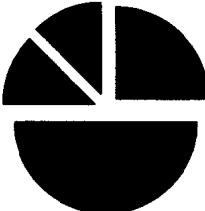
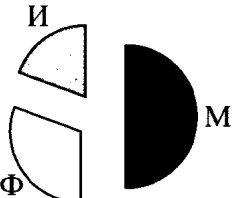
**9.18.** В ячейке B1 записана формула =2 \* \$A1. Какой вид приобретет формула, после того как ячейку B1 скопируют в ячейку C2?

Ответ: \_\_\_\_\_.

**Напоминание.** Для наглядного представления табличной информации часто используют диаграммы. В зависимости от необходимого визуального эффекта выбирают тип диаграммы. В основном это *Линейчатая* и *Гистограмма* (горизонтальные или вертикальные прямоугольники), *Кольцевая* и *Круговая*, а также *График* и *Точечная*.

Основное, что вы должны уметь делать, — понимать, что диаграммы строятся на основе данных, находящихся в таблице и что Кольцевая и Круговая диаграммы отображают соотношение этих данных к их общей сумме. А в остальных диаграммах отображаются абсолютные значения.

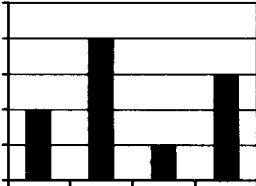

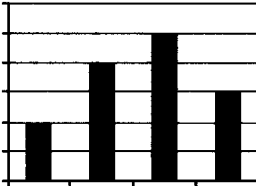

Примеры.

Диаграмма	Смысл диаграммы
	<p>Если принять, что единица измерения вдоль вертикальной оси равна одному, то диаграмма построена по данным «2, 4, 1, 3»</p>
	<p>Данные, по которым построена диаграмма, соотносятся между собой как 1:1:2:4.</p>
	<p>Значение М составляет половину от общей суммы М+И+Ф. Значение Ф чуть больше четверти от общей суммы, значение И — меньше четверти.</p>

9.19. Дан фрагмент электронной таблицы:

	A	B
1	=B1+1	1
2	=A1+2	2
3	=B2-1	
4	=A3	

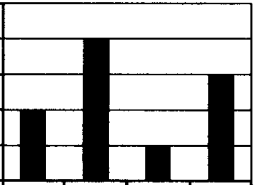

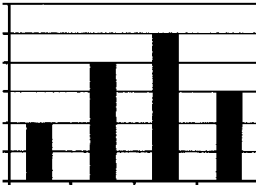

После выполнения вычислений была построена диаграмма по значениям диапазона ячеек A1:A4. Укажите получившуюся диаграмму.

1)  2)  3)  4) 

9.20. Дан фрагмент электронной таблицы:

	A	B	C	D
1	3		5	
2	=A1/3	=(A1+C1+1)/3	=C1-2	=(B2+C2)/6

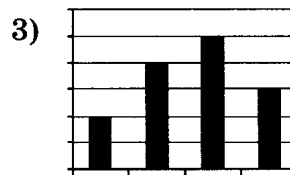
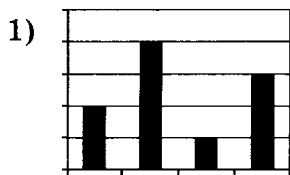
После выполнения вычислений была построена диаграмма по значениям диапазона ячеек A2:D2. Укажите получившуюся диаграмму.

1)  2)  3)  4) 

**9.21.** Дан фрагмент электронной таблицы:

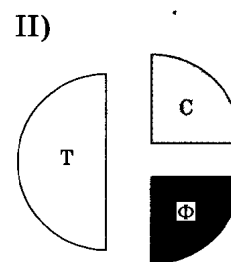
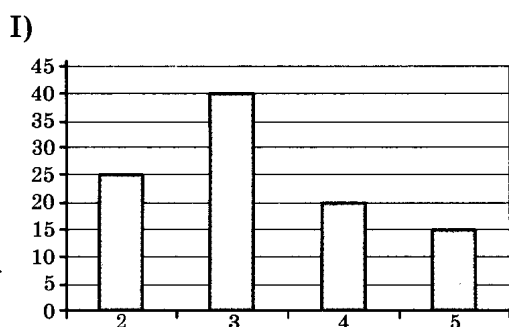
	A	B	C	D
1	3		5	
2	$= (A1+C1)/4$	$= C1-1$	$= A2/2$	$= B2-C2$

После выполнения вычислений была построена диаграмма по значениям диапазона ячеек A2:D2. Укажите получившуюся диаграмму.



**9.22.** В цехе трудятся рабочие трех специальностей — токари (Т), слесари (С) и фрезеровщики (Ф). Каждый рабочий имеет разряд не меньший второго и не больший пятого. На диаграмме I отражено количество рабочих с различными разрядами, а на диаграмме II — распределение рабочих по специальностям.

Каждый рабочий имеет только одну специальность и один разряд.



Имеются четыре утверждения:

- А) все рабочие третьего разряда могут быть токарями
- Б) все рабочие третьего разряда могут быть фрезеровщиками
- В) все слесари могут быть пятого разряда
- Г) все токари могут быть четвертого разряда

Какое из этих утверждений следует из анализа обеих диаграмм?

- 1) А
- 2) Б
- 3) В
- 4) Г

*Рекомендации.* По гистограмме (I диаграмма) посчитайте, сколько всего рабочих в цехе (второго разряда — 25, третьего — 40, четвертого — 20, пятого — 15; всего  $25 + 40 + 20 + 15 = 100$ ). Пользуясь соотношением, полученным из круговой диаграммы (II диаграмма), вычислите количество токарей, слесарей и фрезеровщиков (токарей половина, то есть  $100 / 2 = 50$ ; фрезеровщиков и слесарей по четверти, то есть по  $100 / 4 = 25$ ). Проверьте получившиеся соотношения:

- А) рабочих 3-го разряда 40, а токарей — 50. Могут ли быть все 40 рабочих 3-го разряда среди 50 токарей? Да, могут.
- Б) рабочих 3-го разряда 40, а фрезеровщиков — 25. Могут ли быть все 40 рабочих 3-го разряда 25 фрезеровщиками? Нет, не могут.

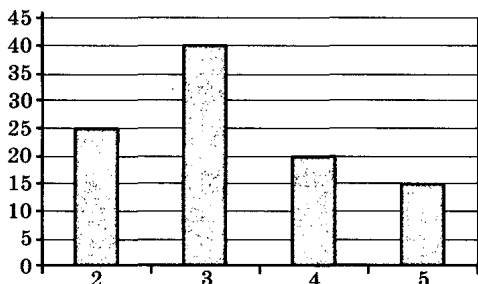
В) слесарей всего 25, а рабочих 5-го разряда — 15. Могут ли быть все 25 слесарей 15 рабочими 5-го разряда? Нет, не могут.

Г) токарей всего 50, а рабочих 4-го разряда — 20. Могут ли быть все 50 токарей 20 рабочими 4-го разряда? Нет, не могут.

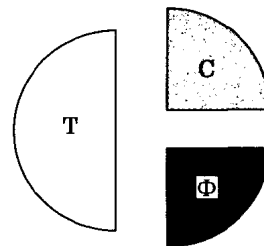
**9.23.** В цехе трудятся рабочие трех специальностей — токари (Т), слесари (С) и фрезеровщики (Ф). Каждый рабочий имеет разряд не меньший второго и не больший пятого. На диаграмме I отражено количество рабочих с различными разрядами, а на диаграмме II — распределение рабочих по специальностям.

Каждый рабочий имеет только одну специальность и один разряд.

I)



II)



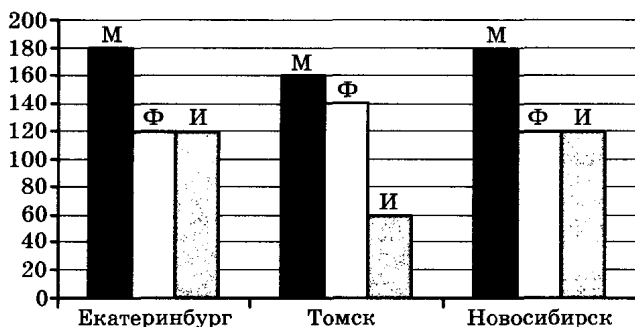
Имеются четыре утверждения:

- все рабочие второго разряда могут быть слесарями
- все фрезеровщики могут быть четвертого разряда
- среди рабочих третьего разряда может не быть ни одного токаря
- у всех токарей разряд может быть не меньше четвертого

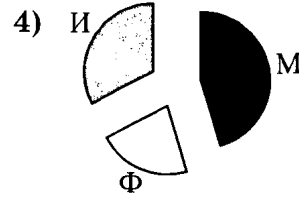
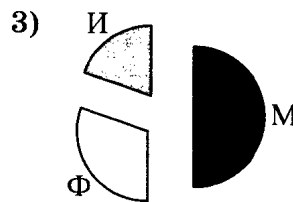
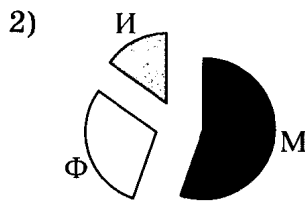
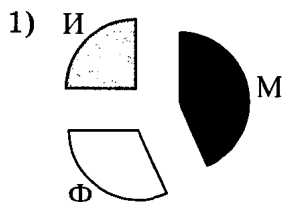
Какие из этих утверждений следуют из анализа обеих диаграмм?

Обведите буквы, соответствующие этим утверждениям.

**9.24.** На диаграмме показано количество призеров олимпиады по информатике (И), математике (М), физике (Ф) в трех городах России.



Какая из диаграмм правильно отражает соотношение общего числа призеров по каждому предмету для всех городов вместе?



### Пример решения

По первой диаграмме получим общее количество призеров по каждому предмету.

В Екатеринбурге призеров по математике 180, в Томске — 160, в Новосибирске — 180.

Итого призеров по математике  $180 + 180 + 160 = 520$ .

Аналогично получим, что призеров по физике всего  $120 + 140 + 120 = 380$ , а призеров по информатике —  $120 + 60 + 120 = 300$ .

Сложим полученные три суммы и получим, что общее число призеров по всем трем предметам равно  $520 + 380 + 300 = 1200$ .

Найдем отношение числа призеров по каждому предмету к общему числу призеров.

По математике —  $520/1200$  — чуть меньше половины.

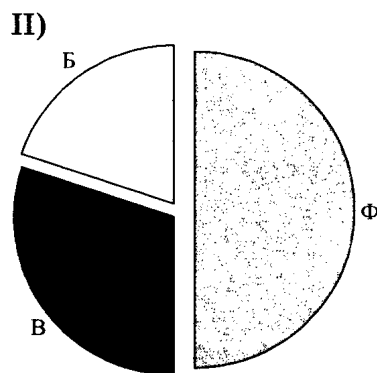
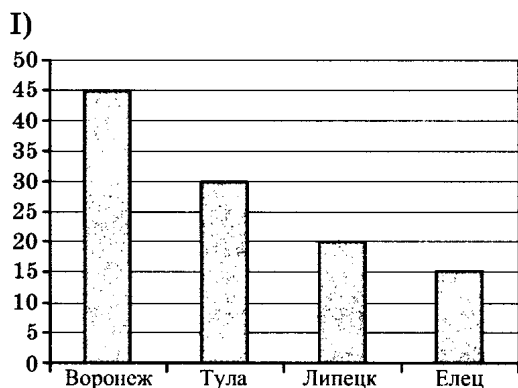
По физике —  $380/1200$  — примерно треть.

По информатике —  $300/1200$  — ровно четверть.

На приведенных диаграммах только первая отображает такое соотношение призеров.

*Рекомендация.* По первой диаграмме получите общее количество призеров по каждому предмету (сложите). Сложите получившиеся числа и посмотрите, как общее число призеров по каждому предмету соотносится с их общим количеством. Прикиньте, как каждое из получившихся отношений соотносится с половиной и с четвертью. Найдите получившееся соотношение среди приведенных диаграмм.

**9.25.** В спортивный пионерский лагерь приехали дети из четырех городов, занимающихся тремя видами спорта: футбол, волейбол и баскетбол. На диаграмме I показано, сколько детей приехало из какого города. На диаграмме II показано распределение детей по видам спорта.



Имеются четыре утверждения:

- а) все футболисты могут быть из Воронежа
- б) среди детей из Тулы может не быть ни одного футболиста
- в) среди волейболистов может не быть ни одного ребенка из Липецка
- г) среди детей из Тулы больше футболистов, чем баскетболистов

Какие из этих утверждений следуют из анализа обеих диаграмм?

Обведите буквы, соответствующие этим утверждениям.

# 10. Базы данных

## Основные понятия

Одной из самых удобных информационных моделей является база данных.

**База данных** — информационная модель, хранящая данные о группе объектов с одинаковым набором свойств. Как правило, база данных хранит информацию о нескольких группах объектов, связанных между собой.

В школьном курсе информатики изучают только **реляционные базы данных**. В них информация хранится в виде таблиц, связанных между собой отношениями подчинения.

Каждая строка таблицы описывает один объект. Этот объект может иметь различные свойства, информация о которых хранится в столбцах таблицы.

Данные в строках, таким образом, могут быть разных типов. Одна строка называется **записью**. Данные в каждом столбце описывают одно свойство объектов. Таким образом, данные в одном столбце имеют одинаковый тип. Столбец таблицы называется **полем**.

Для указания на конкретный объект (запись) таблицы принято вводить специальный, уникальный набор полей (или одно поле). Такой набор называется **ключом**, а поле — **ключевым**. Самый надежный и удобный способ — в каждой таблице хранить поле с уникальным кодом строки (записи).

Для связи между таблицами в подчиненной таблице используются специальные поля, в которых хранится информация о связанных записях главной таблицы. Обычно — ключ.

Для выбора из базы данных объектов с определенными свойствами используются операции сравнения, а также арифметические и логические операции.

Так как в школьном курсе информатики изучаются только реляционные базы данных, все сказанное ниже относится именно к реляционным базам данных.

## Практическая часть

**10.1.** Укажите одно или несколько правильных утверждений. Буквы, соответствующие правильным утверждениям, обведите, остальные — зачеркните.

Для баз данных справедливо:

- а) информация хранится в виде таблиц, связанных отношениями подчинения
- б) строки таблицы хранят информацию об объектах и называются записями
- в) столбцы таблицы хранят информацию о свойствах объектов
- г) информация в столбце всегда одного типа данных
- д) столбец таблицы называется полем

**Напоминание.** Для осуществления каких-либо действий над записями баз данных необходимо указывать, над какими именно записями нужно осуществить эти действия. Для отбора нужных записей используют условия. В условиях указываются имена полей, которые сравниваются друг с другом или с константами при помощи операций соотношения. Эти операции те же, что и в электронных таблицах.

**10.2.** Сопоставьте каждому сравнению (слева) условие, которое это сравнение проверяет (справа). Впишите рядом с буквой сравнения число, соответствующее смыслу этого сравнения:

- |           |  |
|-----------|--|
| а) $>$    | 1) если первый аргумент меньше или равен второму |
| б) $=$    | 2) если первый аргумент меньше второго           |
| в) $\geq$ | 3) если первый аргумент больше или равен второму |
| г) $<>$   | 4) если первый аргумент больше второго           |
| д) $\leq$ | 5) если аргументы не равны                       |
| е) $<$    | 6) если аргументы равны                          |

**10.3.** В таблице представлены результаты тестирования.

Фамилия	Математика	Физика
Иванов	76	54
Петров	38	74
Сидоров	40	43
Орлов	67	67
Пастухов	38	95
Николаев	23	37

Сколько записей в вышеприведенной таблице удовлетворяют условию «Математика  $>$  38»?

Ответ: \_\_\_\_\_.

**10.4.** Сколько записей в вышеприведенной таблице удовлетворяют условию «Физика  $\leq$  54»?

Ответ: \_\_\_\_\_.

**10.5.** Сколько записей в вышеприведенной таблице удовлетворяют условию «Физика  $\geq$  Математика»?

Ответ: \_\_\_\_\_.

**10.6.** Сколько записей в вышеприведенной таблице удовлетворяют условию «Физика  $<>$  Математика»?

Ответ: \_\_\_\_\_.

**10.7.** Сколько записей в вышеприведенной таблице удовлетворяют условию «Математика  $<>$  38»?

Ответ: \_\_\_\_\_.

**Напоминание.** Если требуется проверить сразу несколько условий, возникает необходимость объединить эти условия. Для этого, как вы помните из алгебры логики, используются логические операции.



**10.8.** Сопоставьте каждой логической операции (слева) ее смысл (справа). Впишите рядом с буквой операции число, соответствующее смыслу этой операции:

- |        |   |
|--------|---|
| а) И   | 1) результат меняется на противоположный. Если операнд — ИСТИНА, результат — ЛОЖЬ. И наоборот |
| б) ИЛИ | 2) результат — ИСТИНА, если оба операнда — ИСТИНА   |
| в) НЕ  | 3) результат — ИСТИНА, если хотя бы один операнд — ИСТИНА                                     |

**Напоминание.** Если в выражении присутствует несколько логических операций, они выполняются в соответствии со своими приоритетами.

**10.9.** Укажите порядок выполнения операций в соответствии с их приоритетом (от высшего к низшему). В ответе укажите буквы, соответствующие операциям, в указанном порядке.

- |        |                        |
|--------|------------------------|
| а) ИЛИ | б) выражение в скобках |
| в) И   | г) НЕ                  |

Ответ: \_\_\_\_\_.

**10.10.** Результаты тестирования представлены в таблице:

Фамилия	Пол	Математика	Русский язык	Химия	Информатика	Биология
Аганян	ж	82	56	46	32	70
Воронин	м	43	62	45	74	23
Григорчук	м	54	74	68	75	83
Роднина	ж	71	63	56	82	79
Сергеенко	ж	33	25	74	38	46
Черепанова	ж	18	92	83	28	61

Сколько записей в ней удовлетворяют условию «Пол='ж' ИЛИ Химия>Биология»?

- |      |      |      |      |
|------|------|------|------|
| 1) 5 | 2) 2 | 3) 3 | 4) 4 |
|------|------|------|------|

**Рекомендация.** Выберите записи, которые удовлетворяют первому условию (Пол='ж'). Отметьте их (например, «галочками»). Затем выберите записи, удовлетворяющие второму условию (Химия>Биология). Тоже отметьте их «галочками». Так как для их объединения используется логическая операция ИЛИ, ей удовлетворяют те записи (строки), в которых мы поставили хотя бы одну «галочку».

**10.11.** Результаты тестирования представлены в таблице:

Фамилия	Пол	Математика	Русский язык	Химия	Информатика	Биология
Аганян	ж	82	56	46	32	70
Воронин	м	43	62	45	74	23
Григорчук	м	54	74	68	75	83
Роднина	ж	71	63	56	82	79
Сергеенко	ж	33	25	74	38	46
Черепанова	ж	18	92	83	28	61

Сколько записей в ней удовлетворяют условию «НЕ(Пол='ж' И Математика<Биология)»?

- |      |      |      |      |
|------|------|------|------|
| 1) 5 | 2) 2 | 3) 3 | 4) 4 |
|------|------|------|------|

*Рекомендация.* Такие задания (с операцией НЕ перед всем выражением) можно решать двумя способами. Либо выберите записи, которые удовлетворяют выражению. Тогда все остальные будут ему НЕ удовлетворять, то есть именно то, что нужно найти.

Либо примените закон Де Моргана. Но в этом случае нужно быть очень осторожным и не забыть обратить все условия. В данном случае:

$\text{НЕ}(\text{Пол}='ж' \text{ И Математика} < \text{Биология}) =$

$\text{НЕ}(\text{Пол}='ж') \text{ ИЛИ } \text{НЕ}(\text{Математика} < \text{Биология}) =$

$\text{Пол} < 'ж' \text{ ИЛИ Математика} > \text{Биология}$

**10.12.** Результаты тестирования представлены в таблице:

Фамилия	Пол	Математика	Русский язык	Химия	Информатика	Биология
Аганян	ж	82	56	46	32	70
Воронин	м	43	62	45	74	23
Григорчук	м	54	74	68	75	83
Роднина	ж	71	63	56	82	79
Сергеенко	ж	33	25	74	38	46
Черепанова	ж	18	92	83	28	61

Сколько записей в ней удовлетворяют условию

«Пол='ж' И Математика > Химия»?

1) 5

2) 2

3) 3

4) 4

**10.13.** Результаты тестирования представлены в таблице:

Фамилия	Пол	Математика	Русский язык	Химия	Информатика	Биология
Аганян	ж	82	56	46	32	70
Воронин	м	43	62	45	74	23
Григорчук	м	54	74	68	75	83
Роднина	ж	71	63	56	82	79
Сергеенко	ж	33	25	74	38	46
Черепанова	ж	18	92	83	28	61

Сколько записей в ней удовлетворяют условию

«Пол='ж' И Математика > Русский язык ИЛИ Пол='м' И Информатика > Биология»?

1) 5

2) 2

3) 3

4) 4

*Рекомендация.* Не забудьте, что приоритет операции «Логическое И» выше, чем у операции «Логическое ИЛИ».

**10.14.** Из правил соревнования по тяжелой атлетике:

«Тяжелая атлетика это прямое соревнование, когда каждый атлет имеет три попытки в рывке и три попытки в толчке. Самый тяжелый вес поднятой штанги в каждом упражнении суммируется в общем зачете. Если спортсмен потерпел неудачу во всех трех попытках в рывке, он может продолжить соревнование в толчке, но уже не сможет занять какое-либо место по сумме двух упражнений.

Если два спортсмена заканчивают состязание с одинаковым итоговым результатом, высшее место присуждается спортсмену с меньшим весом. Если же вес спортсменов одинаков, преимущество отдается тому, кто первым поднял победный вес».

Таблица результатов соревнований по тяжелой атлетике:

Фамилия, И.О.	Вес спортсмена	Взято в рывке	Рывок с попытки	Взято в толчке	Толчок с попытки
Айвазян Г.С.	77,1	150,0	3	200,0	2
Викторов М.П.	79,1	147,5	1	202,5	1
Гордезиани Б.Ш.	78,2	147,5	2	200,0	1
Михальчук М.С.	78,2	147,5	2	202,5	3
Пай С.В.	79,5	150,0	1	200,0	1
Шапсугов М.Х.	77,1	147,5	1	200,0	1

Кто победил в общем зачете (сумме двух упражнений)?

- 1) Айвазян Г.С.
- 2) Викторов М.П.
- 3) Михальчук М.С.
- 4) Пай С.В.

**10.15.** В таблице собраны сведения о членах школьной волейбольной команды:

Фамилия	Вес	Рост	Пол
Иванов	87	180	М
Петрова	55	170	Ж
Сидоров	67	155	М
Пупкина	78	160	Ж

Сколько записей удовлетворяют условию:

девочки выше 165 см И легче 60 кг ИЛИ учащиеся тяжелее 75 кг?

Ответ: \_\_\_\_\_

**10.16.** На городской олимпиаде по программированию предлагались задачи трех типов: А, В и С. По итогам олимпиады была составлена таблица, в колонках которой указано, сколько задач каждого типа решил участник. Вот начало таблицы:

Фамилия	А	В	С
Иванов	3	2	1

За правильное решение задачи типа А участнику начислялся 1 балл, за решение задачи типа В — 2 балла и за решение задачи типа С — 3 балла. Победитель определялся по сумме баллов, которая у всех участников оказалась разная. Для определения победителя олимпиады достаточно выполнить следующий запрос:

- 1) отсортировать таблицу по возрастанию значения поля С и взять первую строку
- 2) отсортировать таблицу по убыванию значения поля С и взять первую строку
- 3) отсортировать таблицу по убыванию значения выражения  $A + 2B + 3C$  и взять первую строку
- 4) отсортировать таблицу по возрастанию значения выражения  $A + 2B + 3C$  и взять первую строку

*Рекомендация.* Проанализируйте принцип начисления баллов и выведите формулу суммы баллов. Если победитель выбирается из первой строки запроса, проанализируйте, ка-

кое количество баллов он должен набрать — наибольшее или наименьшее? Как в этом случае должна быть отсортирована таблица?

**10.17.** На городской олимпиаде по программированию предлагались задачи трех типов: А, В и С. По итогам олимпиады была составлена таблица, в колонках которой указано, сколько задач каждого типа решил участник. Вот начало таблицы:

Фамилия	А	В	С
Иванов	4	1	3

За правильное решение задачи типа А участнику начислялся 3 балла, за решение задачи типа В — 2 балла и за решение задачи типа С — 1 балл. Победитель определялся по сумме баллов, которая у всех участников оказалась разная. Для определения победителя олимпиады достаточно выполнить следующий запрос:

- 1) отсортировать таблицу по возрастанию значения поля С и взять первую строку
- 2) отсортировать таблицу по убыванию значения поля С и взять первую строку
- 3) отсортировать таблицу по возрастанию значения выражения  $3A+2B+C$  и взять первую строку
- 4) отсортировать таблицу по убыванию значения выражения  $3A+2B+C$  и взять первую строку

# 11. Сетевые технологии

## Основные понятия

*Адресация в Интернете.* Для обмена информации между компьютерами необходимы:

- канал связи, по которому идет передача данных;
- аппаратные средства, которыми компьютер связывается с каналом связи (модем, сетевая карта);
- программные средства, которыми при этом пользуется компьютер или человек (браузер, почтовая программа, менеджер обмена сообщениями);
- протокол (соглашение, согласно которому происходит процедура обмена).

Для соединения через Интернет каждый компьютер должен обладать протоколом TCP/IP. Согласно этому протоколу, в частности, каждый компьютер должен иметь уникальный адрес, состоящий из четырех чисел (от 0 до 255 каждое), разделенных точками (например, 172.16.1.95). Это называется *IP-адрес*. Так как каждое число IP-адреса лежит в диапазоне 0–255, нетрудно подсчитать, что для его хранения требуется 8 бит, то есть 1 байт.

Также нетрудно подсчитать, что общее количество IP-адресов не может быть больше чем  $256^4$ . То есть, около 4 миллиардов. Это, однако, не означает, что к Интернету не может быть подключено больше 4 миллиардов компьютеров. Специальные технологии позволяют целой группе компьютеров использовать один общий IP-адрес для выхода в Интернет.

Некоторые компьютеры в сети предоставляют свои ресурсы в общее пользование. Они называются *серверами*. Как правило, это серверы с информацией. Так как множество людей часто обращаются к серверам, неудобно помнить для каждого нужного сервера его IP-адрес. Для таких серверов используются буквенные названия, которые называются *доменным адресом*. Например, www.mail.ru. Когда нужно обратиться к серверу по доменному адресу, компьютер сначала посылает запрос в специальную службу (она называется DNS), которая по доменному адресу выдает IP-адрес этого сервера.

Доменные адреса обычно выбирают по территориальному принципу или по принципу принадлежности сервера к организации какого-либо типа.

Пример территориального принципа выбора доменного адреса: sch239.spb.ru (Россия, Санкт-Петербург, школа 239).

По принципу принадлежности организации к какому-либо типу обычно действуют организации из США (родины Интернета). Примеры: whitehouse.gov, pentagon.mil, greenpeace.org, mti.edu.

Для указания местоположения какого-либо ресурса (файла) в сети Интернет используют URL (универсальный указатель местоположения ресурса). Он состоит из трех частей: имени протокола, по которому будет происходить передача файла, затем через символы «://» адрес компьютера, на котором этот файл хранится (IP-адрес или доменный адрес), затем через символ «/» имя файла на этом компьютере (иногда с указанием пути к файлу). Пример: http://masha.ru/risunok.jpg.

Для поиска информации в Интернете используют *поисковые системы* — специальные программно-аппаратные комплексы (базы данных), которые хранят информацию о множестве страниц Интернета и которые предоставляют возможность поиска среди этой информации.

Для поиска нужно указать ключевые слова, которые должны содержаться в искомом документе. Чтобы указать, должны ли эти слова присутствовать в документе одновременно

или же достаточно одного из них, используют логические операции (И и ИЛИ). Операцию И обозначают значком «&» (амперсанд), операцию ИЛИ обозначают значком «|» (вертикальная черта). Приоритет операции И выше, чем у операции ИЛИ.

Для изменения порядка выполнения операций используют скобки.

Если несколько ключевых слов объединены операцией И, количество найденных страниц будет меньше, чем если эти слова искать по отдельности.

Если несколько ключевых слов объединены операцией ИЛИ, количество найденных страниц будет, наоборот, больше.

## Практическая часть

**11.1.** Укажите одно или несколько правильных утверждений. Буквы, соответствующие правильным утверждениям, обведите, остальные — зачеркните.

Для компьютерной сети Интернет справедливо:

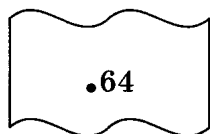
- а) каждый компьютер, подключенный к сети, должен иметь собственный адрес — IP-адрес
- б) IP-адрес состоит из четырех чисел, разделенных точками
- в) для хранения каждого из чисел IP-адреса выделяют по 1 байту
- г) каждое из чисел IP-адреса может быть от 0 до 255
- д) некоторые числа IP-адреса можно заменять на слова (например, www.mail.ru)
- е) некоторые комбинации чисел IP-адреса неприемлемы ввиду договоренностей об IP-адресах
- ж) наибольшее значение (255) для чисел в IP-адресе происходит от размера ячейки памяти, выделяемой для IP-адреса
- з) общее количество IP-адресов, которое может существовать, составляет  $256^4$  (примерно 4 миллиарда)
- и) количество компьютеров в сети не может превышать четырех миллиардов ввиду ограничения на количество возможных IP-адресов

**11.2.** Сколько точек в IP-адресе?

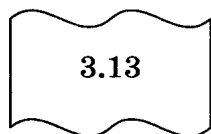
Ответ: \_\_\_\_\_.

**11.3.** Петя записал IP-адрес школьного сервера на листке бумаги и положил его в карман куртки. Петина мама случайно постирала куртку вместе с запиской. После стирки Петя обнаружил в кармане четыре обрывка с фрагментами IP-адреса. Эти фрагменты обозначены буквами А, Б, В и Г. Восстановите IP-адрес.

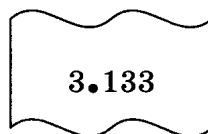
В ответе укажите последовательность букв, обозначающих фрагменты, в порядке, соответствующем IP-адресу.



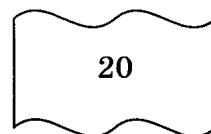
А



Б



В



Г

Ответ: \_\_\_\_\_.

*Рекомендация.* Честно говоря, мы заметили, что проще всего решать такую задачу, действительно написав на четырех клочках бумаги куски IP-адреса и перекладывая их. Правда, мы не уверены, что если вы начнете заниматься этим на экзамене, вам не сделают замечание.

Основная идея решения этой задачи — в недопустимости чисел IP-адреса, которые превышают 255. В данном случае задача решается совсем просто, ввиду быстрого установления двух обрывков на свои места. Обрывок А не может быть первым (из-за точки). Кроме того, так как число в нем больше 25, то справа к нему нельзя дописать ни одной цифры, иначе получатся числа больше 640. Значит, 64 — одно из чисел IP-адреса, причем не первое. А так как больше обрывков, начинающихся с точки, нет, значит, обрывок А стоит последним.

К обрывку В не может быть приставлена справа ни одна цифра, так как число 133 уже трехзначное. Значит, обрывок В должен быть или последним, или после него должна стоять точка. Последнее место уже занято. Значит, обрывок В предпоследний.

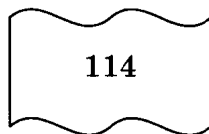
Оставшиеся 2 варианта нетрудно просто проверить.

- 11.4.** Петя записал IP-адрес школьного сервера на листке бумаги и положил его в карман куртки. Петина мама случайно постирала куртку вместе с запиской. После стирки Петя обнаружил в кармане четыре обрывка с фрагментами IP-адреса. Эти фрагменты обозначены буквами А, Б, В и Г. Восстановите IP-адрес.

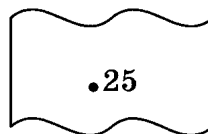
В ответе укажите последовательность букв, обозначающих фрагменты, в порядке, соответствующем IP-адресу.



А



Б



В

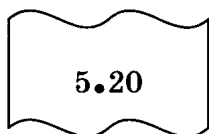


Г

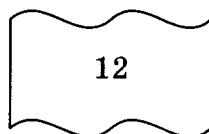
Ответ: \_\_\_\_\_.

- 11.5.** Петя записал IP-адрес школьного сервера на листке бумаги и положил его в карман куртки. Петина мама случайно постирала куртку вместе с запиской. После стирки Петя обнаружил в кармане четыре обрывка с фрагментами IP-адреса. Эти фрагменты обозначены буквами А, Б, В и Г. Восстановите IP-адрес.

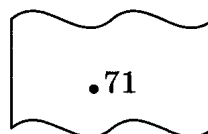
В ответе укажите последовательность букв, обозначающих фрагменты, в порядке, соответствующем IP-адресу.



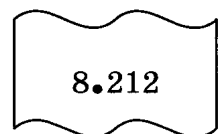
А



Б



В



Г

Ответ: \_\_\_\_\_.

- 11.6.** Укажите одно или несколько правильных утверждений. Буквы, соответствующие правильным утверждениям, обведите, остальные — зачеркните.

- а) Для указания местоположения ресурса в сети Интернет используется специальный адрес, называемый URL
- б) URL — Uniform Resource Locator

- в) URL состоит из трех частей — протокол, символы «://», адрес компьютера в сети, символ «/», местоположение файла на этом компьютере
- г) Пример URL: «http://www.prazdnik.ru/files/holyday.jpg»
- д) Протокол — договоренность о способе обмена информацией
- е) Примеры наиболее часто используемых протоколов: http, ftp
- ж) Адрес компьютера в URL может быть доменным или ip-адресом
- з) Допустимым считается, например, такой URL: «ftp://10.20.4.58/index.html»
- и) Местоположение файла в URL может включать в себя последовательность имен папок от «стартового положения» файлов на сервере, разделенных символами «/»
- к) URL всегда заканчивается именем файла

**11.7.** Доступ к файлу htm.net, находящемуся на сервере com.edu, осуществляется по протоколу ftp. В таблице фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла в сети Интернет.

А	/
Б	com
В	.edu
Г	://
Д	.net
Е	htm
Ж	ftp

Ответ: \_\_\_\_\_.

*Рекомендация.* Чтобы не ошибиться в символах и буквах, сначала аккуратно запишите на черновике получающийся URL. Не забудьте записать между протоколом и именем сервера символы «://», а между именем сервера и именем файла — символ «/». Не старайтесь использовать привычные имена серверов и имена файлов. Не старайтесь расставлять их в том порядке, как вы привыкли ими пользоваться. Просто «тупо» запишите последовательность, которая указана в задаче: протокол, «://», имя сервера, «/», имя файла. Потом подпишите под каждой из получившихся частей URL свою букву по таблице. Не забудьте проверить, что у вас получились все 7 букв и что они не повторяются.

**11.8.** Доступ к файлу www.txt, находящемуся на сервере http.net, осуществляется по протоколу ftp. В таблице фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла.

А	http
Б	.txt
В	.net
Г	ftp
Д	/
Е	www
Ж	://

Ответ: \_\_\_\_\_.



**11.9.** Доступ к файлу txt.com, находящемуся на сервере www.ftp, осуществляется по протоколу http. В таблице фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла.

А	.ftp
Б	txt
В	://
Г	www
Д	http
Е	/
Ж	.com

Ответ: \_\_\_\_\_.

**11.10.** Доступ к файлу com.ftp, находящемуся на сервере www.edu, осуществляется по протоколу http. В таблице фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла.

А	.ftp
Б	/
В	.edu
Г	://
Д	www
Е	com
Ж	http

Ответ: \_\_\_\_\_.

### **Напоминание.** Поиск информации в Интернете.

При пользовании поисковыми системами запрос, который делает пользователь, обрабатывается базой данных, хранящейся на поисковом сервере. Ключевые слова, которые вводит пользователь в тексте запроса, ищутся по отдельности в базе данных, после чего результаты каждого поиска соединяются в зависимости от логических операций, которыми объединены ключевые слова.

Если используется логическая операция «И», то в результат войдут только те страницы, которые содержат одновременно оба ключевых слова (мы говорим «оба», так как логические операции «И» и «ИЛИ» всегда имеют ровно два операнда). Значит, в результате поиска количество страниц, которое найдет поисковый сервер, будет меньше, чем в запросах по каждому ключевому слову в отдельности.

Если используется логическая операция «ИЛИ», то в результат войдут как страницы, которые содержат только одно из ключевых слов, так и страницы, которые содержат только второе ключевое слово. Также результат будет содержать страницы, которые содержат сразу оба ключевых слова. Значит, в результате поиска количество страниц, которое найдет поисковый сервер, будет больше, чем по каждому ключевому слову в отдельности.

Не забудьте, что приоритеты логических операций здесь действуют как обычно. Приоритет операции «И» выше, чем у операции «ИЛИ» (см. главу 4 «Основы логики»).

*Замечание.* Мы, конечно, слегка грешим против истины, для упрощения понимания. В действительности правильнее говорить, что результат запроса с логической операцией «И» не больше

результата запроса с каждым ключевым словом в отдельности. Так же как и результат запроса с логической операцией «ИЛИ» не меньше запросов с этими словами по отдельности.

Эта оговорка работает, когда, например, в базе данных существуют только страницы, в которых оба ключевых слова встречаются одновременно, а по отдельности эти ключевые слова не встречаются. Но мы такой случай для общности и простоты задачи, не рассматриваем. Вам мы советуем при решении такого типа задач тоже считать, что это не наш случай.

**11.11.** В таблице приведены запросы к поисковому серверу. Расположите номера запросов в порядке возрастания количества страниц, которые найдет поисковый сервер по каждому запросу.

Для обозначения логической операции «ИЛИ» в запросе используется символ |, а для логической операции «И» — &.

1	принтеры   продажа
2	принтеры
3	принтеры & продажа

Ответ: \_\_\_\_\_.

**11.12.** В таблице приведены запросы к поисковому серверу. Расположите номера запросов в порядке возрастания количества страниц, которые найдет поисковый сервер по каждому запросу.

Для обозначения логической операции «ИЛИ» в запросе используется символ |, а для логической операции «И» — &.

1	коньки   лыжи
2	коньки & лыжи & санки
3	коньки & лыжи
4	лыжи

Ответ: \_\_\_\_\_.

**11.13.** В таблице приведены запросы к поисковому серверу. Расположите номера запросов в порядке возрастания количества страниц, которые найдет поисковый сервер по каждому запросу.

Для обозначения логической операции «ИЛИ» в запросе используется символ |, а для логической операции «И» — &.

1	принтеры & сканеры & продажа
2	принтеры & продажа
3	принтеры   продажа
4	принтеры   сканеры   продажа

Ответ: \_\_\_\_\_.

*Рекомендация.* Учтите, что условие «логическое И» приводит к уменьшению количества найденных страниц. Соответственно, применение нескольких «логических И» еще более уменьшает число найденных страниц.

В отличие от «И», «логическое ИЛИ» приводит к увеличению числа найденных страниц. Соответственно, применение нескольких «логических ИЛИ» еще более увеличивает количество найденных страниц.

Постарайтесь быть внимательным и обратить внимание на порядок возрастания/убывания. При возрастании нужно сначала указать самое строгое условие (наименьшее число найденных страниц), а в конце — самое «широкое» (больше всего «ИЛИ»). При убывании — наоборот, в обратном порядке.

**11.14.** В таблице приведены запросы к поисковому серверу. Расположите номера запросов в порядке возрастания количества страниц, которые найдет поисковый сервер по каждому запросу.

Для обозначения логической операции «ИЛИ» в запросе используется символ |, а для логической операции «И» — &.

1	физкультура
2	физкультура & подтягивания & отжимания
3	физкультура & подтягивания
4	физкультура   фитнес

Ответ: \_\_\_\_\_.

**11.15.** В таблице приведены запросы к поисковому серверу. Расположите номера запросов в порядке убывания количества страниц, которые найдет поисковый сервер по каждому запросу.

Для обозначения логической операции «ИЛИ» в запросе используется символ |, а для логической операции «И» — &.

1	алгоритм & (программа   данные)
2	алгоритм & программа & данные
3	программа   данные
4	алгоритм   программа   данные

Ответ: \_\_\_\_\_.

**11.16.** В таблице приведены запросы к поисковому серверу. Расположите номера запросов в порядке убывания количества страниц, которые найдет поисковый сервер по каждому запросу.

Для обозначения логической операции «ИЛИ» в запросе используется символ |, а для логической операции «И» — &.

1	алгоритм & программа   данные
2	алгоритм & программа & данные
3	алгоритм   данные
4	алгоритм   программа   данные

Ответ: \_\_\_\_\_.

**11.17.** В таблице приведены запросы к поисковому серверу. Расположите номера запросов в порядке убывания количества страниц, которые найдет поисковый сервер по каждому запросу.

Для обозначения логической операции «ИЛИ» в запросе используется символ |, а для логической операции «И» — &.

1	алгоритм   программа & данные
2	алгоритм   программа   данные
3	программа & данные
4	алгоритм & программа & данные

Ответ: \_\_\_\_\_.

**11.18.** В таблице приведены запросы к поисковому серверу. Расположите номера запросов в порядке убывания количества страниц, которые найдет поисковый сервер по каждому запросу.

Для обозначения логической операции «ИЛИ» в запросе используется символ |, а для логической операции «И» — &.

1	рога & копыта   хвосты
2	рога   копыта   хвосты
3	рога & копыта
4	рога & (копыта   хвосты)

Ответ: \_\_\_\_\_.

# Ответы

## 1. Количество информации.

### Скорость передачи информации

1.1. в), е); 1.2. а), б), г), д); 1.3. 1; 1.4. 3; 1.5. 3; 1.6. 7; 1.7. 6; 1.8. 8; 1.9. 250; 1.10. 490; 1.11. 420; 1.12. 600; 1.13. 560; 1.14. 600; 1.15. 10; 1.16. 5; 1.17. 10; 1.18. 2048; 1.19. 128; 1.20. 25; 1.21. 40; 1.22. 32; 1.23. 8; 1.24. 37,5; 1.25. 3750; 1.26. 3; 1.27. 96; 1.28. 108; 1.29. 4.

## 2. Системы счисления

2.1. Система счисления — б); алфавит системы счисления — г); мощность алфавита — а); основание системы счисления — д).

2.2. Для каждого числа позиционной системы счисления — вес цифры зависит от ее положения в числе.

2.3. В основание (в десятичной — в 10, в восьмеричной — в 8, ...)

2.4. N, 0, N-1; 2.5.  $10_N$ ; 2.6. 0 и 1; 2.7. 6, A, пятнадцать.

2.8. 52436 — разряды, 43210 — номера разрядов, 8 — основание системы счисления

2.9.  $2C_{16}$  и  $2D_{16}$

2.10.

### Система счисления

10	2	8	16	3	4	5
0	0	0	0	0	0	0
1	1	1	1	1	1	1
2	10	2	2	2	2	2
3	11	3	3	10	3	3
4	100	4	4	11	10	4
5	101	5	5	12	11	10
6	110	6	6	20	12	11
7	111	7	7	21	13	12
8	1000	10	8	22	20	13
9	1001	11	9	100	21	14
10	1010	12	A	101	22	20
11	1011	13	B	102	23	21
12	1100	14	C	110	30	22
13	1101	15	D	111	31	23
14	1110	16	E	112	32	24
15	1111	17	F	120	33	30
16	10000	20	10	121	100	31
17	10001	21	11	122	101	32

2.11.  $101000_2$ ,  $220_3$ ,  $40_8$ ,  $C0_{16}$ ,  $22000_4$ ,  $100_7$

2.12.  $10011_2$ ,  $2022_3$ ,  $517_8$ ,  $9FF_{16}$ ,  $3113_4$ ,  $46_7$

2.13.  $10000_2$ ,  $C0_{16}$ ,  $2222_3$ ,  $10A_{16}$

2.14. Пары меньше-большее:

$1111_2-11000_2$ ,  $1112_3-2212_3$ ,  $277_8-630_8$ ,  $10D_{16}-E0F_{16}$ ,  $10000_4-32100_4$ ,  $1FF_{16}-F00_{16}$

2.15. 3, 1, 2

2.16. 3, 2, 1

$$2.17. 2314_3 = 2 \cdot 5^3 + 3 \cdot 5^2 + 1 \cdot 5^1 + 4 \cdot 5^0 = 250 + 75 + 5 + 4 = 334_{10}$$

$$1010011_2 = 2^6 + 2^4 + 2^1 + 1 = 64 + 16 + 2 + 1 = 83_{10}$$

$$2AB_{16} = 2 \cdot 16^2 + 10 \cdot 16^1 + 11 = 512 + 160 + 11 = 683_{10}$$

$$2.18. 1010110_2 = 86_{10}, 21020_3 = 195_{10}, 526_8 = 342_{10}, 3CE_{16} = 974_{10}, 3021_4 = 201_{10}, 256_7 = 139_{10}$$

2.19. 6 целых и 5 в остатке

2.20. 4 целых и 3 в остатке

2.21. 0 целых и 7 в остатке

$$2.22. 53 = 110101_2$$

$$2.23. 202 = 1302_5$$

$$2.24. 57 = 111001_2, 65 = 2102_3, 335 = 517_8, 202 = CA_{16}, 198 = 3012_4, 139 = 256_7$$

2.25. 3, 1, 2

2.26.  $215_8$

2.27.  $374_8, 126_8$

2.28. 4, 1 (группами по 4), 2, 3

2.29.  $6AD_{16}$

2.30.  $3FC_{16}, 75E_{16}$

2.31. 3, 1, 2

2.32.  $111001100_2$

2.33.  $110000011_2, 011010101_2$

2.34. 3, 2 (из 4 разрядов, 4 разряда), 1

2.35.  $110100100111_2$

2.36.  $111001000011_2, 101000001100_2$

2.37. 3; 2.38. 4; 2.39. 1; 2.40. 3; 2.41. 3; 2.42. 4; 2.43. 5; 2.44. 2; 2.45. 3; 2.46. 4; 2.47. 6;

2.48. 4; 2.49. 3; 2.50. 5, 21; 2.51. 6, 14, 22; 2.52. 7, 16, 25; 2.53. 8, 33; 2.54. 5, 13, 21;

2.55. 3, 7, 21; 2.56. 4, 6, 8, 12, 24; 2.57. 11, 22; 2.58. 9, 27; 2.59. 13, 26; 2.60. 4; 2.61. 3;

2.62. 5; 2.63. 4; 2.64. 4; 2.65.  $121_8$ ; 2.66.  $1000101101_2$ ; 2.67.  $1031_4$ ; 2.68.  $17B_{16}$ ; 2.69. 2; 2.70. 4;

2.71. 2; 2.72. 3; 2.73. 1; 2.74.  $1010111_2$ ; 2.75.  $11100011_2$ ; 2.76.  $1111001_2$

2.77. Для  $N=7$ :

1) 237; 2)  $11101101_2, 22210_3, 1422_5$ ; 3)  $11111000_2$ ; 4)  $370_8, F8_{16}$ ; 5)  $427_8$ ; 6) 279;

7)  $D9_{16}$ ; 8) 217; 9)  $100122_3$ ; 10)  $1333_5$

### 3. Кодирование информации

3.1. б); 3.2. все; 3.3. 60; 3.4. 512; 3.5. 384; 3.6. 4; 3.7. 1; 3.8. 4; 3.9. 375; 3.10. 3; 3.11. 3;

3.12. а-5, б-6, в-1, г-4, д-2, е-3; 3.13. а-3, б-4, в-5, г-2, д-1; 3.14. 1; 3.15. 4; 3.16. 2; 3.17. 4;

3.18. 1; 3.19. 8; 3.20. 5; 3.21. 7; 3.22. 16; 3.23. 8; 3.24. 2; 3.25. 256; 3.26. 65536; 3.27. 3;

3.28. 1536; 3.29. 256; 3.30. 320; 3.31. 8; 3.32. 48; 3.33. 2048; 3.34. 1024; 3.35. 32; 3.36. 8;

3.37. 128.

### 4. Основы логики

4.3. а)  $A = 1, B = 1$ ; б)  $A = 0, B = 0$ ; в)  $A = 1, B = 0$ ; г)  $A = 1, B = 1$

4.7. а) 1; б) 2.

4.8. а) 1; б) 3; в) 2; г) 4.

4.9. а) 3; б) 3.

4.10. а) 4; б) 3; в) 2; г) 4.

4.11. а) 3; б) 15.

4.12. а) 2; б) 4.

4.13. а) 0001; б) 1000.

4.14. а) 12; б) 1.

4.15. а) П; б) КРС; в) А, И; г) Валера

## 5. Алгоритмизация и программирование

5.1. а) 25; б) 8

5.2. а)  $+\cdot\cdot\cdot\cdot+$ ; б)  $\cdot+\cdot\cdot++$

5.10. а)  $n(n-1)/2$ ; б)  $n-n^2$

5.14. а) 2; б) 1. 5.15. а) 2; б) 345. 5.17. а) 122111; б) 21211. 5.18. а) 2, A8, H4; б) 2, G1, F3

5.19. а) 511; б) 10. 5.20. а) 2; б) 1. 5.21. а) 1; б) 1

## 6. Информационные модели

6.1. 2; 6.2. 3; 6.3. 3; 6.4. 3,5; 6.5. 17:25; 6.6. 18:15; 6.7. 16:30; 6.8. 16:20; 6.9. 14:10.

## 7. Определение выигрышной стратегии игры (Анализ и построение дерева игры)

7.1. См. полное решение и ответ в тексте

7.2. Выигрывает второй игрок.

I ход I-го игрока	I ход II-го игрока (выигрышный)	Дальнейшие ходы
(5, 3)	(5, 6)	При любом последующем ходе I игрока второй игрок должен передвинуть фишку на $(x+4, y)$ и тем самым выиграть
(1, 6)	(5, 6)	
(1, 7)	(5, 7)	

7.3. Выигрывает второй игрок.

I ход I-го игрока	I ход II-го игрока (выигрышный)	Дальнейшие ходы
(4, 2)	(7, 5)	При любом последующем ходе I игрока второй игрок должен передвинуть фишку на $(x+3, y+3)$ и тем самым выиграть
(4, 5)	(7, 5)	
(1, 5)	(1, 8)	

7.4. См. полное решение и ответ в тексте.

7.5. Выигрывает первый игрок. Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры.

	1-й ход	2-й ход	3-й ход	4-й ход	5-й ход	
Стартовая позиция	I-й игрок (выигрышный ход)	II-й игрок (все варианты хода)	I-й игрок (выигрышный ход)	II-й игрок (все варианты хода)	I-й игрок (выигрышный ход)	
1, 0	<u>4, 0</u>	7, 0	<u>7, 4</u>	10, 4	<u>13, 4</u>	
				7, 7	<u>7, 11</u>	
				7, 8	<u>7, 11</u>	
		4, 3	<u>4, 7</u>	<u>4, 7</u>	7, 7	<u>7, 11</u>
					4, 10	<u>4, 14</u>
					4, 11	<u>4, 14</u>
		4, 4	<u>4, 7</u>	<u>4, 7</u>	7, 7	<u>7, 11</u>
					4, 10	<u>4, 14</u>
					4, 11	<u>4, 14</u>

Таблица содержит все возможные варианты ответных ходов второго игрока на выигрышный ход первого игрока. Из нее видно, что при любом ходе второго игрока у первого имеется ход, приводящий к победе.

**7.6.** Выигрывает первый игрок. Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры.

	1-й ход	2-й ход	3-й ход	4-й ход	5-й ход
Стартовая позиция	I-й игрок (выигрышный ход)	II-й игрок (все варианты хода)	I-й игрок (выигрышный ход)	II-й игрок (все варианты хода)	I-й игрок (выигрышный ход)
1, 2	<u>4, 2</u>	7, 2	<u>7, 5</u>	10, 5	<u>13, 5</u>
				7, 8	<u>7, 12</u>
				7, 9	<u>7, 13</u>
		4, 5	<u>7, 5</u>	10, 5	<u>13, 5</u>
				7, 8	<u>7, 12</u>
				7, 9	<u>7, 13</u>
		4, 6	<u>7, 6</u>	10, 6	<u>13, 6</u>
				7, 9	<u>7, 13</u>
				7, 10	<u>7, 14</u>

Таблица содержит все возможные варианты ответных ходов второго игрока на выигрышный ход первого игрока. Из нее видно, что при любом ходе второго игрока у первого имеется ход, приводящий к победе. (Вообще говоря, это тот случай, когда любой ход первого игрока приводит его к победе.)

**7.7.** Выигрывает первый игрок. Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры.

	1-й ход	2-й ход	3-й ход
Стартовая позиция	I-й игрок (выигрышный ход)	II-й игрок (все варианты хода)	I-й игрок (выигрышный ход)
1, 0	<u>5, 4</u>	9, 4	<u>13, 8</u>
		9, 8	<u>13, 12</u>
		5, 8	<u>9, 12</u>

Таблица содержит все возможные варианты ответных ходов второго игрока на выигрышный ход первого игрока. Из нее видно, что при любом ходе второго игрока у первого имеется ход, приводящий к победе.

## 8. Файловая система компьютера

8.5. а) 3; б) 1. 8.6. а) 3; б) 4. 8.7. а) 3; б) 3; в) 2; г) 4.

## 9. Электронные таблицы

9.1. все. 9.2. а), б), г), д), з). 9.3. б), в), д), з), к). 9.4. а-3, б-9, в-5, г-11, д-1, е-10, ж-4, з-7, и-6, к-8, л-2. 9.5. а), б), г), д), е), ж), з), и) (т.е., все, кроме в). 9.6. а-4, б-5, в-1, г-3, д-2. 9.7. а)1; б) 3; в) 4; г) 12; д) 9; е) 4; ж) 18; з) 256. 9.8. а) 0; б) 7; в) 10; г) 2,5; д) 6

9.9. 4; 9.10. 2; 9.11. 1; 9.12. -11; 9.13. 8; 9.14. 52; 9.15. =\$B2+D\$2; 9.16. =\$C1-A\$3; 9.17. =\$A\$4+D1; 9.18. =2 \* \$A2; 9.19. 2; 9.20. 4; 9.21. 1; 9.22. 1; 9.23. а), в); 9.24. 1; 9.25. б), в).

## 10. Базы данных

10.1. все; 10.2. а-4, б-6, в-3, г-5, д-1, е-2; 10.3. 3; 10.4. 3; 10.5. 5; 10.6. 5; 10.7. 4; 10.8. а-1, б-3, в-1; 10.9. бгва; 10.10. 1; 10.11. 3; 10.12. 2; 10.13. 4; 10.14. 1; 10.15. 3; 10.16. 3; 10.17. 4.

## 11. Сетевые технологии

11.1. а), б), в), г), е), ж), з); 11.2. 3; 11.3. ГБВА; 11.4. БВГА; 11.5. БАГВ; 11.6. все; 11.7. ЖГБВАЕД; 11.8. ГЖАВДЕБ; 11.9. ДВГАЕБЖ; 11.10. ЖГДВБЕА; 11.11. 321; 11.12. 2341; 11.13. 1234; 11.14. 2314; 11.15. 4312; 11.16. 4312; 11.17. 2134; 11.18. 2143.



*Учебно-методическое издание*

**Крылов Сергей Сергеевич  
Ушаков Денис Михайлович**

# ЕГЭ

# ИНФОРМАТИКА

## ТЕМАТИЧЕСКАЯ РАБОЧАЯ ТЕТРАДЬ ФИПИ

Издательство «**ЭКЗАМЕН**»

Гигиенический сертификат  
№ 77.99.60.953.Д.000454.01.09 от 27.01.2009 г.

Редактор *Г.А. Лонцова*  
Технический редактор *Т.В. Фатюхина*  
Корректор *О.Ю. Казанаева*  
Дизайн обложки *Л.В. Демьянова*  
Компьютерная верстка *М.В. Демина*

105066, Москва, ул. Нижняя Красносельская, д. 35, стр. 1.

[www.examen.biz](http://www.examen.biz)

E-mail: по общим вопросам: [info@examen.biz](mailto:info@examen.biz);  
по вопросам реализации: [sale@examen.biz](mailto:sale@examen.biz)  
тел./факс 641-00-30 (многоканальный)

Общероссийский классификатор продукции  
ОК 005-93, том 2; 953005 — книги, брошюры, литература учебная

Отпечатано по технологии CtP  
в ОАО «Печатный двор» им. А.М. Горького  
197110, Санкт-Петербург, Чкаловский пр., 15

**По вопросам реализации обращаться по тел.: 641-00-30 (многоканальный).**