

**ПОДГОТОВКА К ЕГЭ
ВЫСШИЙ УРОВЕНЬ КАЧЕСТВА**



ИНФОРМАТИКА

СБОРНИК ЗАДАНИЙ

- ✓ Более 300 заданий частей А, В и С
- ✓ Ответы и комментарии



**ПОДГОТОВКА К ЕГЭ
ВЫСШИЙ УРОВЕНЬ КАЧЕСТВА**



Е.М. Зорина, М.В. Зорин

ИНФОРМАТИКА

СБОРНИК ЗАДАНИЙ

Рекомендовано «Институтом содержания и методов обучения»
Российской академии образования



Москва **эксмо** 2011

УДК 373.167.1:004
ББК 32.97
З-86

Рецензент:
Ю. Г. Коротенков — кандидат
физико-математических наук

Зорина Е. М.
З-86 ЕГЭ 2012. Информатика : сборник заданий / Е. М. Зорина,
М. В. Зорин. — М. : Эксмо, 2011. — 208 с. — (ЕГЭ. Сборник зада-
ний).

ISBN 978-5-699-50665-1

Издание адресовано выпускникам средней школы и абитуриентам для подготов-
ки к единому государственному экзамену (ЕГЭ) по информатике.

Пособие включает:

- задания уровней А, В, С по всем темам ЕГЭ;
- методические рекомендации;
- ответы ко всем заданиям.

Пособие окажет помощь учителям, репетиторам и родителям при подготовке уча-
щихся к ЕГЭ по информатике.

УДК 373.167.1:004
ББК 32.97

ISBN 978-5-699-50665-1

© Зорина Е. М., Зорин М. В., 2011
© Оформление. ООО «Издательство «Эксмо», 2011

СОДЕРЖАНИЕ

Введение	4
ИНФОРМАЦИЯ И ЕЕ КОДИРОВАНИЕ	6
АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ	30
ОСНОВЫ ЛОГИКИ	66
МОДЕЛИРОВАНИЕ И КОМПЬЮТЕРНЫЙ ЭКСПЕРИМЕНТ	78
ПРОГРАММНЫЕ СРЕДСТВА ИКТ	92
ТЕХНОЛОГИЯ ОБРАБОТКИ ГРАФИЧЕСКОЙ ИНФОРМАЦИИ	99
ТЕХНОЛОГИЯ ОБРАБОТКИ ИНФОРМАЦИИ В ЭЛЕКТРОННЫХ ТАБЛИЦАХ (НА ПРИМЕРЕ EXCEL)	107
ТЕХНОЛОГИЯ ХРАНЕНИЯ, ПОИСКА И СОРТИРОВКИ В БАЗАХ ДАННЫХ (НА ПРИМЕРЕ ACCESS)	118
ТЕЛЕКОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ	132
ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ	141
Ответы	151

ВВЕДЕНИЕ

Сборник, который вы держите в руках, предназначен для самостоятельной работы учащихся выпускных классов, готовящихся к сдаче ЕГЭ по информатике. Эта книга также может быть полезна методистам и преподавателям информатики при организации подготовки учащихся к единому государственному экзамену.

Важно знать содержание и структуру экзаменационной работы по информатике. Экзаменационная работа состоит из трех частей и содержит 32 задания. Часть 1(А) содержит 18 заданий из всех тематических блоков, кроме заданий по технологии телекоммуникаций и технологии программирования. Задания части 1(А) предполагают выбор одного ответа из четырех предложенных.

Часть 2(В) содержит 10 заданий с кратким ответом.

Часть 3(С) содержит 4 задания с развернутым ответом, которые направлены на проверку сформированности важнейших умений записи и анализа алгоритмов.

Задания частей А и В включают в себя все три уровня сложности (базовый, повышенный, высокий), а часть С — задания только высокого уровня сложности.

На экзамен отводится 4 часа (240 минут), из которых на части А и В методисты советуют потратить 1,5 часа, а на часть С — 2,5 часа.

Задания, включенные в данный сборник, сгруппированы в разделы, которые содержатся в спецификации ЕГЭ по информатике, так как являются наиболее важными в курсе ИКТ. Авторам показалось неактуальным рассматривать все разделы, которые представлены в кодификаторе ЕГЭ по информатике, потому что некоторые из них ни разу за последние несколько лет не проверялись на экзамене. Такими разделами являются: «Социальная информатика», «Основные устройства информационных и коммуникационных технологий» и «Технология обработки текстовой информации» (этот раздел рассматривается в ГИА по информатике для 9 класса). Кроме этого, данное пособие является дидактическим сборником заданий, которое не заменяет учебник, а только дополняет его,

поэтому включение тем, не имеющих пока практических заданий в формате ЕГЭ, представляется нецелесообразным.

Сборник состоит из теоретической и практической частей. В теоретической части подробно не рассмотрен только раздел «Технологии программирования», хотя приведен список рекомендованной литературы. Все объяснения написаны на доступном языке, потому что в первую очередь рассчитаны на самообразование учащихся. В практической части даны примеры тематических заданий в формате ЕГЭ с сохранением всех типов и разновидностей, которые встречались в экзаменационных заданиях за последние 5 лет.

Каждый раздел сборника состоит из нескольких частей: часть 1 — это задания с выбором одного правильного ответа, часть 2 — задания с кратким ответом, часть 3 (которая присутствует не во всех разделах) — это задания с развернутым ответом.

По всем разделам даются ответы к заданиям.

Задания с развернутым ответом сопровождаются примерным содержанием правильного ответа и системой оценивания (описанием того, при каких условиях ответ можно считать полностью или частично выполненным и какого балла ответ заслуживает).

Советуем сначала прочитать один раздел из теоретической части, затем выполнить по нему задания из практической части, проверить, а потом, при анализе ошибок, снова вернуться к теории.

Специфика данного сборника заключается в тематическом структурировании заданий, позволяющем использовать их при текущей тематической проверке (самопроверке) знаний по мере изучения отдельных элементов содержания курсов информатики и ИКТ. Именно это обеспечивает систематическую и плодотворную подготовку к ЕГЭ.

Вместе с тем при непосредственной подготовке к экзамену рекомендуем использовать наряду с этим пособием сборники, представляющие собой цельные варианты КИМ, подобные вариантам экзаменационной работы ЕГЭ.

Надеемся, что эта книга окажет вам существенную помощь при подготовке к единому экзамену.

Желаем успеха!

ИНФОРМАЦИЯ И ЕЕ КОДИРОВАНИЕ

Понятие «**информация**» в различных областях знаний понимается по-разному. Например, по определению К. Шеннона, информация — это снятая неопределенность (такой подход позволил количественно измерять информацию). В кибернетике, по мнению Н. Винера, информация — это знания, которые участвуют в управлении. В информатике термин «информация» принято понимать как *меру уменьшения неопределенности* нашего знания о состоянии какого-либо объекта или системы (выражена формулой Хартли, которая приведена ниже).

Свойства информации:

- объективность;
- достоверность;
- полнота;
- актуальность;
- полезность;
- понятность.

Основные информационные процессы:

- обработка;
- использование;
- поиск;
- сбор;
- хранение;
- получение;
- передача;
- защита.

Передача и представление информации могут осуществляться с помощью языков, которые являются знаковыми системами. Каждая знаковая система строится на основе опреде-

ленного алфавита и правил выполнения операций над знаками. Например, молекулу ДНК можно рассматривать как знак биологического алфавита, а цифры 1 и 0 как знаки алфавита двоичной системы счисления.

Один символ — буква, цифра, знак препинания, пробел, математический или графический символ.

В кодах ASCII:

$$1 \text{ символ} = 8 \text{ бит} = 1 \text{ байт.}$$

В международном стандарте Unicode:

$$1 \text{ символ} = 16 \text{ бит} = 2 \text{ байта.}$$

Например, слово «информатика» содержит 11 символов и в кодировке ASCII или КОИ-8 будет иметь объем 88 бит, т.е. 11 байт, а в кодировке Unicode — 22 байта.

Существует формула Хартли, которая имеет следующий вид:

$$n = p^i,$$

где n — число равновероятных событий, i — количество информации, полученной в результате совершения события, p — количество возможных вариантов. Или, адаптируя эту формулу под условия задач, можно сказать, что p — количество используемых символов, а i — длина строки символов или сигналов.

Например, чтобы узнать, сколько различных символов длиной 5 сигналов можно закодировать с помощью кода Морзе, надо преобразовать формулу:

p — станет равно 2, потому что в азбуке Морзе используются только точки и тире, а i — равным 5, потому что такова длина сигнала. Следовательно, $n=2^5$, т.е. 32.

Таблица степеней числа 2

2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1024	512	256	128	64	32	16	8	4	2	1

Таблица степеней числа 2 написана не так, как принято математически, т.е. слева направо, а наоборот, потому что так

проще ею пользоваться при переводе из двоичной системы в десятичную.

Кратные единицы измерения количества информации:

1 байт = 8 бит

1 килобайт = $1024 \text{ б} = 2^{10} \text{ б}$

1 мегабайт = $1024 \text{ кб} = 2^{10} \text{ кб}$

1 гигабайт = $1024 \text{ мб} = 2^{10} \text{ мб}$

1 терабайт = $1024 \text{ гб} = 2^{10} \text{ гб}$

1 эксобайт = $1024 \text{ тб} = 2^{10} \text{ тб}$

Для определения скорости передачи информации используются другие единицы измерения информации:

Бит/сек

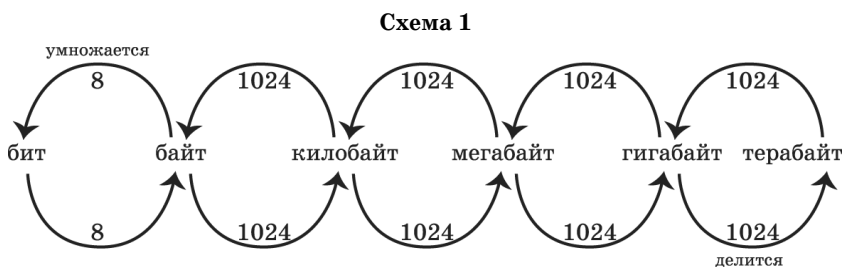
1 килобит = $2^{10} \text{ бит} = 1024 \text{ бит} = 128 \text{ байт}$

1 мегабит = $2^{10} \text{ кбит} = 128 \text{ килобайт}$

1 гигабит = $2^{10} \text{ мбит} = 128 \text{ мегабайт}$

Например, чтобы определить, какого размера файл мы можем передать, если знаем скорость интернет-соединения и время, надо умножить скорость на время, а потом по приведенной ниже схеме перевести в нужные по заданию единицы измерения информации.

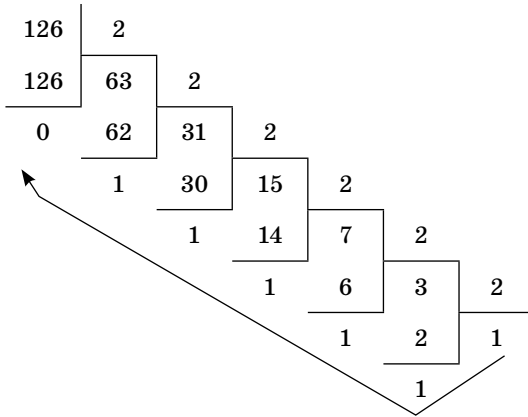
Последовательность действий при переводе одних единиц измерения информации в другие приведена на схеме 1.



Значащими нулями называются нули, стоящие справа от первой единицы.

Для перевода десятичного числа в двоичное можно использовать два способа.

1-й способ (деление на основание системы счисления)



Собираем остатки (единицы и нули) снизу вверх по стрелке. $126_{10} = 1111110_2$. Таким образом, ответ формулируется так: в двоичной записи десятичного числа 126 имеется только 1 значащий ноль.

2-й способ (метод «разностей»)

$$126_{10} = N_2$$

Надо найти по таблице степеней числа 2 (которая приведена выше) самое большое число, меньшее или равное заданному числу 126. Это 64. Вычитаем: $126 - 64 = 62$. Начинаем заполнять таблицу, выписав все степени двойки до 2^6 . Если используем данную степень, то ставим единицу, а если нет, то ноль. Теперь ищем степень двойки меньше или равную уже 62. Это 32. $62 - 32 = 30$. Степень меньше 30, это 16. $30 - 16 = 14$. $14 - 8 = 6$. $6 - 4 = 2$. $2 - 2 = 0$. Значит, в таблице под степенью 2^0 будет стоять ноль. Следовательно, $126_{10} = 1111110_2$. Количество значащих нулей в записи числа 126 равно 1.

64	32	16	8	4	2	1
1	1	1	1	1	1	0

Алфавит восьмеричной системы счисления состоит только из цифр 0, 1, 2, 3, 4, 5, 6, 7.

Таблица степеней числа 8

8^5	8^4	8^3	8^2	8^1	8^0
32768	4096	512	64	8	1

Алфавит десятичной системы счисления состоит только из цифр 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

Таблица степеней числа 10

10^5	10^4	10^3	10^2	10^1	10^0
100000	10000	1000	100	10	1

Алфавит шестнадцатеричной системы счисления состоит из цифр 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 и букв латинского алфавита, которые заменяют числа от 10 до 15: A(10), B(11), C(12), D(13), E(14), F(15).

Таблица степеней числа 16

16^5	16^4	16^3	16^2	16^1	16^0
1048576	65536	4096	256	16	1

Перевод чисел по схеме $N_8 \rightarrow N_2$

Каждая цифра заменяется триадой (триада — три разряда).

Например, переведем число 72 из восьмеричной системы в двоичную. Каждый разряд восьмеричного числа кодируется 3 двоичными разрядами.

Таким образом, $72_8 = 111010_2$.

$\underbrace{111}_7 \quad \underbrace{010}_2$

Перевод чисел по схеме $N_{16} \rightarrow N_2$

Каждая цифра заменяется тетрадой (тетрада — четыре разряда).

Например, переведем число 1D из шестнадцатеричной системы в двоичную. Каждый разряд шестнадцатеричного числа кодируется 4 двоичными разрядами. Таким образом:

$1D_{16} = 11101_2$.

$\underbrace{0001}_1 \quad \underbrace{1101}_D$

Перевод чисел по схеме $N_q \rightarrow N_{10}$ (где q — основание системы)

Например, переведем число 72 из восьмеричной системы в десятичную:

7	2
8^1	8^0

$$72_8 = 7 * 8 + 2 * 1 = 58_{10}.$$

Например, переведем число 1D из шестнадцатеричной системы в десятичную:

1	D
16^1	16^0

$$1D_{16} = 1 * 16 + 13 * 1 = 29_{10}.$$

При таком переводе следует помнить, что любое число в нулевой степени равно единице, а в первой степени — самому себе.

Теперь сложим получившиеся десятичные числа:

$$58 + 29 = 87_{10}.$$

А затем по любому из представленных выше способов переведем это число в двоичную систему. То есть $87_{10} = 1010111_2$.

Перевод чисел по схеме $N_2 \rightarrow N_{10} \rightarrow N_8 \rightarrow N_{16}$

Возьмем двоичное число: 10111011_2 .

Чтобы перевести его в десятичную систему счисления, надо под каждой цифрой справа налево выписать двоичные разряды и сложить те, которые окажутся под единицами:

1	0	1	1	1	0	1	1
128	64	32	16	8	4	2	1

$$128 + 32 + 16 + 8 + 2 + 1 = 187_{10}.$$

Теперь, чтобы перевести двоичное число в восьмеричную систему счисления, надо разбить его на триады (по три разряда) справа налево, подписать степени двойки отдельно под

каждой триадой и так же отдельно сложить те разряды, которые окажутся под единицами, а потом просто выписать числа слева направо:

	1	0	1	1	1	0	1	1
4	2	1	4	2	1	4	2	1
2			7			3		

$$10111011_2 = 273_8.$$

Аналогичным способом переводят и в шестнадцатеричную систему, вот только разбивают на тетрады (по четыре разряда), потому что 16 — четвертая степень двойки:

1	0	1	1	1	0	1	1
8	4	2	1	8	4	2	1
В (11)				В (11)			

$$10111011_2 = \text{ВВ}_{16}.$$

Если тетрада или триада будет равна нулю, то ее все равно необходимо указывать в записи числа.

Правила двоичного сложения

$$0 + 0 = 0 \quad 0 + 1 = 1 \quad 1 + 0 = 1$$

$1 + 1 = 10$ (результат сложения двух единиц: *ноль и единица переноса* в старший разряд)

Складываем заданные числа в двоичной системе:

$$\begin{array}{r}
 111 \\
 + \quad 11101 \\
 \hline
 111010 \\
 \hline
 1010111
 \end{array}$$

Правила двоичного умножения

$$0 * 0 = 0 \quad 0 * 1 = 0 \quad 1 * 0 = 0 \quad 1 * 1 = 1$$

Соблюдаются правила арифметики при самом умножении, а при сложении получившихся чисел надо использовать правила сложения двоичных чисел.

Работа в системах счисления с основанием N

Например, надо определить все основания систем счисления, в которых запись числа 22 оканчивается на 4.

Для этого можно представить задание в виде уравнения:

$$x4_n = 22_{10},$$

где n — основание системы счисления, а x — вторая цифра в записи числа.

Для того чтобы определить наибольшее основание системы, необходимо x приравнять к единице и развернуто записать число, получившееся слева:

$$n^1 * 1 + n^0 * 4 = 22_{10}.$$

Известно, что всегда $n^0 = 1$, а $n^1 = n$. Преобразуем уравнение:

$$\begin{aligned} n + 4 &= 22_{10} \\ n &= 18. \end{aligned}$$

Итак, мы нашли наибольшее основание системы:

$$14_{18} = 22_{10}.$$

Теперь, чтобы определить другие возможные основания, нам необходимо найти сомножители полученного числа 18:

$$\begin{aligned} 2 * 9 &= 18 \\ 3 * 6 &= 18. \end{aligned}$$

Затем проанализируем полученные числа на соответствие условиям задачи. Известно, что в n -ной системе счисления могут использоваться только числа на единицу меньше основания n . И хотя в системах счисления, где основание больше десяти, используются буквы, но они также эквивалентны цифрам и подчиняются тому же правилу. Исходя из этого и условий задачи, мы можем сказать, что нам не подойдут основания систем счисления, меньшие или равные 4, так как эта

цифра используется в задании. Поэтому из четырех полученных сомножителей числа 18 нам не подойдут 2 и 3.

Теперь мы уже знаем ответ: 6, 9, 18.

Немного отличается задача, где надо указать все десятичные числа, не превосходящие 25, запись которых в системе счисления с основанием четыре оканчивается на 11, но принцип остается тот же.

Необходимо составить таблицу степеней числа 4 до той степени, пока она не будет превосходить 25.

4^3	4^2	4^1	4^0
64	16	4	1

Составим уравнение, подобное тому, что приведено выше, чтобы определить возможные десятичные числа:

$$x11_4 = y (<=25).$$

В системе счисления с основанием 4 могут использоваться только цифры 0, 1, 2, 3, поэтому x может быть равен любой из этих цифр.

Проверим сначала ноль. Тогда уравнение примет вид:

$$11_4 = 4^0 * 1 + 4^1 * 1 = 5.$$

Этот вариант подходит. Затем цифру один:

$$111_4 = 4^0 * 1 + 4^1 * 1 + 4^2 * 1 = 21.$$

Это тоже верное решение. А вот уже цифра два нам даст число 37, что не отвечает условиям задачи. По этой же причине не подходит и цифра 3.

ЗАДАНИЯ

Часть 1

При выполнении заданий этой части необходимо выбрать один верный ответ из представленных четырех вариантов и поставить знак «х» в клеточку, номер которой соответствует номеру выбранного вами ответа.

1. Автоматическое устройство осуществило перекодировку двух информационных сообщений равной длины из 8-битной кодировки ASCII в 16-битную кодировку Unicode в одно новое сообщение, которое при этом увеличилось на 240 байт. Какова длина каждого из исходных сообщений в символах?
 - 1) 120
 - 2) 230
 - 3) 460
 - 4) 920
2. Автоматическое устройство осуществило перекодировку информационного сообщения из 16-битной кодировки Unicode в 8-битную кодировку КОИ-8. При этом информационный объем сообщения уменьшился на 16 байтов. Сколько бит было в первоначальном сообщении?
 - 1) 512
 - 2) 374
 - 3) 256
 - 4) 128
3. Автоматическое устройство осуществило перекодировку информационного сообщения. При этом информационный объем сообщения увеличился в 2 раза. В какой кодировке стало записано это сообщение?
 - 1) КОИ-8
 - 2) ASCII
 - 3) Unicode
 - 4) Windows
4. Автоматическое устройство осуществило перекодировку информационного сообщения из 8-битной кодировки ASCII в два равных по длине информационных сообщения в 16-битной кодировке Unicode. При этом информационный объем каждого нового сообщения увеличился на 8 байт. Какова длина первоначального сообщения в байтах?
 - 1) 8
 - 2) 9
 - 3) 16
 - 4) 24

5. Автоматическое устройство осуществило перекодировку информационного сообщения из 16-битной кодировки Unicode в 8-битную кодировку КОИ-8 и потом удалило половину сообщения. После этого информационный объем сообщения стал равен 2^4 байт. Сколько бит было в первоначальном сообщении?

1) 64
2) 128
3) 256
4) 512

6. Считая, что каждый символ кодируется 2 байтами, оцените объем следующего предложения в кодировке Unicode:

Компьютер — универсальное средство обработки информации.

1) 110 символов
2) 124 байта
3) 896 бит
4) 1024 бита

7. Считая, что каждый символ кодируется 1 байтом, оцените объем следующего предложения в кодировке ASCII:

`http://www.fipi.ru`

1) 264 бита
2) 144 бита
3) 36 байт
4) 14 байт

8. В кодировке КОИ-8 каждый символ кодируется 8 битами. Определите информационный объем следующего предложения:

Mail.ru — почтовый сервер.

1) 200 бит
2) 64 бита
3) 30 байт
4) 26 байт

9. В кодировке Unicode каждый символ кодируется 16 битами. Определите информационный объем следующего предложения:

*Нет повести печальнее на свете,
Чем повесть о Ромео и Джульетте.*

- | | |
|------------|-------------|
| 1) 512 бит | 3) 128 байт |
| 2) 256 бит | 4) 64 байта |

10. В кодировке Unicode на каждый символ отводится 2 байта. Определите информационный объем слова из 17 символов в этой кодировке:

- 1) 272 бита
- 2) 136 бит
- 3) 128 бит
- 4) 114 бит

11. В марафоне участвуют 12 спортсменов. Специальное устройство регистрирует прохождение финиша, записывая его номер с использованием минимально возможного количества бит, одинакового для каждого бегуна. Каков информационный объем сообщения, записанного устройством, после того как финиш пересекли $\frac{2}{3}$ спортсменов?

- 1) 6 байт
- 2) 32 бита
- 3) 3 байта
- 4) 48 бит

12. На олимпиаде участники получают баллы от 1 до 10 на каждом этапе, которые затем кодируются специальным устройством минимально возможным количеством бит для каждого участника. Каков информационный объем сообщения, записанного устройством, после того как 75 участников прошли по 3 этапа?

- 1) 900 бит
- 2) 300 байт
- 3) 256 бит
- 4) 110 байт

- 13.** 131 спортсмен участвовал в соревнованиях по толканию ядра. Все результаты записывались специальным автоматическим устройством с помощью минимально возможного количества бит, одинакового для всех спортсменов. Каков информационный объем сообщения, если известно, что наилучшим результатом было 37 метров?
- 1) 100 байт
 - 2) 126 байт
 - 3) 712 бит
 - 4) 786 бит
- 14.** Единый государственный экзамен сдавали 64 ученика. Максимальный балл, который можно было получить во время экзамена, — 100. Набранный балл (число, а не цифра) каждого ученика был закодирован с использованием двоичного кода минимально возможным количеством бит, равным для всех. Определите информационный объем сообщения, содержащего набранные баллы каждого ученика.
- 1) 450 бит
 - 2) 217 бит
 - 3) 112 байт
 - 4) 56 байт
- 15.** Для передачи секретного сообщения используются только десятичные цифры. При этом все цифры (не числа) кодируются одинаковым минимально возможным количеством бит. Определите информационный объем сообщения длиной в 140 символов.
- 1) 140 бит
 - 2) 556 бит
 - 3) 70 байт
 - 4) 64 байта
- 16.** Сколько мегабайт информации содержит сообщение объемом 2^{32} бит?
- | | |
|---------|--------|
| 1) 1024 | 3) 448 |
| 2) 512 | 4) 256 |

17. Сколько бит информации содержит сообщение объемом 128 килобайт?
1) 2^{16} 2) 2^{19} 3) 2^{20} 4) 2^{23}
18. В бассейне ведется наблюдение за температурой воды. Результатом одного измерения является целое число от +10 до +35 градусов, которое записывается при помощи минимально возможного количества бит. Было сделано 27 измерений. Определите информационный объем всех результатов наблюдения.
1) 54 байта
2) 65 байт
3) 128 бит
4) 162 бита
19. Метеорологическая станция ведет наблюдения за скоростью ветра. Результатом одного измерения является целое число от 1 до 20, которое записывается при помощи минимально возможного количества бит. Определите информационный объем всех результатов наблюдения за март.
1) 175 бит
2) 155 бит
3) 55 байт
4) 20 байт
20. В донорский пункт пришло 56 человек. Количество миллилитров, взятых у каждого донора (от 50 до 500), записали минимально возможным количеством бит. Определите информационный объем этих результатов.
1) 525 бит
2) 500 бит
3) 256 байт
4) 63 байта
21. Дано $a = 177_8$ и $b = 73_{16}$. Какое из чисел c , записанных в двоичной системе счисления, отвечает условию $a > c > b$?
1) 1111111_2
2) 1110110_2
3) 1101110_2
4) 1110010_2

22. Дано $a=461_8$ и $b=6B_{16}$. Какое из чисел c , записанных в двоичной системе счисления, отвечает условию $(a + b) > c$?
- 1) 110011100_2
 - 2) 110011101_2
 - 3) 110101010_2
 - 4) 110010100_2
23. Дано $a = A5_{16}$ и $b = 201_8$. Какое из чисел c , записанных в двоичной системе счисления, отвечает условию $a > (b + c)$?
- 1) 100011_2
 - 2) 101100_2
 - 3) 110001_2
 - 4) 101011_2
24. Дано $a = 48_{16}$ и $b = 72_8$. Какое из чисел c , записанных в десятичной системе счисления, отвечает условию $a > b > c$?
- 1) 47
 - 2) 58
 - 3) 61
 - 4) 72
25. Дано $a=31_{16}$ и $b=75_8$. Какое из чисел c , записанных в десятичной системе счисления, отвечает условию $a < c < b$?
- 1) 45
 - 2) 52
 - 3) 61
 - 4) 63
26. Сколько единиц в двоичной записи числа 371?
- 1) 7
 - 2) 6
 - 3) 3
 - 4) 4
27. Сколько единиц в двоичной записи шестнадцатеричного числа 7C?
- 1) 6
 - 2) 2
 - 3) 5
 - 4) 4

28. Количество значащих нулей в двоичной записи числа 261 равно
- | | |
|------|------|
| 1) 6 | 3) 3 |
| 2) 8 | 4) 5 |
29. Количество значащих нулей в двоичной записи восьмеричного числа 150 равно
- | | |
|------|------|
| 1) 5 | 3) 3 |
| 2) 6 | 4) 4 |
30. Количество значащих единиц в десятичной записи двоичного числа 1100101 равно
- 1) 1
 - 2) 2
 - 3) 3
 - 4) 4
31. Чему равна сумма чисел 110_8 и 110_2 ?
- 1) 110110_2
 - 2) 93_{10}
 - 3) 117_8
 - 4) $4E_{16}$
32. Чему равна сумма чисел 19_{16} и 31_8 ?
- 1) 110001_2
 - 2) 50_{10}
 - 3) 50_8
 - 4) 50_{16}
33. Чему равна сумма чисел 110_2 и 110_{16} ?
- 1) 1010100_2
 - 2) 220_{10}
 - 3) 426_8
 - 4) 120_{16}
34. Чему равно произведение чисел 22_8 и 11_2 ?
- 1) 110110_2
 - 2) 33_8
 - 3) 1100011_2
 - 4) 55_8

35. Значение выражения $31_{16} - 31_8$ равно

- 1) 25_{10}
- 2) 11000_2
- 3) 0
- 4) 1_{16}

36. Значение выражения $15_{16} + 7_8 + 9_{10}$ равно

- 1) 100101_2
- 2) 101010_2
- 3) 111100_2
- 4) 110011_2

37. Значение выражения $100_{16} + 100_8 + 100_2$ равно

- 1) 257_{10}
- 2) 300_{10}
- 3) 324_{10}
- 4) 341_{10}

38. Чему равна сумма чисел $16_{16} + 16_8$?

- 1) 32_8
- 2) 32_{16}
- 3) 44_8
- 4) 44_{16}

39. Чему равно выражение $100_{16} - 10_8 - 1_2$?

- 1) $5E_{16}$
- 2) $F7_{16}$
- 3) 1101101_2
- 4) 111110_2

40. Чему равно выражение $21_{16} + 11_8 - 1001_2$?

- 1) 22_{16}
- 2) 44_8
- 3) 33_{10}
- 4) 100000_2

41. Для кодирования букв А, Б, В, Г решили использовать двухразрядные последовательные двоичные числа (от 00 до 11 соответственно). Если таким способом закодировать

последовательность символов ГБАВ и записать результат шестнадцатеричным кодом, то получится

- 1) C3
- 2) AE
- 3) 7F
- 4) D2

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

- 1) 5A
- 2) 8C
- 3) F1
- 4) AA

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

- 1) 55
- 2) 125
- 3) 77
- 4) 265

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

- 1) 8C
- 2) D8
- 3) D0
- 4) AC

45. Для кодирования букв А, Б, В, Г решили использовать двухразрядные последовательные двоичные числа (от 00 до 11 соответственно). Если таким способом закодировать

последовательность символов ГБАГ и записать результат восьмеричным кодом, то получится

- 1) 239
- 2) 232
- 3) 330
- 4) 323

46. Для кодирования букв А, В, С, D решили использовать двухразрядные последовательные двоичные числа (от 00 до 11 соответственно). Если таким способом закодировать последовательность символов CADB и записать результат шестнадцатеричным кодом, то получится

- 1) 8D
- 2) 98
- 3) C5
- 4) CD

47. Для кодирования символов &, @, :, ! решили использовать двухразрядные последовательные двоичные числа (от 00 до 11 соответственно). Если таким способом закодировать последовательность символов &: @!@ и записать результат шестнадцатеричным кодом, то получится

- 1) D8
- 2) CD
- 3) 9D
- 4) 9C

48. Для кодирования символов +, −, *, : решили использовать двухразрядные последовательные двоичные числа (от 00 до 11 соответственно). Если таким способом закодировать последовательность символов * +−: и записать результат восьмеричным кодом, то получится

- 1) 198
- 2) 207
- 3) 305
- 4) 220

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

К	М	А	Р	И
000	10	111	01	101

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

- 1) 11001011011
- 2) 0001011111
- 3) 0111110000111
- 4) 100011101

50. 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	Д	Е
110	011	10	01	101

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

- 1) 001011010
- 2) 11010110101
- 3) 11011111010
- 4) 101100010

Часть 2

Ответом к заданиям этой части является набор символов, которые следует записать справа от номера соответствующего задания начиная с первой клеточки. Каждый символ пишете в отдельной клеточке в соответствии с приведенными образцами (если имеются бланки ЕГЭ).

51. Световое табло состоит из лампочек. Каждая лампочка может находиться в одном из трех состояний («включено», «выключено» или «мигает»). Какое наименьшее

количество лампочек должно находиться на табло, чтобы с его помощью можно было передать 14 различных сигналов?

52. Световое табло состоит из лампочек. Каждая лампочка может находиться в одном из трех состояний («включено», «выключено» или «мигает»). Какое наименьшее количество лампочек должно находиться на табло, чтобы с его помощью можно было передать 29 различных сигналов?
53. Световое табло состоит из лампочек. Каждая лампочка может находиться в одном из двух состояний («включено» или «выключено»). Какое наименьшее количество лампочек должно находиться на табло, чтобы с его помощью можно было передать 44 различных сигнала?
54. Световое табло состоит из лампочек. Каждая лампочка может находиться в одном из четырех состояний («красный», «зеленый», «синий» или «желтый»). Какое наименьшее количество лампочек должно находиться на табло, чтобы с его помощью можно было передать 11 различных сигналов?
55. Световое табло состоит из лампочек. Каждая лампочка может находиться в одном из трех состояний («горит в полную силу», «горит в полсилы» или «не горит»). Какое наименьшее количество лампочек должно находиться на табло, чтобы с его помощью можно было передать 15 различных сигналов?
56. В азбуке Морзе для кодирования символов используются только «точки» или «тире». Сколько различных символов можно закодировать, используя код Морзе длиной не менее 4 и не более 6 символов?
57. Римская система счисления вместо цифр использует буквы латинского алфавита «I», «V», «X» и «L». Сколько различных (необязательно правильных) чисел длиной от одной до 4 букв можно было бы составить?

58. Римская система счисления вместо цифр использует буквы латинского алфавита «Х», «L» и «М». Сколько различных (необязательно правильных) чисел длиной от 2 до 5 букв можно было бы составить?
59. Письменность неизвестного племени состоит всего из двух знаков: «круг» и «квадрат». Сколько слов может содержать этот язык, если известно, что все слова в нем не короче 3 и не длиннее 6 символов?
60. Для кодирования символов на маяке используют 3 вида сигналов: «красный», «зеленый» и «желтый». Сколько различных символов длиной не менее двух и не более 4 символов можно закодировать с помощью этих сигналов?
61. Укажите через запятую в порядке возрастания все десятичные числа, не превосходящие 15, запись которых в системе счисления с основанием три оканчивается на 11.
62. Укажите через запятую в порядке возрастания все десятичные числа, не превосходящие 40, запись которых в системе счисления с основанием четыре оканчивается на 12.
63. Укажите через запятую в порядке возрастания все десятичные числа, не превосходящие 70, запись которых в системе счисления с основанием пять оканчивается на 23.
64. Укажите через запятую в порядке возрастания все десятичные числа, не превосходящие 75, запись которых в системе счисления с основанием шесть оканчивается на 55.
65. Укажите через запятую в порядке возрастания все десятичные числа, не превосходящие 80, запись которых в системе счисления с основанием семь оканчивается на 30.
66. В системе счисления с некоторым основанием число 32 записывается в виде 112. Укажите это основание.
67. В системе счисления с некоторым основанием число 75 записывается в виде 203. Укажите это основание.



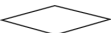

- 68.** Укажите через запятую в порядке возрастания все основания систем счисления, в которых запись числа 14 заканчивается на 4.
- 69.** Укажите через запятую в порядке возрастания все основания систем счисления, в которых запись числа 11 заканчивается на 5.
- 70.** Укажите через запятую в порядке возрастания все основания систем счисления, в которых запись числа 19 заканчивается на 4.
- 71.** Скорость передачи данных через аналоговый модем равна 128 000 бит/с. Через данное соединение передают файл размером 375 килобайт. Определите время передачи файла в секундах.
- 72.** Скорость передачи данных через спутниковый канал равна 256 000 бит/с. Через данное соединение передают файл размером 625 килобайт. Определите время передачи файла в секундах.
- 73.** Скорость передачи данных через WAP-соединение равна 512 000 бит/с. Через данное соединение передают файл размером 875 килобайт. Определите время передачи файла в секундах.
- 74.** Скорость передачи данных через аналоговый модем равна 131 072 бит/с. Через данное соединение передают файл размером 2 мегабайта. Определите время передачи файла в секундах.
- 75.** Скорость передачи данных через выделенный канал равна 16 384 байт/с. Через данное соединение передают файл размером 1 мегабайт. Определите время передачи файла в секундах.
- 76.** Скорость передачи данных через ADSL-соединение равна 256 000 бит/с. Передача файла через данное соединение заняла 3 минуты. Определите размер файла в килобайтах.

-
77. Скорость передачи данных через выделенный канал равна 512 000 бит/с. Передача файла через данное соединение заняла 46 секунд. Определите размер файла в килобайтах.
78. Скорость передачи данных через WAP-соединение равна 128 000 бит/с. Передача файла через данное соединение заняла 1 минуту и 20 секунд. Определите размер файла в килобайтах.
79. Известно, что длительность непрерывного подключения к сети Интернет с помощью модема для некоторых АТС не превышает 10 минут. Определите максимальный размер файла (в килобайтах), который может быть передан за время такого подключения, если модем передает информацию в среднем со скоростью 64 килобита/с ? (Впишите только число.)
80. Сколько секунд потребуется модему, передающему сообщение со скоростью 64000 бит/с, чтобы передать цветное растровое изображение размером 300×400 пикселей, при условии, что использована палитра с максимально возможным количеством цветов?

АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ

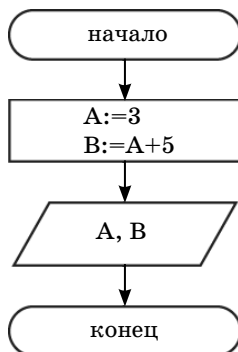
Блок-схема — графический способ описания алгоритма с использованием геометрических фигур.

На блок-схеме шаги алгоритма обозначаются специальными геометрическими фигурами, внутри которых записываются действия. Направление выполнения алгоритма обозначается стрелками.

	начало (конец) алгоритма
	действие
	проверка условия
	ввод или вывод

Линейный алгоритм

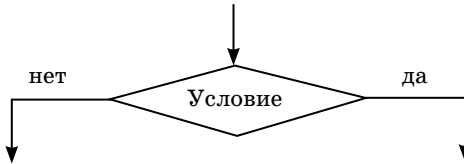
Алгоритм называется линейным, если он содержит несколько шагов и все шаги выполняются последовательно друг за другом от начала до конца.



Разветвляющийся алгоритм

Алгоритм называется разветвляющимся, если порядок выполнения шагов изменяется в зависимости от выполнения некоторых условий.

Если условие выполнено, то для решения задачи используется ветвь *да*, в противном случае — ветвь *нет*.



Циклический алгоритм

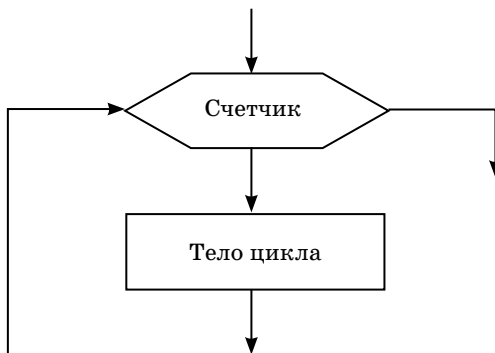
Алгоритм называется циклическим, если определенная последовательность шагов повторяется несколько раз в зависимости от заданной величины (параметра цикла).

Тело цикла — это шаги алгоритма, которые повторяются несколько раз.

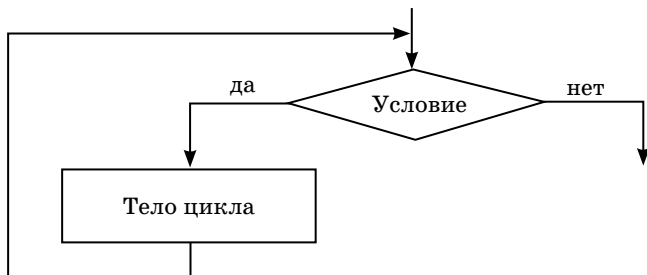
Параметр цикла — величина, от которой зависит *число повторений* в цикле.

Цикл заканчивается, когда параметр цикла принимает нужное значение.

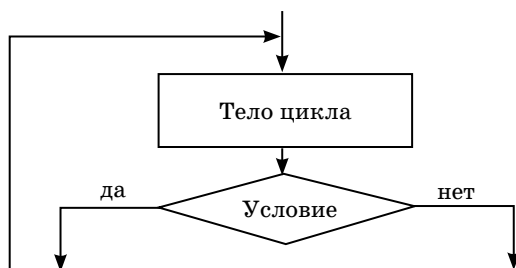
Цикл со счетчиком



Цикл с предусловием



Цикл с постусловием



Операция присваивания

Операция присваивания в блок-схемах, а также на языках Паскаль и Алгоритмический выглядит как $:=$, а на Бейсике записывается просто знаком равенства в форме:

Имя переменной = выражение.

Операция присваивания обозначает действие, во время которого в определенную ячейку памяти записывается значение, полученное в результате выполнения *выражения*.

Например, $K:=1$ означает, что число 1 запишется в ячейку с именем К.

В случае когда запись выглядит как $K:=K+1$, сначала вычисляется выражение, находящееся справа от знака равенства с последним имеющимся значением К, а потом итоговый результат присваивается переменной К, стирая предыдущее значение.

Операции div и mod

Операция \backslash (**div**) вычисляет результат деления нацело первого аргумента на второй, то есть:

$$7 \backslash 5 = 1 \quad 14 \text{ div } 5 = 2$$

Операция **MOD (mod)** вычисляет остаток от деления первого аргумента на второй, то есть:

$$7 \text{ MOD } 5 = 2 \quad 14 \text{ mod } 5 = 4$$

7	5
5	1
2	

\swarrow $7 \backslash 5$ \swarrow $7 \text{ mod } 5$

Операции \backslash и MOD (Бейсик), div и mod (Паскаль и Алгоритмический), — операции, вычисляющие результат деления нацело первого аргумента на второй и остаток от деления соответственно.

Работа с массивами

Массив — упорядоченный набор данных, обозначаемый одним именем.

Каждый элемент массива имеет свой порядковый номер. Массивы могут быть числовыми или строковыми. Массивы служат для хранения и обработки больших объемов информации. Создание массива начинается с объявления его типа и резервирования для него места в памяти компьютера.

Типы обрабатываемых данных

В основном массивы обрабатывают числовые (содержащие числа) и строковые (содержащие текст и символы) данные.

Одномерный массив

Например: $A[10]$ — одномерный массив, содержащий 10 ячеек, где $[i]$ — указывает на номер ячейки. Например, содержимое ячейки $A[5]$ будет равно 2.

1	2	3	4	5	6	7	8	9	10	Номера ячеек
0	5	4	9	2	4	6	4	7	4	Содержимое ячеек

Двумерный массив

Такой массив можно представить 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.

Например, содержимое ячейки $A[5,6]$ будет равно 1.

		j (номер столбца)						
		1	2	3	4	5	6	7
i (номер строки)	1	0	1	2	3	4	5	6
	2	-1	0	1	2	3	4	5
	3	-2	-1	0	1	2	3	4
	4	-3	-2	-1	0	1	2	3
	5	-4	-3	-2	-1	0	1	2
	6	-5	-4	-3	-2	-1	0	1
	7	-6	-5	-4	-3	-2	-1	0

Двумерный массив, так как происходит работа вложенных циклов, заполняется по строкам, то есть сначала последовательно заполняются ячейки первой строки (от $A[1,1]$ до $A[1,j]$), потом второй (от $A[2,1]$ до $A[2,j]$) и т.д. Считывание данных происходит в той же последовательности.

Поиск в массиве

Поиск в числовом массиве

В числовых массивах обычно производится поиск наименьшего, наибольшего или среднего значения элементов.

Например, для того чтобы найти наименьший элемент (то есть элемент с наименьшим значением), необходимо завести переменную `MIN` и присвоить ее нулю. Затем в цикле заполнить массив (если это не было сделано раньше), а потом в массиве со счетчиком сначала приравнять переменную `MIN` первому элементу массива. Следующий шаг — последовательно в этом же цикле сравнить, является ли каждый следующий элемент меньше значения переменной `MIN`. Если какой-то элемент оказывается меньше, чем значение `MIN`, то переменной `MIN` присваивается это значение. В конце цикла переменная `MIN` будет содержать значение наименьшего элемента. Если есть необходимость запомнить и его индекс, то для этого следует завести дополнительную переменную, чье значение будет изменяться вместе с переменной `MIN`, но содержать именно индекс ячейки.

Подобным алгоритмом следует воспользоваться и для поиска наибольшего элемента массива.

Для поиска среднего значения элементов массива необходимо завести переменные `SUMM` и `SR`, заполнить массив, а затем в цикле сложить последовательно значение всех элементов массива, используя переменную `SUMM`. После окончания цикла необходимо найти среднее значение элементов (`SR`) путем деления значения переменной `SUMM` на количество элементов в массиве.

Поиск в строковом массиве

Поиск в строковом массиве обычно реализуется в форме поиска индекса элемента массива, совпадающего с заданным.

Например, строковый массив содержит алфавит, и необходимо определить, какой порядковый номер будет у заданной

буквы. В данном случае порядковый номер будет совпадать с индексом элемента, содержащего данную букву.

Сначала надо заполнить массив буквами по алфавиту и завести переменную N под индекс заданной буквы. С клавиатуры ввести букву, чей индекс мы будем искать в массиве, затем в цикле со счетчиком необходимо сопоставить заданную букву со всеми элементами и в случае обнаружения совпадения запомнить в переменной N индекс данного элемента.

Сортировка массива

Существует довольно много способов сортировки, но самым распространенным является так называемый «метод пузырька», смысл которого заключается в том, что минимальные (максимальные) элементы, «как пузырьки, всплывают на поверхность», так они перемещаются в начало массива, меняясь местами с другими элементами.

Рассмотрим задачу упорядочения (сортировки) числового массива по возрастанию значений его элементов. Пусть имеется целочисленный числовой массив, состоящий, например, из 10 элементов и заполненный случайными числами.

Далее в цикле необходимо найти минимальный элемент в массиве среди элементов с 1-го по 10-й. Теперь надо поменять найденный минимальный элемент с элементом с индексом 1 (стоящим на первом месте), то есть выполняем перестановку. Затем снова проводим поиск минимального элемента, но уже среди элементов со 2-го по 10-й и делаем перестановку.

Многokrатно повторяем процедуру поиска и перестановки минимального элемента среди оставшихся неупорядоченных элементов массива. Сортировка реализуется с помощью цикла со счетчиком, и количество повторений равно $N - 1$, где N — количество элементов в массиве. В результате массив упорядочивается.

По аналогичному алгоритму происходит сортировка по убыванию, только в этом случае осуществляется поиск максимального элемента массива.

Основные функции

Математика	Бейсик	Паскаль
x^2	X^2	<code>sqr(x)</code>
x^3	X^3	$x*x*x$ или $\exp(3*\ln(x))$
\sqrt{x}	<code>SQR(X)</code>	<code>sqrt(x)</code>
$\sqrt[3]{x}$	$X^{(1/3)}$	$\exp((1/3)*\ln(x))$
$\sin x$	<code>SIN(X)</code>	<code>sin(x)</code>
$\cos x$	<code>COS(X)</code>	<code>cos(x)</code>
$\operatorname{tg} x$	<code>TAN(X)</code>	$\sin(x)/\cos(x)$
$ x $	<code>ABS(X)</code>	<code>abs(x)</code>
$\ln x$	<code>LOG(X)</code>	<code>ln(x)</code>
e^x	<code>EXP(X)</code>	<code>exp(x)</code>

ЗАДАНИЯ

Часть 1

При выполнении заданий этой части необходимо выбрать один верный ответ из представленных четырех вариантов и поставить знак «х» в клеточку, номер которой соответствует номеру выбранного вами ответа.

1. Определите значение переменной **c** после выполнения следующего фрагмента программы:

Бейсик	Паскаль	Алгоритмический
$a = -3$ $a = -a - a$ $b = a + 2$ $c = -b + a*b$	$a := -3;$ $a := -a - a;$ $b := a + 2;$ $c := -b + a*b;$	$a := -3$ $a := -a - a$ $b := a + 2$ $c := -b + a*b$

1) 12

2) 32

3) 40

4) 48

2. Определите значение переменной **c** после выполнения следующего фрагмента программы:

Бейсик	Паскаль	Алгоритмический
$a = 4$ $a = a * (-a)$ $b = -a/2$ $c = (a + b) + 3$	$a := 4;$ $a := a * (-a);$ $b := -a/2;$ $c := (a + b) + 3;$	$a := 4$ $a := a * (-a)$ $b := -a/2$ $c := (a + b) + 3$

- 1) -5
- 2) 8
- 3) -12
- 4) 14

3. Определите значение переменной **c** после выполнения следующего фрагмента программы:

Бейсик	Паскаль	Алгоритмический
$a = 8$ $a = -5 + (a + 2)$ $b = (-a + a) * a$ $c = (b - a) * (2 - a)$	$a := 8;$ $a := -5 + (a + 2);$ $b := (-a + a) * a;$ $c := (b - a) * (2 - a);$	$a := 8$ $a := -5 + (a + 2)$ $b := (-a + a) * a$ $c := (b - a) * (2 - a)$

- 1) 15
- 2) 12
- 3) 10
- 4) -5

4. Определите значение переменной **c** после выполнения следующего фрагмента программы:

Бейсик	Паскаль	Алгоритмический
$a = -3$ $b = (a + 7) - a$ $b = b - a$ $c = (-b * a) - (a * a)$	$a := -3;$ $b := (a + 7) - a;$ $b := b - a;$ $c := (-b * a) - (a * a);$	$a := -3$ $b := (a + 7) - a$ $b := b - a$ $c := (-b * a) - (a * a)$

- 1) -31
- 2) -25
- 3) 21
- 4) 16

5. Определите значение переменной **c** после выполнения следующего фрагмента программы:

Бейсик	Паскаль	Алгоритмический
$a = 6$ $a = (a + 3) - a - a$ $b = 21 + a * a$ $c = b / (-a) + 5$	$a := 6;$ $a := (a + 3) - a - a;$ $b := 21 + a * a;$ $c := b / (-a) + 5;$	$a := 6$ $a := (a + 3) - a - a$ $b := 21 + a * a$ $c := b / (-a) + 5$

- 1) 25 2) 15 3) -15 4) -30

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

Бейсик	Паскаль	Алгоритмический
$a = 1235$ $b = (a \setminus 100) + 25$ $a = (a \bmod 100) - b$ (\ и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно)	$a := 1235;$ $b := (a \text{ div } 100) + 25;$ $a := (a \bmod 100) - b;$ {div и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно}	$a := 1235$ $b := (a \text{ div } 100) + 25$ $a := (a \bmod 100) - b$ (div и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно)

- 1) $a = -2, b = 37$
 2) $a = 37, b = 35$
 3) $a = 5, b = 123$
 4) $a = -12, b = 35$

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

Бейсик	Паскаль	Алгоритмический
$a = (20 * 100) + (3 * 7)$ $b = (a \setminus 1000) + (a \bmod 100)$ $a = (a - b) \bmod 10$	$a := (20 * 100) + (3 * 7);$ $b := (a \text{ div } 1000) + (a \bmod 100);$ $a := (a - b) \bmod 10;$	$a := (20 * 100) + (3 * 7)$ $b := (a \text{ div } 1000) + (a \bmod 100)$ $a := (a - b) \bmod 10$

Бейсик	Паскаль	Алгоритмический
(\ и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно)	{div и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно}	(div и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно)

1) $a = 19, b = 22$ 2) $a = 8, b = 23$ 3) $a = 9, b = 22$ 4) $a = 98, b = 24$

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

Бейсик	Паскаль	Алгоритмический
$a = 2521$ $b = ((a \setminus 100) * 101) \bmod 1000$ $a = (b - (b \bmod 10)) / 10$ (\ и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно)	$a := 2521;$ $b := ((a \text{ div } 100) * 101) \bmod 1000;$ $a := (b - (b \bmod 10)) / 10;$ {div и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно}	$a := 2521$ $b := ((a \text{ div } 100) * 101) \bmod 1000$ $a := (b - (b \bmod 10)) / 10$ (div и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно)

1) $a = 52, b = 252$ 2) $a = 52, b = 525$ 3) $a = 521, b = 525$ 4) $a = 25, b = 251$

9. Определите значение целочисленных переменных a и b после выполнения фрагмента программы:

Бейсик	Паскаль	Алгоритмический
$a = 2143$ $b = 7 + (a \setminus 100) * 10$ $a = (10 * b - a) \bmod 10$ (\ и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно)	$a := 2143;$ $b := 7 + (a \operatorname{div} 100) * 10;$ $a := (10 * b - a) \bmod 10;$ { div и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно}	$a := 2143$ $b := 7 + (a \operatorname{div} 100) * 10$ $a := (10 * b - a) \bmod 10$ (div и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно)

1) $a = 210, b = 27$

2) $a = 27, b = 210$

3) $a = 7, b = 217$

4) $a = 43, b = 270$

10. Определите значение целочисленных переменных a и b после выполнения фрагмента программы:

Бейсик	Паскаль	Алгоритмический
$a = 5010$ $b = (a \bmod 100) * 11$ $a = (a + (b \setminus 10)) \setminus 10$ (\ и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно)	$a := 5010;$ $b := (a \bmod 100) * 11;$ $a := (a + (b \operatorname{div} 10)) \operatorname{div} 10;$ { div и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно}	$a := 5010$ $b := (a \bmod 100) * 11$ $a := (a + (b \operatorname{div} 10)) \operatorname{div} 10$ (div и mod — операции, вычисляющие результаты деления нацело первого аргумента на второй и остаток от деления соответственно)

1) $a = 502, b = 110$

2) $a = 52, b = 121$

3) $a = 501, b = 110$

4) $a = 151, b = 521$

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

Бейсик	Паскаль	Алгоритмический
<pre> k:=2 FOR i=1 TO n b=A(i,i) A(i,i) = A(i,k) A(i,k) = b NEXT i </pre>	<pre> k:=2; for i:=1 to n do begin b:=A[i,i]; A[i,i]:= A[i,k]; A[i,k]:= b end; </pre>	<pre> k:=2 нц для i от 1 до n b:=A[i,i] A[i,i]:= A[i,k] A[i,k]:= b кц </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) элементы диагонали и k -й строки таблицы
- 2) две строки в таблице
- 3) элементы диагонали и k -го столбца таблицы
- 4) два столбца в таблице

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

Бейсик	Паскаль	Алгоритмический
<pre> k:=2 FOR i=1 TO n b=A(i,i) A(i,i) = A(k, i) A(k, i) = b NEXT i </pre>	<pre> k:=2; for i:=1 to n do begin b:=A[i,i]; A[i,i]:= A[k,i]; A[k, i]:= b end; </pre>	<pre> k:=2 нц для i от 1 до n b:=A[i,i] A[i,i]:= A[k,i] A[k, i]:= b кц </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 после исполнения данного алгоритма в массиве изменяют значения:

- 1) только элементы диагонали
- 2) элементы диагонали и k -й строки
- 3) все элементы массива
- 4) элементы диагонали и k -го столбца таблицы

13. Дан фрагмент программы, обрабатывающей двумерный массив A размера 5×5 .

Бейсик	Паскаль	Алгоритмический
<pre>k=5 FOR i=1 TO 5 c=A(i,i) A(i,i) = A(k, i) A(k, i) = c k=k - 1 NEXT i</pre>	<pre>k:=5; for i:=1 to 5 do begin c:=A[i,i]; A[i,i]:= A[k, i]; A[k, i]:= c; k:=k - 1 end;</pre>	<pre>k:=5 нц для i от 1 до 5 c:=A[i,i] A[i,i]:= A[k, i] A[k, i]:= c k:=k - 1 кц</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) две строки
- 2) элементы диагонали и k -й строки
- 3) два столбца
- 4) элементы двух диагоналей

14. Дан фрагмент программы, обрабатывающей два массива $A[1..10]$ и $B[1..10]$.

Бейсик	Паскаль	Алгоритмический
<pre>k=10 FOR i=1 TO 10 A(i)=k + 1 B(k) = A(i) k=k - 1 NEXT i</pre>	<pre>k:=10; for i:=1 to 10 do begin A[i]:=k + 1; B[k]:= A[i]; k:=k - 1 end;</pre>	<pre>k:=10 нц для i от 1 до 10 A[i]:= k + 1 B[k]:= A[i] k:=k - 1 кц</pre>

Представим массив в виде строк длиной в 10 ячеек, в которой величина i является номером ячейки. Тогда данный алгоритм:

- 1) меняет местами элементы двух массивов
- 2) заполняет массив А с начала, а массив В с конца
- 3) меняет местами элементы массивов, стоящие на четных местах
- 4) заполняет массив А четными числами, а массив В — нечетными

15. Дан фрагмент программы, обрабатывающей два массива $A[1..10]$ и $B[1..10]$.

Бейсик	Паскаль	Алгоритмический
<pre> k:=2 FOR i=1 TO 10 A(i)=i + 1 NEXT i FOR i=1TO5 B(k) = A(11 - k) k=k + 2 NEXT i </pre>	<pre> k:=2; for i:=1 to 10 do A[i]:=i+1; for i:=1to5do begin A[i]:=i + 1; B[k]:= A[11 - k]; k:=k + 2 end; </pre>	<pre> k:=2 нц для i от 1 до 10 A[i]:= i +1 кц нц для i от 1 до 5 B[k]:= A[11 - k] k:=k + 2 кц </pre>

Представим массив в виде строк длиной в 10 ячеек, в которой величина i является номером ячейки. Тогда данный алгоритм:

- 1) заполняет четные ячейки массива В значениями нечетных ячеек массива А
- 2) меняет местами первые 5 элементов массива А с последними 5 массива В
- 3) заполняет массив В элементами массива А в обратном порядке
- 4) меняет местами первый элемент массива А с последним массива В

16. Значения двумерного массива А размера 7×7 задаются с помощью вложенного оператора цикла в представленном фрагменте программы:

Бейсик	Паскаль	Алгоритмический
FOR i=1 TO 7 FOR j=1 TO 7 A(i,j)=i*j - j NEXT j NEXT i	for i:=1 to 7 do for j:=1 to 7 do A[i,j]:=i*j - j;	<u>нц для i от 1 до 7</u> <u>нц для j от 1 до 7</u> A[i,j]:= i*j - j <u>кц</u> <u>кц</u>

Сколько строк в массиве будут иметь нулевые значения?

- 1) одна
- 2) две
- 3) три
- 4) ни одной

17. Значения двумерного массива А размера 5×5 задаются с помощью вложенного оператора цикла в представленном фрагменте программы:

Бейсик	Паскаль	Алгоритмический
FOR i=1 TO 5 FOR j=1 TO 5 A(i,j)= j - i A(j,i)= i NEXT j NEXT i	for i:=1 to 5 do for j:=1 to 5 do begin A[i,j]:= j - i; A[j,i]:=i end;	<u>нц для i от 1 до 5</u> <u>нц для j от 1 до 5</u> A[i,j]:= j - i A[j,i]:= i <u>кц</u> <u>кц</u>

Сколько в массиве столбцов с одинаковыми значениями?

- 1) один
- 2) два
- 3) ни одного
- 4) четыре

18. Значения двумерного массива А размера 5×5 задаются с помощью вложенного оператора цикла в представленном фрагменте программы:

Бейсик	Паскаль	Алгоритмический
FOR n=1 TO 5 FOR k=1 TO 5 A(k,n)=(k - n)*n NEXT k NEXT n	for n:=1 to 5 do for k:=1 to 5 do A[k,n]:= (k - n)*n;	<u>нц для n от 1 до 5</u> <u>нц для k от 1 до 5</u> A[k,n]:= (k - n)*n <u>кц</u> <u>кц</u>

Сколько элементов массива будут иметь положительные значения?

- 1) ни одного
- 2) 10
- 3) 15
- 4) 25

19. Значения двух массивов $A[1..10]$ и $B[1..10]$ задаются с помощью следующего фрагмента программы:

Бейсик	Паскаль	Алгоритмический
<pre>FOR i=1 TO 10 A(i)=10 - i NEXT i FOR i=1 TO 10 B(i)=A(5) - A(i) NEXT i</pre>	<pre>for i:=1 to 10 do A[i]:= 10 - i; for i:=1 to 10 do B[i]:= A[5] - A[i];</pre>	<pre><u>нц для i от 1 до 10</u> A[i]:= 10 - i <u>кц</u> <u>нц для i от 1 до 10</u> B[i]:= A[5] - A[i] <u>кц</u></pre>

Сколько элементов массива будут иметь отрицательные значения?

- 1) 0
- 2) 5
- 3) 10
- 4) 4

20. Значения двух массивов $A[1..10]$ и $B[1..10]$ задаются с помощью следующего фрагмента программы:

Бейсик	Паскаль	Алгоритмический
<pre>FOR n=1 TO 10 A(n)= -n - n B(11 - n)=A(n) NEXT n FOR n=1 TO 10 A(n)=B(n) + 10 NEXT n</pre>	<pre>for n:=1 to 10 do begin A[n]:= -n - n; B[11 - n]:= A[n] end; for n:=1 to 10 do A[n]:= B[n] + 10;</pre>	<pre><u>нц для n от 1 до 10</u> A[n]:= -n - n B[11 - n]:=A[n] <u>кц</u> <u>нц для n от 1 до 10</u> A[n]:= B[n] + 10 <u>кц</u></pre>

Чему равен элемент массива $A[8]$?

- 1) 0
- 2) -16
- 3) -6
- 4) 4

- 21.** Цепочка из трех бусин, помеченных латинскими буквами, формируется по следующему правилу. В конце цепочки стоит одна из бусин А, В, С. На первом месте — одна из бусин С, D, Е, которой нет в середине. А в середине — одна из бусин А, В, Е, D, не стоящая на третьем месте. Какая из перечисленных цепочек создана по этому правилу?
- 1) АВА
 - 2) CCC
 - 3) DAC
 - 4) CDE
- 22.** Для составления цепочек использованы буквы: А, В, М, О, Р. В середине цепочки стоит одна из бусин М, В, А. На третьем — любая гласная, если первая буква согласная, и любая согласная, если первая буква гласная. На первом месте — одна из бусин Р, О, М, А, не стоящая в середине или в конце цепочки. Какая из перечисленных цепочек создана по этому правилу?
- 1) OBP
 - 2) POP
 - 3) MMB
 - 4) MAV
- 23.** Для составления цепочек разрешается использовать бусины 5 типов, обозначаемых А, Б, О, К, И. Каждая цепочка должна состоять из трех бусин, при этом должны соблюдаться следующие правила:
- 1) На первом месте стоит одна из гласных букв.
 - 2) После гласной буквы в цепочке не может снова идти гласная, а после согласной — согласная.
 - 3) Последней буквой не может быть согласная буква.
- Какая из перечисленных цепочек создана по этому правилу?
- 1) АОБ
 - 2) БОК
 - 3) ОКИ
 - 4) ИКБ

24. В формировании цепочки из четырех бусин используются некоторые правила. В конце цепочки стоит одна из бусин К, Р, О, Т. На первом месте — одна из бусин К, О, Т, М, которой нет на последнем месте. На третьем месте — одна из бусин К, Р, если первая буква гласная, или одна из бусин М, Т, если первая буква согласная. Какая из перечисленных цепочек могла быть создана с учетом этих правил?

1) РМТО

3) ОТТО

2) МАКТ

4) ТКТО

25. Для составления цепочек используются бусины, помеченные буквами А, Б, Д, Е, К, Л, по следующему правилу. На первом месте в цепочке стоит одна из бусин А, Д, К, Л. На втором — одна из бусин А, Б, Е, Д, не стоящая в цепочке на первом или последнем месте. На третьем — любая гласная, если первая буква гласная, и любая согласная, если первая буква согласная. На четвертом месте — согласная буква, не стоящая на первом или втором месте. Какая из перечисленных цепочек создана по этому правилу?

1) БДАЕ

3) ЛКАД

2) АБЕЛ

4) КУАЛ

26. 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.

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

НАЧАЛО

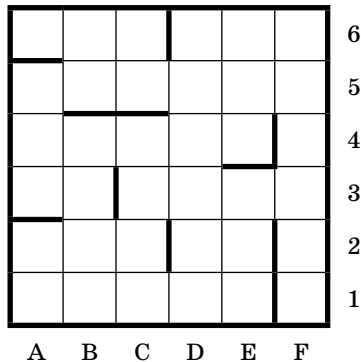
ПОКА < справа свободно > **вправо**

ПОКА < сверху свободно > **вверх**

ПОКА < слева свободно > **влево**

ПОКА < снизу свободно > **вниз**

КОНЕЦ



1) 1

2) 2

3) 3

4) 4

27. 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.

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

НАЧАЛО

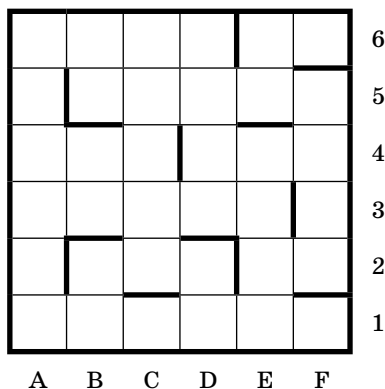
ПОКА < *слева свободно* > *влево*

ПОКА < *сверху свободно* > *вверх*

ПОКА < *справа свободно* > *вправо*

ПОКА < *снизу свободно* > *вниз*

КОНЕЦ



1) 1

2) 2

3) 3

4) 0

- 28.** 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.

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

НАЧАЛО

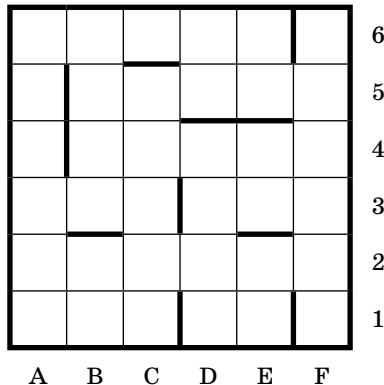
ПОКА < сверху свободно > вверх

ПОКА < слева свободно > влево

ПОКА < снизу свободно > вниз

ПОКА < справа свободно > вправо

КОНЕЦ



1) 1

2) 2

3) 3

4) 0

29. 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.

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

НАЧАЛО

ПОКА < снизу свободно > вниз

ПОКА < справа свободно > вправо

ПОКА < сверху свободно > вверх

ПОКА < слева свободно > влево

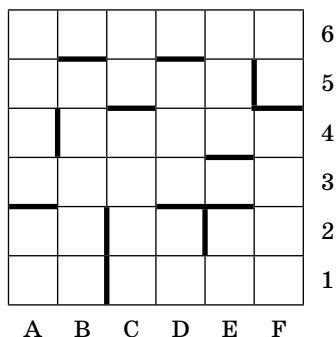
КОНЕЦ

1) 1

2) 2

3) 3

4) 0



30. 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.

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

НАЧАЛО

ПОКА < слева свободно > влево

ПОКА < справа свободно > вправо

ПОКА < снизу свободно > вниз

ПОКА < сверху свободно > вверх

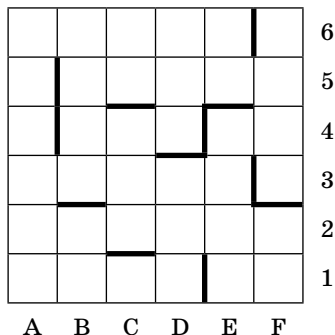
КОНЕЦ

1) 5

2) 9

3) 3

4) 7



- 31.** Исполнитель Черепашка перемещается на экране компьютера, оставляя след в виде линии. В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существуют две команды:

Вперед n , вызывающая передвижение Черепашки на n шагов в направлении движения.

Направо t , вызывающая изменение направления движения на t градусов по часовой стрелке.

Вместо n и t должны стоять целые числа.

Запись:

Повтори 5 [Команда 1 Команда 2]

означает, что последовательность команд в квадратных скобках повторится 5 раз.

Какое число необходимо записать вместо n в следующем алгоритме:

Повтори 6 [Вперед 20 Направо n],

чтобы на экране появился правильный прямоугольник?

- 1) 15
- 2) 45
- 3) 60
- 4) 90

- 32.** Исполнитель Черепашка перемещается на экране компьютера, оставляя след в виде линии. В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существуют две команды:

Вперед n , вызывающая передвижение Черепашки на n шагов в направлении движения.

Направо t , вызывающая изменение направления движения на t градусов по часовой стрелке.

Вместо n и t должны стоять целые числа.

Запись:

Повтори 5 [Команда 1 Команда 2]

означает, что последовательность команд в квадратных скобках повторится 5 раз.

Какая фигура появится на экране после выполнения следующего алгоритма:

Повтори 8 [Вперед 30 Направо 60]?

- 1) квадрат
- 2) треугольник
- 3) ломаная линия
- 4) шестиугольник

33. В приведенном ниже фрагменте алгоритма, записанном на алгоритмическом языке, переменные **a**, **b**, **c** имеют тип «строка», а переменные **i**, **k** — тип «целое». Используются следующие функции:

Длина (*a*) — возвращает количество символов в строке *a*. (Тип «целое».)

Извлечь (*a*, *i*) — возвращает *i*-тый (слева) символ в строке *a*. (Тип «строка».)

Склеить (*a*, *b*) — возвращает строку, в которой записаны сначала все символы строки *a*, а затем все символы строки *b*. (Тип «строка»)

Значения строк записываются в одинарных кавычках. (Например, $a := \text{'дом'}$.)

Фрагмент алгоритма:

```

i := Длина (a) - 5
k := 1
b := 'A'
пока i > 2
  нц
    c := Извлечь (a, i)
    b := Склеить (c, b)
    i := i - k
  кц

```

Какое значение будет у переменной **b** после выполнения вышеприведенного фрагмента алгоритма, если значение переменной **a** было «ИНФОРМАТИКА»?

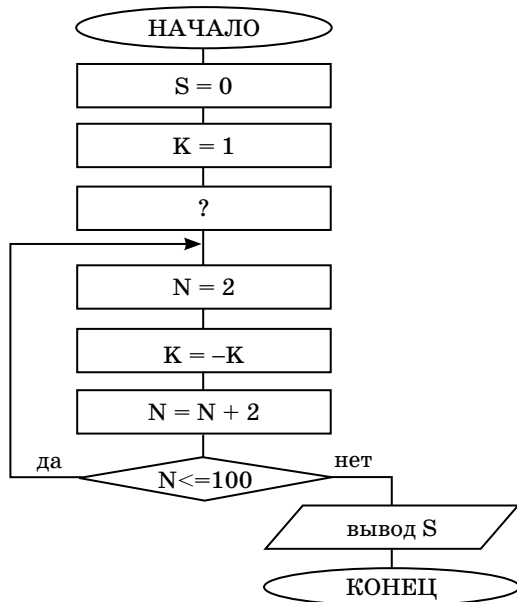
- 1) «ФОРМА»
- 2) «ФИРМА»
- 3) «МИНОР»
- 4) «АРФА»

34. Исполнитель Робот действует на клетчатой доске, между соседними клетками которой могут стоять стены. Робот передвигается по клеткам доски и может выполнять команды 1 (вверх), 2 (вниз), 3 (вправо), 4 (влево), переходя на соседнюю клетку в направлении, указанном в скобках. Если в этом направлении между клетками стоит стена, то Робот разрушается. Робот успешно выполнил программу вида: **1313133244**.

Какую последовательность из четырех команд должен выполнить Робот, чтобы вернуться в ту клетку, где он был перед началом выполнения программы, и не разрушиться вне зависимости от того, какие стены стоят на поле?

- 1) 4423 2) 3113 3) 4242 4) 1313

35. Какая команда должна быть в блок-схеме подсчета суммы четных чисел $S = 2 + 4 + 6 \dots + 100$ на месте вопросительного знака?



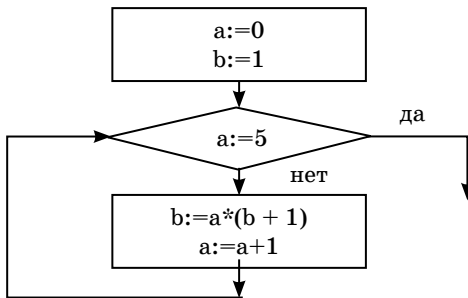
- 1) $S = S + N$
2) $S = S + K$

- 3) $S = N + K$
4) $S = S + N * K$

Часть 2

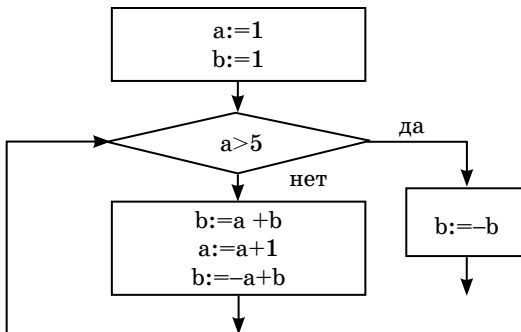
Ответом к заданиям этой части является набор символов, которые следует записать справа от номера соответствующего задания, начиная с первой клеточки. Каждый символ пишете в отдельной клеточке в соответствии с приведенными образцами (если имеются бланки ЕГЭ).

- 36.** Запишите значение переменных **b** после выполнения фрагмента алгоритма:



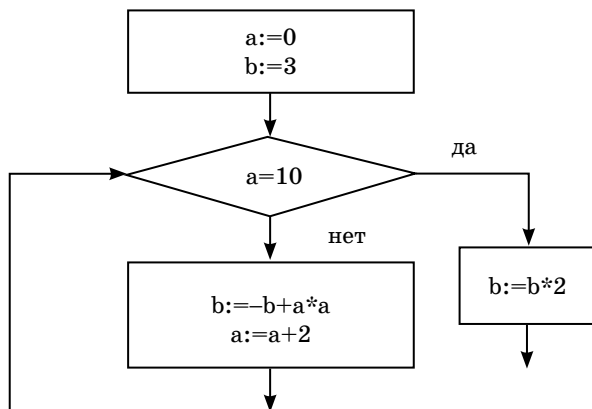
Примечание. Знаком $:=$ обозначена операция присваивания, знаком $*$ обозначена операция умножения.

- 37.** Запишите значение переменных **b** после выполнения фрагмента алгоритма:



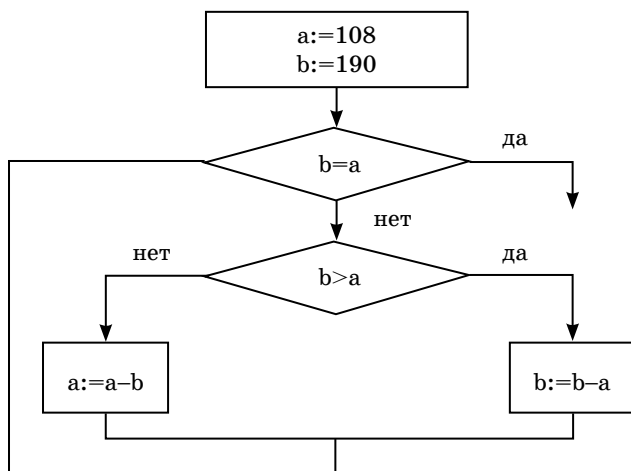
Примечание. Знаком $:=$ обозначена операция присваивания, знаком $*$ обозначена операция умножения.

38. Запишите значение переменной **b** после выполнения фрагмента алгоритма:



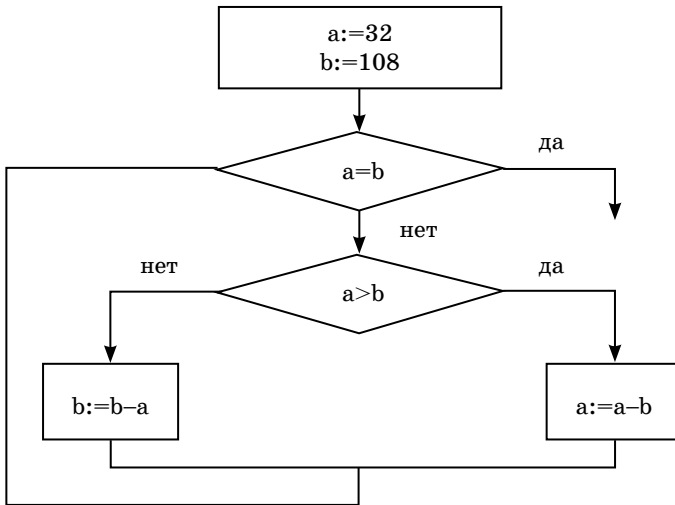
Примечание. Знаком $:=$ обозначена операция присваивания, знаком $*$ обозначена операция умножения.

39. Запишите значение переменной **b** после выполнения фрагмента алгоритма:



Примечание. Знаком $:=$ обозначена операция присваивания, знаком $*$ обозначена операция умножения.

40. Запишите значение переменных **b** после выполнения фрагмента алгоритма:



Примечание. Знаком $:=$ обозначена операция присваивания, знаком $*$ обозначена операция умножения.

41. У исполнителя Калькулятор две команды, которым присвоены номера:

1. прибавь 1

2. умножь на 3

Выполняя первую из них, Калькулятор прибавляет к числу на экране 1, а выполняя вторую, умножает его на 3. Запишите порядок команд в программе получения из числа 1 числа 45, содержащей не более 5 команд, указывая лишь номера команд. (Например, программа 21211 — это программа, содержащая следующие команды:

умножь на 3

прибавь 1

умножь на 3

прибавь 1

прибавь 1,

которые преобразуют число 3 в 32.)

- 42.** У исполнителя Калькулятор две команды, которым присвоены номера:

1. вычти 3

2. умножь на 4

Выполняя первую из них, Калькулятор уменьшает число на экране на 3, а выполняя вторую, увеличивает его в 4 раза. Запишите порядок команд в программе получения из числа 2 числа 500, содержащей не более 5 команд, указывая лишь номера команд. (Например, программа 21211 — это программа, содержащая следующие команды:

умножь на 4

вычти 3

умножь на 4

вычти 3

вычти 3,

которые преобразуют число 5 в 62.)

- 43.** У исполнителя Калькулятор две команды, которым присвоены номера:

1. умножь на 2

2. прибавь 3

Выполняя первую из них, Калькулятор умножает число на экране на 2, а выполняя вторую, прибавляет к нему 3. Запишите порядок команд в программе получения из числа 0 числа 30, содержащей не более 5 команд, указывая лишь номера команд. (Например, программа 21211 — это программа, содержащая следующие команды:

прибавь 3

умножь на 2

прибавь 3

умножь на 2

умножь на 2,

которые преобразуют число 1 в 44.)

- 44.** У исполнителя Квадратор две команды, которым присвоены номера:

1. возведи в квадрат

2. вычти 1

Первая из них возводит в квадрат число на экране, а вторая — уменьшает его на 1. Запишите порядок команд в программе получения из числа 5 числа 225, содержащей не более 4 команд, указывая лишь номера команд. (Например, программа 21211 — это программа, содержащая следующие команды:

вычти 1
 возведи в квадрат
 вычти 1
 возведи в квадрат
 возведи в квадрат,
 которые преобразуют число 3 в 81.)

45. У исполнителя Утроитель две команды, которым присвоены номера:

1. прибавь 2

2. умножь на 3

Первая из них увеличивает число на экране на 2, а вторая — утраивает его. Запишите порядок команд в программе получения из числа 2 числа 40, содержащей не более 5 команд, указывая лишь номера команд. (Например, программа 21211 — это программа, содержащая следующие команды:

умножь на три
 прибавь 2
 умножь на три
 прибавь 2
 прибавь 2,
 которые преобразуют число 3 в 37.)

46. Строки (цепочки символов латинских букв) создаются по следующему правилу. Первая строка состоит из одного символа — латинской буквы «А». Каждая из последующих цепочек создается такими действиями: в очередную строку сначала записывается буква, чей порядковый номер в алфавите соответствует номеру строки (на “i”-м месте пишется “i”-я буква алфавита), к ней справа дважды подряд приписывается предыдущая строка. Вот первые 4 строки, созданные по этому правилу:

А
ВАА
СВААВАА
ДСВААВААСВААВАА

Латинский алфавит (для справки):
ABCDEFGHIJKLMNOPQRSTUVWXYZ.

Запишите шесть символов подряд, стоящих в пятнадцатой строке с одиннадцатого по шестнадцатое место (считая слева направо).

47. Строки (цепочки символов латинских букв) создаются по следующему правилу. Первая строка состоит из одного символа — латинской буквы «А». Каждая из последующих цепочек создается такими действиями: в очередную строку сначала записывается буква, чей порядковый номер в алфавите соответствует номеру строки (на “i”-м месте пишется “i”-я буква алфавита, к ней справа дважды подряд приписывается предыдущая строка). Вот первые 4 строки, созданные по этому правилу:

А
ВАА
СВААВАА
ДСВААВААСВААВАА

Латинский алфавит (для справки):
ABCDEFGHIJKLMNOPQRSTUVWXYZ.

Сколько раз встречается буква «А» в первых семи строках (суммарно)?

48. Цепочки символов (строки) создаются по следующему правилу. Первая строка состоит из одного символа — цифры «1». Каждая из последующих цепочек создается такими действиями: в начало записывается число — номер строки по порядку (для “i”-й строки ставится число “i”, далее дважды подряд записывается предыдущая строка). Вот первые 4 строки, созданные по этому правилу:

1
211
3211211
432112113211211

Сколько раз в общей сложности встречаются в седьмой строке нечетные цифры (1, 3, 5, 7)?

49. Цепочки символов (строки) создаются по следующему правилу. Первая строка состоит из одного символа — цифры «1». Каждая из последующих цепочек создается следующими действиями: в очередную строку дважды записывается предыдущая цепочка цифр (одна за другой, подряд), а в конце приписывается еще одно число — номер строки по порядку (на “i”-м месте дописывается число “i”). Вот первые 4 строки, созданные по этому правилу:

1
112
1121123
112112311211234

Какая цифра стоит в восьмой строке на 120-м месте (считая слева направо)?

50. Цепочки символов (строки) создаются по следующему правилу. Первая строка состоит из одного символа — цифры «1». Каждая из последующих цепочек создается следующими действиями: в очередную строку дважды записывается предыдущая цепочка цифр (одна за другой, подряд), а в конце приписывается еще одно число — номер строки по порядку (на “i”-м месте дописывается число “i”). Вот первые 4 строки, созданные по этому правилу:

1
112
1121123
112112311211234

Запишите пять символов подряд, стоящих в седьмой строке со 119-го по 123-е место (считая слева направо).

Часть 3

На бланке запишите сначала номер задания, а затем — полное решение. Ответы записывайте четко и разборчиво.

- 51.** Два игрока играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. В начале игры фишка находится в точке с координатами $(4,3)$. Ход состоит в том, что игрок перемещает фишку из точки с координатами (x,y) в одну из трех точек: или в точку с координатами $(x+2,y)$, или в точку с координатами $(x+3,y)$, или в точку с координатами $(x,y+4)$. Выигрывает игрок, после хода которого расстояние от фишки до точки с координатами $(0,0)$ не меньше 12 единиц. Кто выигрывает при безошибочной игре обоих партнеров — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.
- 52.** Два игрока играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. В начале игры фишка находится в точке с координатами $(7,15)$. Ход состоит в том, что игрок перемещает фишку из точки с координатами (x,y) в одну из трех точек: или в точку с координатами $(x-2,y)$, или в точку с координатами $(x,y-2)$, или в точку с координатами $(x,y-3)$. Выигрывает игрок, после хода которого расстояние от фишки до точки с координатами $(0,0)$ не более 11 единиц. Кто выигрывает при безошибочной игре обоих партнеров — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.
- 53.** Два игрока играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. В начале игры фишка находится в точке с координатами $(5,3)$. 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 $(x, y + 2)$, или в точку с координатами $(x, y + 3)$. Выигрывает игрок, после хода которого расстояние от фишки до точки с координатами $(0, 0)$ не менее 13 единиц. Кто выигрывает при безошибочной игре обоих партнеров — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.

54. Два игрока играют в следующую игру. Перед ними лежат две кучки камней, в первой из которых 3, а во второй 1 камень. У каждого игрока неограниченно много камней. Игроки ходят по очереди. Ход состоит в том, что игрок или увеличивает в 3 раза число камней в какой-то кучке, или добавляет два камня в какую-то кучку. Выигрывает игрок, после хода которого общее число камней в двух кучках становится не менее 30. Кто выигрывает при безошибочной игре — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.
55. Два игрока играют в следующую игру. Перед ними лежат две кучки камней, в первой из которых 3, а во второй 5 камней. У каждого игрока неограниченно много камней. Игроки ходят по очереди. Ход состоит в том, что игрок или увеличивает в 2 раза число камней в какой-то кучке, или добавляет три камня в какую-то кучку. Выигрывает игрок, после хода которого общее число камней в двух кучках становится не менее 18. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен ходить выигрывающий игрок? Ответ обоснуйте.

ОСНОВЫ ЛОГИКИ

В экзаменационных заданиях используются следующие соглашения:

символ 1 используется для обозначения истины (истинного высказывания);

символ 0 — для обозначения лжи (ложного высказывания).

Обозначения для логических связей (операций):

а) *отрицание* (инверсия, логическое НЕ) обозначается \neg (например, $\neg A$);

A	$\neg A$
0	1
1	0

б) *конъюнкция* (логическое умножение, логическое И) обозначается \wedge (например, $A \wedge B$) либо $\&$ (например, $A \& B$);

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

в) *дизъюнкция* (логическое сложение, логическое ИЛИ) обозначается \vee (например, $A \vee B$) либо $|$ (например, $A | B$);

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

d) *следование* (импликация) обозначается \rightarrow (например, $A \rightarrow B$);

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

Приоритеты логических операций

1. Операции в скобках.
2. Инверсия (отрицание).
3. Конъюнкция (логическое умножение).
4. Дизъюнкция (логическое сложение).
5. Импликация (следование).

Тождества

двойного отрицания

$$\neg(\neg A) = A$$

логического сложения

$$1) A \vee 0 = A$$

$$2) A \vee 1 = 1$$

$$3) A \vee A = A$$

$$4) A \vee \neg A = 1$$

логического умножения

- 1) $A \wedge 0 = 0$
- 2) $A \wedge 1 = A$
- 3) $A \wedge A = A$
- 4) $A \wedge \neg A = 0$

Законы алгебры логики

В большинстве учебников по информатике приняты следующие определения:

Коммуникативный (переместительный) закон

- 1) $A \vee B = B \vee A$
- 2) $A \wedge B = B \wedge A$

Ассоциативный (сочетательный) закон

- 3) $(A \vee B) \vee C = A \vee (B \vee C)$
- 4) $(A \wedge B) \wedge C = A \wedge (B \wedge C)$

Дистрибутивный (распределительный) закон

- 5) $(A \vee B) \wedge C = (A \wedge C) \vee (B \wedge C)$
- 6) $A \wedge B \vee C = (A \vee C) \wedge (B \vee C)$

Законы де Моргана

- 7) $\neg(A \vee B) = \neg A \wedge \neg B$
- 8) $\neg(A \wedge B) = \neg A \vee \neg B$
- 9) $A \rightarrow B = (\neg B) \rightarrow (\neg A) = \neg A \vee B$
- 10) $(A \wedge B) \vee \neg(A \wedge B) = (\neg A \vee B) \wedge (A \vee \neg B)$ или $(A \wedge B) \vee \neg A \wedge \neg B = (\neg A \vee B) \wedge (A \vee \neg B)$

Например, необходимо упростить выражение $\neg B \wedge (A \vee \neg(A \wedge B))$. Сначала применим закон де Моргана, и выражение примет вид:

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

Теперь используем сочетательный закон:

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

Затем применим тождество логического сложения:

$$\neg B \wedge (1 \vee \neg B).$$

Снова воспользуемся тождеством логического сложения:

$$\neg B \wedge 1.$$

А теперь тождеством логического умножения:

$$\neg B.$$

Из приведенного примера следует, что выражение $\neg B \wedge (A \vee \neg(A \wedge B))$ упростилось до $\neg B$.

Рассмотрим еще один пример:

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

$(A \vee C) \wedge (\neg B \vee C)$ — распределительный закон

$(A \vee 0) \wedge (\neg B \vee C)$ — тождество логического умножения

$A \wedge (\neg B \vee C)$ — тождество логического сложения

Ответ: $A \wedge (\neg B \vee C)$ или по распределительному закону $(A \wedge \neg B) \vee (A \wedge C)$.

ЗАДАНИЯ

Часть 1

При выполнении заданий этой части необходимо выбрать один верный ответ из представленных четырех вариантов и поставить знак «х» в клеточку, номер которой соответствует номеру выбранного вами ответа.

1. Для какого из указанных значений X истинно высказывание $\neg ((X > 3) \rightarrow (X > 4))$?
 - 1) 1
 - 2) 2
 - 3) 3
 - 4) 4
2. Для какого из указанных значений X истинно высказывание $\neg ((X > 56) \rightarrow (X > 198)) \wedge (X < 369)$?
 - 1) 100
 - 2) 200
 - 3) 300
 - 4) 400

3. Для какого из указанных значений X истинно высказывание $((X < 5) \vee (X > 5)) \rightarrow (X > 15)$?
- 1) 1
 - 2) 5
 - 3) 10
 - 4) 15
4. Для какого из указанных значений X ложно высказывание $((X > 3) \rightarrow (X > 6)) \vee (\neg (X < 4) \rightarrow (X < 5))$?
- 1) 5
 - 2) 2
 - 3) 3
 - 4) 4
5. Для какого из указанных значений X истинно высказывание $((X > 4) \rightarrow (X > 5)) \wedge (X < 4) \rightarrow (X < 5)$?
- 1) 6
 - 2) 5
 - 3) 3
 - 4) 4
6. Укажите, какое логическое выражение равносильно выражению $\neg(A \vee B) \vee C$.
- 1) $\neg A \vee \neg B \vee \neg C$
 - 2) $(\neg A \wedge \neg B) \vee C$
 - 3) $(A \wedge B) \vee \neg C$
 - 4) $A \wedge \neg B \wedge C$
7. Укажите, какое логическое выражение равносильно выражению $\neg(A \vee \neg(B \vee C))$.
- 1) $\neg A \vee \neg B \vee \neg C$
 - 2) $A \wedge \neg(B \wedge \neg C)$
 - 3) $\neg A \wedge (B \vee C)$
 - 4) $A \vee (\neg B \wedge \neg C)$
8. Укажите, какое логическое выражение равносильно выражению $\neg(A \wedge B) \wedge C$.
- | | |
|------------------------------------|---------------------------------|
| 1) $(\neg A \vee \neg B) \wedge C$ | 3) $(A \wedge B) \wedge \neg C$ |
| 2) $A \wedge \neg B \wedge \neg C$ | 4) $A \wedge \neg(B \wedge C)$ |

9. Укажите, какое логическое выражение равносильно выражению $\neg(A \wedge B) \vee C$.

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

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

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

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

10. Укажите, какое логическое выражение равносильно выражению $\neg(A \vee \neg B \vee \neg C)$.

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

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

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

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

11. 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
1	1	0	1
1	0	1	0
1	1	1	1

Какое выражение соответствует F?

1) $\neg X \vee Y \vee \neg Z$

2) $X \wedge Y \wedge \neg Z$

3) $X \vee \neg Y \vee Z$

4) $\neg X \vee \neg Y \vee \neg Z$

12. 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
1	0	0	1
0	1	0	1
0	1	1	0

Какое выражение соответствует F?

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

13. 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	1	0
1	1	1	0

Какое выражение соответствует F?

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

14. 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	0
0	1	1	0
1	0	1	0

Какое выражение соответствует F?

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

15. 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	0	0	1
0	1	1	0

Какое выражение не соответствует F?

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

Часть 2

Ответом к заданиям этой части является набор символов, которые следует записать справа от номера соответствующего задания начиная с первой клеточки. Каждый символ пишете в отдельной клеточке в соответствии с приведенными образцами (если имеются бланки ЕГЭ).

16. Каково наибольшее целое число X, при котором истинно высказывание

$$(10 < X \cdot X \cdot X) \rightarrow (20 > (X+1) \cdot (X+1) \cdot (X+1))?$$

17. Каково наибольшее целое число X, при котором истинно высказывание

$$(10 < X \cdot (X+1)) \rightarrow (10 > (X+1) \cdot (X+2))?$$

18. Каково наибольшее целое число X, при котором истинно высказывание

$$(25 < X \cdot (X+1) \cdot (X+2)) \rightarrow (X < (X-1))?$$

19. Каково наибольшее целое число X , при котором истинно высказывание

$$(10 < X \cdot X) \rightarrow (10 > (X+1) \cdot (X+2))?$$

20. Каково наименьшее целое число X , при котором ложно высказывание

$$(10 < X \cdot X \cdot X) \rightarrow (10 > ((X+1) \cdot (X+1) - X))?$$

21. A, B, C — целые числа, для которых истинно высказывание:

$$((A < C) \rightarrow ((2B+1) > C)) \wedge \neg((C < A) \rightarrow (C < (2B+1)))$$

Чему равно C , если $A = 8, B = 3$?

22. Укажите значения логических переменных K, L, M, N , при которых логическое выражение

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

ложно.

Ответ запишите в виде строки из 4 символов: значений переменных K, L, M и N (в указанном порядке). Так, например, строка 1001 соответствует тому, что $K = 1, L = 0, M = 0$ и $N = 1$.

23. Укажите значения логических переменных K, L, M и N , при которых логическое выражение

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

истинно.

Ответ запишите в виде строки из 4 символов: значений переменных K, L, M и N (в указанном порядке). Так, например, строка 1001 соответствует тому, что $K = 1, L = 0, M = 0$ и $N = 1$.

24. Сколько различных решений имеет уравнение

$$(\neg K \vee N) \rightarrow (L \wedge M \vee K) = 0,$$

где K, L, M, N — логические переменные?

В ответе не нужно перечислять все различные наборы значений K, L, M, N , при которых выполнено данное равенство. В качестве ответа нужно указать количество таких наборов.

25. Сколько различных решений имеет уравнение

$$((K \wedge L) \rightarrow M) \wedge (N \wedge K \vee L) = 1,$$

где K, L, M, N — логические переменные?

В ответе не нужно перечислить все различные наборы значений K, L, M, N , при которых выполнено данное равенство. В качестве ответа нужно указать количество таких наборов.

26. Команды четырех соседних школ соревновались в волейбольном турнире. Болельщики поделились своими предположениями:

1-й болельщик: 556-я победит, а 450-я — будет второй.

2-й болельщик: Нет, 556-я будет последней, а вот победит 324-я.

3-й болельщик: Уверен, что первой будет 445-я, а 450-я — третьей.

Когда соревнования закончились, выяснилось, что каждый болельщик был не прав в одном из своих прогнозов. Какое место заняли команды школы № 324, школы № 445, школы № 450 и школы № 556? (В ответе перечислите подряд без пробелов места школьных команд в указанном порядке номеров школ.)

27. В конкурсе «А ну-ка, парни!» в финал вышли четыре мальчика: Никита, Руслан, Сергей и Толя. Девочки решили поделиться своими предположениями об итоговом распределении мест:

Оля: Сережа точно будет вторым, а Толик — четвертым.

Аня: Уверена, что Никита будет первым, а вторым — Руслан.

Кристина: Ерунда. Это Никита будет вторым, а Толик — третьим.

Когда подвели итоги, оказалось, что каждая девочка была права только в одном из своих прогнозов. Какое место заняли Никита, Руслан, Сергей и Толя? (В ответе перечислите подряд без пробелов места мальчиков в указанном порядке имен.)

- 28.** 1 апреля три подружки Аня, Света и Оля решили разыграть одноклассников и договорились, что одна из них будет говорить сегодня только ложь, другая — только правду, а третья — через раз то ложь, то правду.

Они втроем прогуляли классный час и пришли только на праздник, хотя раньше так никогда не поступали. Классный руководитель выслушала девочек:

Аня: Я всегда прогуливаю классный час, а Света скажет вам неправду.

Света: Я сегодня первый раз прогуляла классный час.

Оля: Аня сегодня говорит только чистую правду.

Классный руководитель догадалась, кто из девочек сегодня в какой роли выступает. Расположите первые буквы имен девочек в таком порядке: «Всегда говорит правду», «Всегда лжет», «Говорит правду через раз». (Пример: если бы имена девочек были: Зина, Маша и Вера, ответ тогда был бы: МВЗ.)

- 29.** Проверяя дневники, классный руководитель заметил, что мальчику Роме исправлены все двойки за неделю, а сделать это могли только три его друга: Максим (М), Андрей (А) и Костя (К), которые задержались на перемену в классе. Они были вызваны к директору, где их спросили о том, кто подделал оценку:

Андрей: Максим этого не делал, это все Костя красной ручкой!

Костя: Я этого не делал, потому что оценку исправил Максим!

Максим: Ничего я не исправлял! Да и Андрей тоже...

Стало известно, что один из мальчиков сказал чистую правду, второй все соврал, а третий сказал правду только в половине своего ответа. Кто же подделал оценку Роме? Запишите только первую букву имени.

- 30.** На перемене в кабинете биологии 8 ребят баловались и разбили дорогой микроскоп. Их всех вызвали к директору и выслушали:

Ира: Это Антон разбил.

Наташа: Нет, Антон не бил!

Сергей: А я тоже знаю, что это Наташа разбила!

Антон: Нет, ни Наташа, ни Сергей этого не делали!

Оля: А я видела, что разбил Сергей!

Максим: Это кто-то чужой!

Костя: Это либо Наташа, либо Сергей — больше никому!

Кто разбил микроскоп, если известно, что из этих восьми высказываний истинны только два?

(Ответ запишите в виде первой буквы имени.)

МОДЕЛИРОВАНИЕ И КОМПЬЮТЕРНЫЙ ЭКСПЕРИМЕНТ

Модель — это упрощенное представление о реальном объекте, процессе или явлении, которое является средством его обследования.

Модель — любой аналог, образ (изображение, описание, схема, символ, формула, чертеж, график, план, карта, таблица и т.п.) какого-либо объекта исследования.

Модель сохраняет наиболее важные и существенные для цели исследования характеристики и свойства оригинала.

Один и тот же объект может иметь множество моделей.

Модель необходима для того, чтобы:

- понять, как устроен реальный объект;
- научиться управлять объектом или процессом;
- прогнозировать прямые или косвенные последствия реализации заданных способов и форм воздействия на объект.

Моделирование — построение и изучение моделей с целью получения новых знаний, научного познания объективного мира или дальнейшего совершенствования характеристик объекта исследования.

В процессе классификации объектов часто строятся модели, которые имеют **иерархическую структуру**. Группы объектов, обладающих одинаковыми общими свойствами, называются *классами объектов*. Внутри класса объектов могут быть выделены подклассы, объекты которых обладают некоторыми особенными свойствами, в свою очередь подклассы могут делиться на еще более мелкие группы и так далее. Такой процесс систематизации объектов называется процессом классификации.

Классификация моделей

По области использования:

- игровые;
- имитационные;
- научно-технические;
- опытные;
- учебные.

С учетом фактора времени:

- статические;
- динамические;
- дискретные;
- непрерывные.

По области знаний:

- биологические;
- исторические;
- математические;
- социологические и т.п.

По способу представления:

- материальные;
- информационные;
- вербальные;
- знаковые.

Информационная модель — описание реального объекта (процесса, явления) в образной или знаковой форме, а также совокупность информации, характеризующая его существенные свойства и состояния и взаимосвязь с внешним миром. Информационные модели делятся на **описательные** и **формальные**.

Описательные информационные модели — это модели, созданные на естественном языке (т.е. на языке общения между людьми) в устной или письменной форме.

Формальные информационные модели — это модели, созданные на формальном языке (т.е. научном, профессиональном или программирования). Примеры формальных моделей: все виды формул, таблицы, графы, карты, схемы и т.д.

По форме представления:

- геометрические;
- компьютерные;

- логические;
- математические;
- словесные;
- специальные;
- структурные и т.п.

Большую часть моделей, которыми пользуется человек, принято называть системами.

Система — это целое, состоящее из элементов, взаимосвязанных между собой. Однако связи элементов создают систему только тогда, когда образуют новый целостный объект, обладающий свойствами, которые не были присущи просто совокупности данных элементов.

Систематизация (классификация) — процесс превращения множества объектов в систему.

Структура системы — определенный порядок объединения элементов системы.

Структурные информационные модели:

- алгоритмы;
- иерархические;
- сетевые;
- схемы (графы);
- табличные.

Рассмотрим подробнее самые значительные из них.

Табличной информационной моделью является прямоугольная таблица, которая состоит из столбцов и строк. Такой тип моделей применяется для описания ряда объектов, обладающих одинаковыми наборами свойств. Это один из наиболее используемых типов информационных моделей.

В иерархической информационной модели объекты распределены по уровням. Каждый элемент более высокого уровня может состоять из элементов нижнего уровня, а элемент нижнего уровня может входить в состав только одного элемента более высокого уровня.

Сетевые информационные модели применяются для отражения систем со сложной структурой, в которых связи между элементами имеют произвольный характер. Например, это различные региональные части глобальной компьютерной сети Интернет.

Компьютерные модели — это модели, реализованные на компьютере средствами программного обеспечения.

Компьютерные модели:

- текст;
- графика;
- таблицы;
- звук;
- алгоритмы.

Основные этапы моделирования на компьютере:

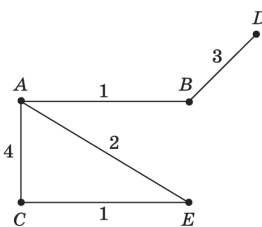
- Построение модели.
- Формализация модели.
- Построение компьютерной модели.
- Проведение компьютерного эксперимента.
- Анализ результатов моделирования.

В заданиях ЕГЭ встречаются задачи на сравнение табличной формы моделей и моделей в виде схем, которые описывают один объект.

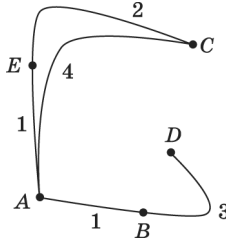
Например, в таблице приведена стоимость перевозок между соседними железнодорожными станциями. Укажите схему, соответствующую таблице.

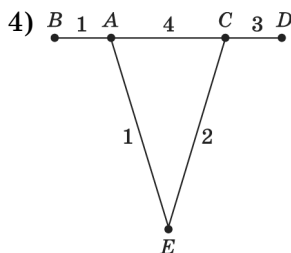
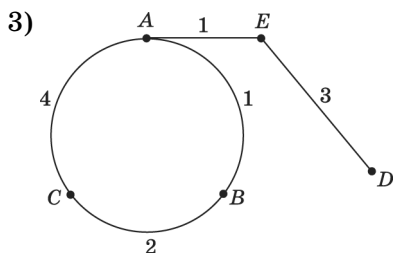
	A	B	C	D	E
A		1	4		1
B	1			3	
C	4				2
D		3			
E	1		2		

1)



2)





Решение:

Представим значения отрезков линий в виде четырех таблиц:

1			2			3			4		
AE	2	–	AE	1	+	AE	1	+	AE	1	+
AB	1	+	AB	1	+	AB	1	+	AB	1	+
AC	4	+	AC	4	+	AC	4	+	AC	4	+
BD	3	+	BD	3	+	DE	3	–	CE	2	+
CE	1	–	CE	2	+	BC	2	–	CD	3	–

Анализируя полученную таблицу, можно сделать вывод, что только чертеж № 2 удовлетворяет представленной таблице.

Ниже приведен пример задачи, в которой необходимо определить вид табличной модели, которая удовлетворяет определенным условиям.

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 А в В 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		

3)

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

2)

	A	B	C	D	E
A			3	1	1
B			4		
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		

В этой задаче необходимо сделать модель перевозок в виде схемы для каждой из четырех таблиц, а затем подсчитать стоимость проезда.

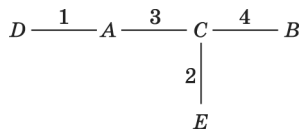
Рассмотрим алгоритм построения схемы на примере **третьей таблицы**.

Имеется станция А, которая связана со станциями С и D.

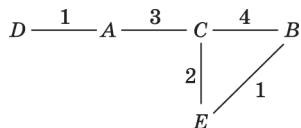
D — A — C

Станция D не связана ни с одной станцией, кроме А.

Станция С, кроме станции А, связана еще со станциями В и Е.



Из дальнейшего анализа таблицы видно, что станции В и Е связаны между собой:



Теперь подсчитаем стоимость обоих возможных путей:

$$АСВ = 3 + 4 = 7$$

$$АСЕВ = 3 + 2 + 1 = 6$$

Второй вариант подходит под условие задачи.

Аналогично можно нарисовать схему по любой таблице и наоборот.

Математические модели реализуются с помощью компьютерных экспериментов на различных языках программирования (например, в заданиях 1—5 в теме «Технологии программирования») и в электронных таблицах EXCEL.

ЗАДАНИЯ

Часть 1

При выполнении заданий этой части необходимо выбрать один верный ответ из представленных четырех вариантов и поставить знак «х» в клеточку, номер которой соответствует номеру выбранного вами ответа.

1. Между четырьмя крупными аэропортами, обозначенными кодами СЕВЕРНЫЙ, ЮЖНЫЙ, ЗАПАДНЫЙ и ВОСТОЧНЫЙ, ежедневно выполняются авиарейсы. Приведен фрагмент расписания перелетов между этими аэропортами.

Аэропорт вылета	Аэропорт прилета	Время вылета	Время прилета
СЕВЕРНЫЙ	ЮЖНЫЙ	7:20	9:40
ЗАПАДНЫЙ	ВОСТОЧНЫЙ	11:25	12:30
ЮЖНЫЙ	СЕВЕРНЫЙ	11:40	13:30
ВОСТОЧНЫЙ	ЗАПАДНЫЙ	13:15	15:20
СЕВЕРНЫЙ	ЗАПАДНЫЙ	13:50	17:30

Аэропорт вылета	Аэропорт прилета	Время вылета	Время прилета
ЗАПАДНЫЙ	СЕВЕРНЫЙ	14:20	16:40
ЮЖНЫЙ	ЗАПАДНЫЙ	14:35	18:30
ЗАПАДНЫЙ	ЮЖНЫЙ	16:30	18:15
ВОСТОЧНЫЙ	СЕВЕРНЫЙ	18:40	20:25
СЕВЕРНЫЙ	ВОСТОЧНЫЙ	20:30	22:50

Путешественник находится в аэропорту ЮЖНЫЙ в полночь (0:00). Определите самое раннее время, когда он может оказаться в аэропорту ЗАПАДНЫЙ.

- 1) 18:30
- 2) 17:30
- 3) 15:20
- 4) 14:30

2. Между четырьмя крупными железнодорожными станциями, обозначенными кодами СЕРОВО, МУЛИНО, ПЕРВАК и ДОЛИНА, ежедневно выполняются железнодорожные переезды. Приведен фрагмент расписания переездов между этими станциями.

Станция выбытия	Станция прибытия	Время отправления	Время прибытия
ДОЛИНА	МУЛИНО	5:20	7:30
МУЛИНО	СЕРОВО	9:20	11:30
СЕРОВО	МУЛИНО	10:40	12:30
ПЕРВАК	ДОЛИНА	11:10	13:20
ДОЛИНА	СЕРОВО	11:40	15:30
МУЛИНО	ДОЛИНА	12:10	14:40
СЕРОВО	ДОЛИНА	12:40	16:20

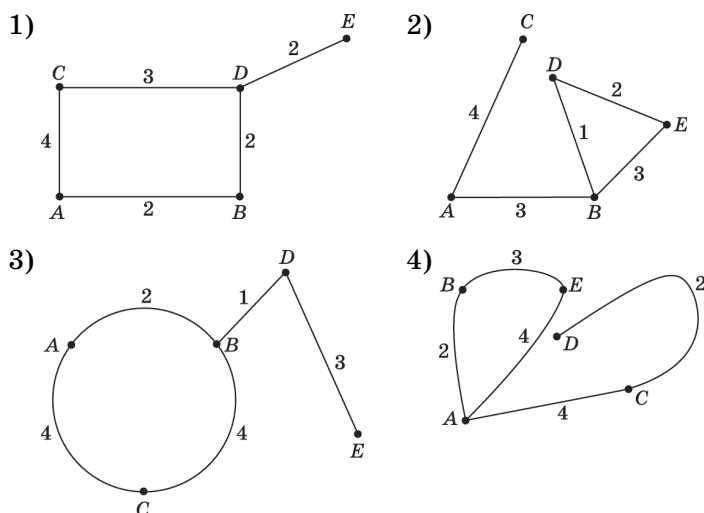
Станция выбытия	Станция прибытия	Время отправления	Время прибытия
СЕРОВО	ПЕРВАК	14:30	16:10
ДОЛИНА	ПЕРВАК	16:30	18:30
ПЕРВАК	СЕРОВО	18:40	20:50

Путешественник находится на станции СЕРОВО в полночь (0:00). Определите самое раннее время, когда он может оказаться на станции ДОЛИНА.

- 1) 13:20 2) 14:40 3) 16:20 4) 18:30

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

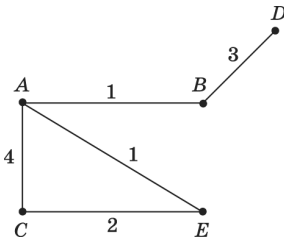
	A	B	C	D	E
A		2	4		
B	2			1	3
C	4				
D		1			2
E		3		2	



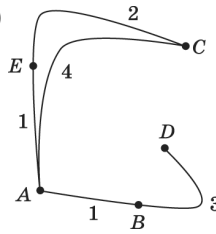
4. В таблице приведена стоимость перевозок между соседними железнодорожными станциями. Укажите схему, соответствующую таблице.

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

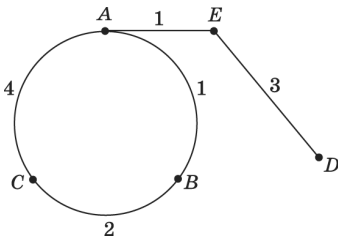
1)



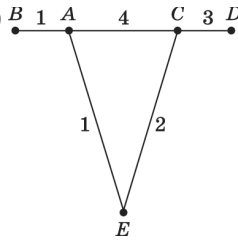
2)



3)



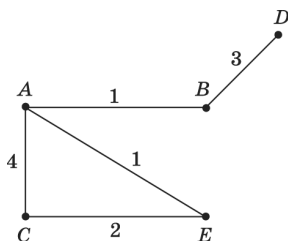
4)



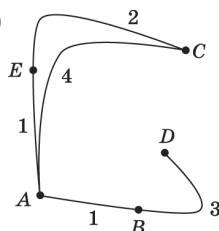
5. В таблице приведена стоимость перевозок между соседними железнодорожными станциями. Укажите схему, соответствующую таблице.

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

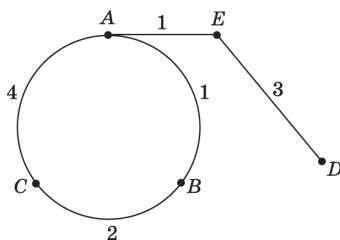
1)



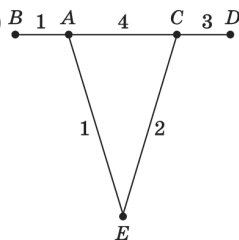
2)



3)



4)



6. 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 А в В не больше 6». Стоимость проезда по маршруту складывается из стоимостей проезда между соответствующими соседними станциями.

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

1)

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

2)

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

3)

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

4)

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

7. 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 А в В 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
B			4		1
C	3	4		2	
D			2		
E	1	1			

3)

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

4)

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

8. 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 А в В не больше 6».

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

1)

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

2)

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

3)

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

4)

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

9. Грунтовая дорога проходит последовательно через прибрежные населенные пункты А, В, С и D. При этом длина дороги между А и В равна 15 км, между В и С — 45 км и между С и D — 20 км. Расстояние по воде между А и D 60 км, и работает паромное сообщение. Оцените минимально возможное время движения велосипедиста из пункта А в пункт С, если его скорость по грунтовой дороге 20 км/час, а паром (которым можно воспользоваться) движется со скоростью 40 км/час.

- 1) 1,5 часа
- 2) 2 часа
- 3) 2,5 часа
- 4) 3 часа

10. Шоссе проходит последовательно через населенные пункты А, В, С и D. При этом длина дороги между А и В равна 15 км, между В и С — 10 км, между С и D — 5 км, а по реке между А и D — 20 км. Путешественник может идти пешком со скоростью 5 км/час или дополнительно воспользоваться лодкой, двигающейся со скоростью 10 км/час. Оцените минимально возможное время движения путешественника из пункта А в пункт С.

- 1) 1,5 часа
- 2) 2,5 часа
- 3) 3 часа
- 4) 3,5 часа

ПРОГРАММНЫЕ СРЕДСТВА ИКТ

Файл — именованная область памяти на техническом носителе. Каждый файл имеет имя, которое состоит из двух частей: собственного имени и расширения. Расширение отделяется от собственного имени точкой.

Все файлы хранятся в каталогах (папках). Обычно на диске имеется один главный каталог (корневой каталог), который не имеет имени. Имена остальных каталогов образуются аналогично собственным именам файлов. Но имена каталогов не имеют расширений. В любом каталоге могут находиться другие каталоги (подкаталоги).

Файловая система — способ организации служебной и пользовательской информации о файлах на носителях и взаимосвязи между ними. На носителе она образует иерархическую древообразную структуру.

Полный путь от диска до нужного файла записывается в одну строку. После имени диска ставится двоеточие, каталоги отделяются друг от друга левым слэшем (\). Полное имя файла состоит из пути доступа к файлу и его имени.

Например, полный путь до файла ЕГЭ.doc, который расположен на съемном диске G в каталоге ЭКЗАМЕНЫ, будет выглядеть так:

G	:	ЭКЗАМЕНЫ	\	ЕГЭ.doc
Имя диска		Имя каталога		Имя файла с расширением

G:\ ЭКЗАМЕНЫ\ ЕГЭ.doc

Путь доступа к файлу состоит из логического имени устройства и имен каталогов (разделенных символом \), всегда начинается с имени диска (гибкого A:\, жесткого C:\ или

съемного) и заканчивается именем каталога, в котором непосредственно хранится файл.

Например: A:\disk\music\my.mp3

Для групповых операций с файлами 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.

Например, маске ?a???* удовлетворяют имена файлов: dad22, para123, failege2009, aa123.

ЗАДАНИЯ

Часть 1

При выполнении заданий этой части необходимо выбрать один верный ответ из представленных четырех вариантов и поставить знак «х» в клеточку, номер которой соответствует номеру выбранного вами ответа.

1. Для групповых операций с файлами 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.

Определите, какое из указанных имен файлов удовлетворяет маске:

?ch?*.*t

- 1) 4school.ppt
- 2) school.ptp
- 3) school4.ptt
- 4) sch.ppt

2. Для групповых операций с файлами 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.

Определите, какое из указанных имен файлов удовлетворяет маске:

***a???.doc**

- 1) abcd.doc
- 2) mama.doc
- 3) partner.doc
- 4) dad_56.doc

3. Для групповых операций с файлами 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.

Определите, какое из указанных имен файлов удовлетворяет маске:

??e4*?.

- | | |
|------------|-----------|
| 1) ssole43 | 3) pre4 |
| 2) 554e455 | 4) a5e4o3 |

4. Для групповых операций с файлами 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.

Определите, какое из указанных имен файлов **не** удовлетворяет маске:

?ad?.d

1) cado.doc

3) 4ad4.docum

2) 4padron.doc

4) parady.dc

5. Для групповых операций с файлами 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.

Определите, какое из указанных имен файлов **не** удовлетворяет маске:

?*o??.*t

1) fon_ton1.ppt

2) cobr.pt

3) ooo.ppt

4) ozon2.mult

6. В некотором каталоге хранился файл ИКТ.doc. После того как в этом каталоге создали подкаталог и переместили в созданный подкаталог файл ИКТ.doc, полное имя файла стало

C:\USER\Документы\ЕГЭ\информатика\ИКТ.doc. Каково полное имя каталога, в котором хранился файл до перемещения?

- 1) C:\USER\Документы
- 2) C:\USER\ЕГЭ
- 3) C:\USER\Документы\ЕГЭ
- 4) C:\USER\ЕГЭ\Документы

7. В некотором каталоге хранился файл Demo.pdf. После того как в этом каталоге создали подкаталог и переместили в созданный подкаталог файл Demo.pdf, полное имя файла стало

A:\Demo ЕГЭ\Информатика\ЕГЭ\Demo\Demo.pdf. Каково полное имя каталога, в котором хранился файл до перемещения?

- 1) A:\Demo
- 2) A:\Demo ЕГЭ\Demo
- 3) A:\ Информатика
- 4) A:\Demo ЕГЭ\Информатика

8. В некотором каталоге хранился файл Литература.ppt. После того как в этом каталоге создали подкаталог и переместили в созданный подкаталог файл Литература.ppt, полное имя файла стало

F:\ Литература\Подготовка\ЕГЭ\Demo\Литература.ppt. Каково полное имя каталога, в котором хранился файл до перемещения?

- 1) F:\Литература \ЕГЭ\Demo
- 2) F:\Литература\Подготовка\ЕГЭ
- 3) A:\ Подготовка\ЕГЭ
- 4) F:\Литература\Подготовка

9. В некотором каталоге хранился файл Задание С3.doc. После того как в этом каталоге создали подкаталог и переместили в созданный подкаталог файл Задание С3.doc, полное имя файла стало

D:\ЕГЭ\Информатика\Части\Часть В\Задание С3.doc. Каково имя вновь созданного каталога?

- 1) Часть С
- 2) Часть В
- 3) Части
- 4) Информатика

10. В некотором каталоге хранился файл Задание ege.htm. После того как в этом каталоге создали подкаталог и переместили в созданный подкаталог файл ege.htm, полное имя файла стало

A:\Exam\Information\ECDL\EXAM\ege.htm. Каково имя вновь созданного подкаталога?

- 1) EGE
- 2) Information
- 3) ECDL
- 4) EXAM

11. Перемещаясь из одного каталога в другой, пользователь последовательно посетил каталоги pupils, class, school, c:\, doc, class, 8a. Каково полное имя каталога, из которого начал перемещение пользователь?

Примечание. При каждом перемещении пользователь либо спускался в каталог на уровень ниже, либо поднимался на уровень выше.

- 1) C:\school\doc\class\pupils
- 2) C:\school\doc\class
- 3) C:\doc\class\8a
- 4) C:\school\8a\class\pupils

12. Перемещаясь из одного каталога в другой, пользователь последовательно посетил каталоги OLE, MAK, PP, A:, SOS, MAK, OLE. Каково полное имя каталога, из которого начал перемещение пользователь?

Примечание. При каждом перемещении пользователь либо спускался в каталог на уровень ниже, либо поднимался на уровень выше.

- 1) A:\OLE\MAK\SOS
- 2) A:\SOS\MAK\OLE
- 3) A:\PP\MAK
- 4) A:\PP\MAK\OLE

13. Перемещаясь из одного каталога в другой, пользователь последовательно посетил каталоги EGE, ICT, EGE, USERS, DOC, PUPILS, CLASS. Каково полное имя каталога, из которого начал перемещение пользователь?

Примечание. При каждом перемещении пользователь либо спускался в каталог на уровень ниже, либо поднимался на уровень выше.

- 1) C:\USER\DOC\USERS\EGE\ICT
- 2) C:\USER\DOC\USERS\EGE
- 3) C:\USER\DOC\PUPILS\CLASS
- 4) C:\USER\DOC\PUPILS\CLASS\USERS\EGE

14. Перемещаясь из одного каталога в другой, пользователь последовательно посетил каталоги ЛИТЕРАТУРА, XX ВЕК, ПИСАТЕЛИ, XX ВЕК, ЛИТЕРАТУРА, C:\, XX ВЕК, ПОЭТЫ. Каково полное имя каталога, из которого начал перемещение пользователь?

Примечание. При каждом перемещении пользователь либо спускался в каталог на уровень ниже, либо поднимался на уровень выше.

- 1) C:\ЛИТЕРАТУРА\XX ВЕК\ПОЭТЫ
- 2) C:\XX ВЕК\ПОЭТЫ
- 3) C:\ЛИТЕРАТУРА\XX ВЕК\ПИСАТЕЛИ
- 4) C:\ЛИТЕРАТУРА\ПОЭТЫ

15. Перемещаясь из одного каталога в другой, пользователь последовательно посетил каталоги ЕГЭ, МАТЕМАТИКА, А:\, ПРЕДМЕТЫ, МАТЕМАТИКА, ПРЕДМЕТ, АЛГЕБРА. Каково полное имя каталога, из которого начал перемещение пользователь?

Примечание. При каждом перемещении пользователь либо спускался в каталог на уровень ниже, либо поднимался на уровень выше.

- 1) A:\ЕГЭ\ПРЕДМЕТ\МАТЕМАТИКА\АЛГЕБРА
- 2) A:\МАТЕМАТИКА\ЕГЭ
- 3) A:\ПРЕДМЕТ\МАТЕМАТИКА\АЛГЕБРА
- 4) A:\ПРЕДМЕТЫ\АЛГЕБРА

ТЕХНОЛОГИЯ ОБРАБОТКИ ГРАФИЧЕСКОЙ ИНФОРМАЦИИ

Растровое представление

Графическая информация представляется в виде изображения, которое формируется из точек (пикселей). Растр — прямоугольная сетка пикселей на экране монитора. **Пиксел(ь)** — минимальный элемент раstra.

Некоторые форматы растровых файлов: bmp, gif, jpeg.

Число цветов (то есть количество цветов в палитре), воспроизводимых на экране монитора (**P**), и число бит, отводимых в видеопамати на каждый пиксел (**b**), можно найти по формуле $P = 2^b$.

Таблица цветов в палитре:

Количество цветов	1 пиксел
2 (черно-белое)	1 бит
16	4 бита
256	8 бит
65536 (high color)	16 бит
16777216 (true color)	24 бита

Таблицу степеней числа два и схему перевода из одних единиц измерения информации в другие можно найти в разделе «Информация и ее кодирование».

Векторное представление

Графическое изображение формируется из *объектов* (линий, прямоугольников, окружностей и т.д.), которые называются *графическими примитивами*.

Некоторые форматы векторных файлов: wmf, eps, cjm.

Растровая сетка в этом случае совпадает с координатной сеткой.

Графическая кодировка RGB

Для фона и текста задаются цвета с использованием шестнадцатеричного трехбайтового кода. Это объясняется тем, что каждая цифра шестнадцатеричного числа кодируется двоичной тетрадой, состоящей из 4 двоичных цифр. Поэтому шестнадцатеричный код длиной в 6 цифр имеет 24 двоичных разряда (бита) или 3 байта. Более подробно об этом можно прочитать в разделе «Информация и ее кодирование».

Перед кодом цвета в HTML ставится символ «#». Например, #00FFFF — голубой цвет (aqua). Записи bgcolor=#00FFFF и bgcolor= aqua эквивалентны.

Кодирование цвета выполняется по распространенной системе RGB (Red, Green, Blue):

#FF0000	Red (красная составляющая)
#00FF00	Green (зеленая составляющая)
#0000FF	Blue (синяя составляющая)

Из курса физики (раздел «Оптика») известно, что любой цвет можно представить смешением трех основных цветов (красного, зеленого и синего). В зависимости от «веса» составляющих результирующий цвет меняется в широких пределах.

Каждая из трех составляющих кодируется двумя шестнадцатеричными цифрами и может принимать значения от 00 до FF в зависимости от интенсивности.

Число цветов, которое можно получить, смешивая красный, зеленый и синий, зависит от ширины диапазона (от 0 до 255), в котором может меняться каждая из трех основных составляющих.

В приведенной системе кодирования каждая из составляющих цвета задается одним байтом (8 битами), то есть может принимать 256 разных значений (по вышеприведенной

формуле $2^8 = 256$). Тогда число всех цветов в этой кодировке: $256 \times 256 \times 256 = 16777216$ или $2^8 \times 2^8 \times 2^8 = 2^{24}$.

Имя	Код	Описание
Black	#000000	Черный
White	#FFFFFF	Белый
Red	#FF0000	Красный
Green	#00FF00	Зеленый
Blue	#0000FF	Синий

Другие способы графического кодирования

Существуют и другие способы задания цвета. Например, цветовая модель CMYK (Cyan — голубой, Magenta — пурпурный, Yellow — желтый, black — черный) является альтернативой RGB для печатающих устройств, хотя охватывает меньший объем цветов.

Существует цветовая модель HSB (Hue — цветовой оттенок, Saturation — насыщенность, Brightness — яркость). Для отображения на мониторе компьютера модель HSB преобразуется в модель RGB, а для печати на принтере — в CMYK.

Кодирование звука

Сигнал — это способ передачи информации. Бывают *аналоговые* и *дискретные* (цифровые). Качество дискретного сигнала зависит от частоты дискретизации и разрешения звуковой карты.

Разрешение (R) — это количество распознаваемых дискретных уровней сигналов.

Частота дискретизации (v) — это количество выборок в секунду. Одно измерение в секунду соответствует частоте 1 Гц (Герц). 1000 измерений в секунду — 1 КГц.

Длительность звучания (t) — измеряется в секундах.

Формула для расчета цифрового моноаудиофайла:
 $N = vtR$.

Например, надо оценить объем моноаудиофайла длительностью звучания 10 с при частоте дискретизации 22,05 КГц и разрешении 8 бит. Формула принимает вид: $N = 8 \cdot 10 \cdot (22,05 \cdot 1000)$. Из этого следует, что объем искомого файла равен 220500 байт, или 215,332 Кб.

ЗАДАНИЯ

Часть 1

При выполнении заданий этой части необходимо выбрать один верный ответ из представленных четырех вариантов и поставить знак «х» в клеточку, номер которой соответствует номеру выбранного вами ответа.

1. Для кодирования цвета фона страниц Интернета используется атрибут `bgcolor="#XXXXXX"`, где в кавычках задается шестнадцатеричное значение интенсивности цветных компонентов 24-битовой RGB-модели. Какой цвет будет у страницы, заданной тэгом

`<body bgcolor="#000000">?`

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

2. Пользователь хочет создать страницу Интернета с красным фоном. Он пользуется атрибутом `bgcolor="#XXXXXX"`, где в кавычках задается шестнадцатеричное значение интенсивности цветных компонентов 24-битовой RGB-модели. Какой тэг он должен написать?

- 1) `<body bgcolor="#FFFFFF">`
- 2) `<body bgcolor="#FF0000">`
- 3) `<body bgcolor="#00FF00">`
- 4) `<body bgcolor="#0000FF">`

3. Двое учащихся 10 класса совместно делали сайт, состоящий из нескольких Web-страниц Интернета. Первый сде-

лал фон с помощью тэга `<body bgcolor="#000000">`, а второй тут же изменил его на `<body bgcolor="#FFFFFF">`. Как поменялся цвет фона?

- 1) с красного на зеленый
 - 2) с белого на красный
 - 3) с черного на белый
 - 3) с белого на черный
4. Ученик создал Web-страницу Интернета и задал сначала фон красным, а потом изменил его на синий. Как изменился вид атрибута `bgcolor` для тэга `<body>`?
- 1) с `<body bgcolor="#00FF00">` на `<body bgcolor="#FFFFFF">`
 - 2) с `<body bgcolor="#FF0000">` на `<body bgcolor="#0000FF">`
 - 3) с `<body bgcolor="#0000FF">` на `<body bgcolor="#00FF00">`
 - 4) с `<body bgcolor="#000000">` на `<body bgcolor="#0000FF">`
5. В какой системе цветопередачи палитра цветов задается шестнадцатеричными значениями цветовых компонент?
- 1) HSB
 - 2) CMYK
 - 3) RGB
 - 4) WBRK
6. В какой системе цветопередачи палитра цветов формируется путем сложения красного, зеленого и синего цветов?
- 1) HSB
 - 2) RGB
 - 3) WBRK
 - 4) CMYK
7. В какой системе цветопередачи палитра цветов формируется путем наложения голубой, желтой, пурпурной и черной красок?
- 1) HSB
 - 2) RGB
 - 3) WBRK
 - 4) CMYK

8. В какой системе цветопередачи палитра цветов формируется путем установки значений оттенка цвета, насыщенности и яркости?
- 1) HSB
 - 2) RGB
 - 3) WBRK
 - 4) CMYK
9. Укажите минимальный объем памяти (в килобайтах), достаточный для хранения любого растрового изображения размером 128×128 пикселей, если известно, что в изображении используется палитра из 256 цветов?
- 1) 256
 - 2) 8
 - 3) 16
 - 4) 128
10. Укажите минимальный объем памяти (в килобайтах), достаточный для хранения любого растрового изображения размером 64×64 пикселей, если известно, что в изображении используется палитра из 65 536 цветов?
- 1) 8
 - 2) 16
 - 3) 64
 - 4) 128
11. Для хранения растрового изображения размером 64×128 пикселей отвели 8 килобайт памяти. Какое максимально возможное число цветов в палитре изображения?
- 1) 16
 - 2) 2
 - 3) 256
 - 4) 65 536
12. Для хранения растрового изображения размером 128×256 пикселей отвели 4 килобайта памяти. Какое максимально возможное число цветов в палитре изображения?
- | | |
|-------|-----------|
| 1) 16 | 3) 256 |
| 2) 2 | 4) 65 536 |

13. Сколько информации (в килобайтах) содержится в картинке экрана с разрешающей способностью 512×768 пикселей и 16 цветами?
 - 1) 192
 - 2) 400
 - 3) 536
 - 4) 1024
14. Сколько информации (в килобайтах) содержится в картинке экрана с разрешающей способностью 256×1280 пикселей и 256 цветами?
 - 1) 256
 - 2) 320
 - 3) 580
 - 4) 768
15. Растровый файл, содержащий черно-белый рисунок, имеет объем полтора килобайта. Какой размер будет иметь рисунок в пикселях?
 - 1) 1024
 - 2) 10 240
 - 3) 12 288
 - 4) 65 536
16. Файл, содержащий нецветное растровое изображение, имеет объем 768 байт. Каков размер изображения в пикселях?
 - 1) 5762
 - 2) 6144
 - 3) 7281
 - 4) 9600
17. В процессе преобразования растрового графического файла количество цветов уменьшилось с $16\,777\,216$ до 256. Во сколько раз уменьшился информационный объем файла?
 - 1) 5
 - 2) 2
 - 3) 3
 - 4) 4

- 18.** В процессе преобразования растрового графического файла количество цветов увеличилось с 16 до 65 536. Во сколько раз увеличился информационный объем файла?
- 1) 5
 - 2) 2
 - 3) 3
 - 4) 4
- 19.** Изначально растровое изображение имело палитру 65 536 цветов. В процессе его преобразования информационный объем уменьшился в 2 раза. Сколько цветов стало в палитре?
- 1) 16
 - 2) 2
 - 3) 256
 - 4) 65 536 (не изменился)
- 20.** Изначально растровое изображение имело палитру 2 цвета. В процессе его преобразования информационный объем увеличился в 8 раз. Сколько цветов стало в палитре?
- 1) 16
 - 2) 2
 - 3) 256
 - 4) 65 536

ТЕХНОЛОГИЯ ОБРАБОТКИ ИНФОРМАЦИИ В ЭЛЕКТРОННЫХ ТАБЛИЦАХ (НА ПРИМЕРЕ EXCEL)

EXCEL является наиболее распространенной программой для работы с электронными таблицами, и только она входит в школьный курс информатики, поэтому все нижеприведенные примеры ориентированы именно на этот программный продукт.

Рабочее поле *Excel* разделено линиями по вертикали на *столбцы* (обозначаются латинскими буквами *A, B, C, D, ..., IV*, всего их 256), а по горизонтали на *строки* (обозначаются числами *1, 2, 3, ..., 65536*).

Пересечение строк и столбцов образует клетки, называемые *ячейками* таблицы. Каждая ячейка имеет свой *адрес*, состоящий из имени столбца и номера строки, например: *B3, \$DA\$108, P\$57*.

В формулах могут использоваться ссылки на адреса ячеек. Существуют два основных типа ссылок: относительные и абсолютные. Различия между абсолютными и относительными ссылками проявляются при копировании формулы из активной ячейки в другие ячейки.

Относительные ссылки. При перемещении или копировании формулы из активной ячейки относительные ссылки автоматически изменяются в зависимости от положения ячейки, в которую скопирована формула. При смещении положения ячейки на одну строку в формуле увеличиваются на единицу номера строк, а при перемещении на один столбец на одну букву смещаются имена столбцов.

Так, при копировании формулы из активной ячейки *C1*, содержащей относительные ссылки на ячейки *A1* и *B1*, в ячейку *D2* имена столбцов и номера строк в формуле изме-

няться на один шаг соответственно вправо и вниз. При копировании формулы в ячейку Е3 имена столбцов и номера строк в формуле изменятся на два шага соответственно вправо и вниз и т.д.

Таблица 1

Относительные ссылки

	А	В	С	Д	Е
1			=A1*B1		
2				=B2*C2	
3					=C3*D3

Абсолютные ссылки. Абсолютные ссылки в формулах используются для указания фиксированного адреса ячейки. При перемещении или копировании формулы абсолютные ссылки не изменяются. В абсолютных ссылках перед неизменяемым именем столбца и номером строки ставится знак доллара (например, \$A\$1).

Так, при копировании формулы из активной ячейки С1, содержащей абсолютные ссылки на ячейки \$A\$1 и \$B\$1, значения столбцов и строк в формуле не изменятся.

Таблица 2

Абсолютные ссылки

	А	В	С	Д	Е
1			=\$A\$1*\$B\$1		
2				=\$A\$1*\$B\$1	
3					=\$A\$1*\$B\$1

Смешанные ссылки. В формуле можно использовать смешанные ссылки, в которых координата столбца относительная, а строки — абсолютная (например, A\$1), или, наоборот, координата столбца абсолютная, а строки — относительная (например, \$B1).

Таблица 3

Смешанные ссылки

	A	B	C	D	E
1			=A\$1*\$B1		
2				=B\$1*\$B2	
3					=C\$1*\$B3

Несколько ячеек можно объединить в группу и записать их через двоеточие:

A1:C3 — группа ячеек (A1, B1, C1, A2, B2, C2, A3, B3, C3).

Построение диаграмм и графиков

Электронные таблицы позволяют визуализировать данные, размещенные на рабочем листе, в виде **диаграммы**. Диаграмма наглядно отображает зависимости между данными, что облегчает восприятие и помогает при анализе и сравнении данных.

Диаграмма (график) — наглядное графическое представление числовых данных.

Диаграммы предназначены для сравнения нескольких величин или нескольких значений одной величины и слежения за изменением их значений и т.п. Они могут быть различных типов, которые представляют данные в разной форме.

Типы диаграмм:

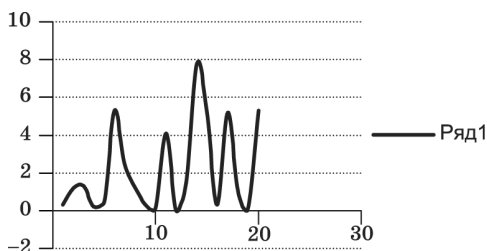
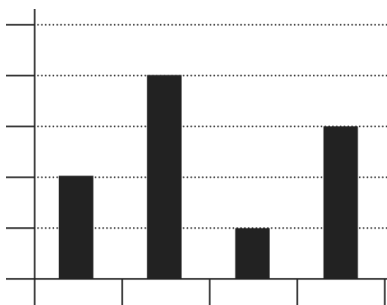


График — для слежения за изменением нескольких величин при переходе от одной точки к другой.



Гистограмма (столбчатая диаграмма) — для сравнения нескольких величин в нескольких точках.



Круговая — для сравнения нескольких величин в одной точке.

Формулы и функции в EXCEL

Формула — это выражение, состоящее из операндов, соединенных знаками операций. Формула в Excel должна начинаться со знака = (равно).

Ввод функции также начинается со знака равно. После имени функции в круглых скобках указывается список аргументов (числа, текст, адреса ячеек и выражения, содержащие другие функции), разделенных точкой с запятой.

Наиболее часто используемые функции Excel:

СУММ(B2:B5)	Вычисление суммы числовых значений для диапазона ячеек B2:B5
СРЗНАЧ(B2:B5)	Вычисление среднего значения для диапазона ячеек B2:B5

МАКС(B2:B5)	Вычисление максимального значения для диапазона ячеек B2:B5
МИН(B2:B5)	Вычисление минимального значения для диапазона ячеек B2:B5
СЧЕТ(B2:B5)	Подсчет общего количества чисел для диапазона ячеек B2:B5

ЗАДАНИЯ

Часть 1

При выполнении заданий этой части необходимо выбрать один верный ответ из представленных четырех вариантов и поставить знак «х» в клеточку, номер которой соответствует номеру выбранного вами ответа.

1. В электронной таблице значение формулы =СУММ(A1:A3) равно 8. Чему равно значение ячейки A4, если значение формулы =СРЗНАЧ(A1:A4) равно 3?
1) 5
2) 2
3) 8
4) 4
2. В электронной таблице значение формулы =СУММ(A1:A4) равно 13, а значение формулы =СРЗНАЧ(A1:A5) равно 3. Чему равно значение формулы =СУММ(A1:A5)?
1) 15
2) 16
3) 24
4) 28
3. В электронной таблице значение формулы =СУММ(B1:B5) равно 24. Чему равно значение ячейки B6, если значение формулы =СРЗНАЧ(B1:B6) равно 4?
1) 1 2) 2 3) 0 4) 4

Примечание. Символ \$ в формуле обозначает абсолютную адресацию.

1) $= 4 * \$B\$6 - \$C3$

2) $= 2 * \$B\$4 - \$C3$

3) $= 2 * \$C\$4 - \$D1$

4) $= 2 * \$C\$6 - \$D3$

9. Дан фрагмент электронной таблицы:

	A	B	C
1	10	30	$=A1 + B\$1 - 10$
2	20	40	

Чему станет равным значение ячейки C2, если в нее скопировать формулу из ячейки C1?

Примечание. Символ \$ в формуле обозначает абсолютную адресацию.

1) 40

2) 50

3) 20

4) 30

10. Дан фрагмент электронной таблицы:

	A	B	C	D
1	10	20	30	50
2	40	10	$=A1 + C\$1 - \$B1$	
3	20	30		

Чему станет равным значение ячейки D3, если в нее скопировать формулу из ячейки C2?

Примечание. Символ \$ в формуле обозначает абсолютную адресацию.

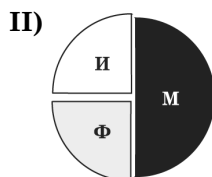
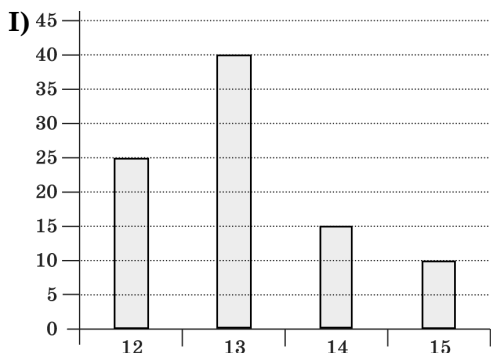
1) 60

2) 30

3) 50

4) 40

11. Во всех школах района работают учителя физики (Ф), математики (М) и информатики (И). Каждый учитель имеет разряд, не меньший 12 и не больший 15. На диаграмме I отражено количество учителей с различными разрядами, а на диаграмме II — распределение учителей по специальностям.



Имеются четыре утверждения:

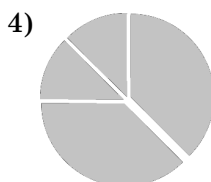
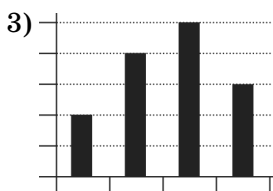
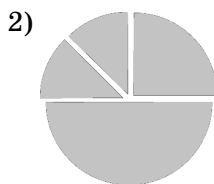
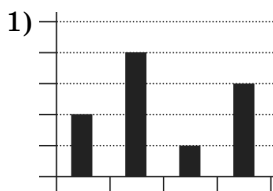
- А) Все физики могут иметь 15-й разряд
 Б) Все учителя с 13-м разрядом могут быть математиками
 В) Все математики могут иметь 14-й разряд
 Г) Все учителя с 13-м разрядом могут быть информатиками
 Какое из этих утверждений следует из анализа этих диаграмм?

- 1) А 2) Б 3) В 4) Г

12. Дан фрагмент электронной таблицы:

	А	В
1	=B2 – 1	0
2	=A1 + B1	2
3	=A2*B2 + 1	
4	=A3	

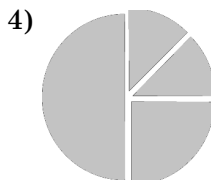
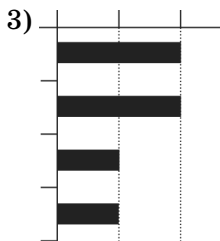
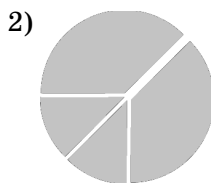
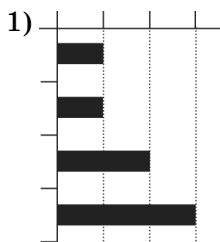
После выполнения вычислений была построена диаграмма по значениям диапазона ячеек А1:А4. Укажите получившуюся диаграмму.



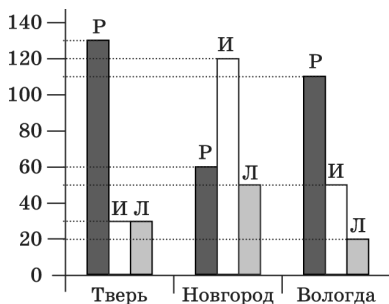
13. Дан фрагмент электронной таблицы:

	A	B	C	D
1		3	5	
2	$= C1 + B1$	$= (B1 + C1)/2$	$= A2 - B2$	$= C1 + C2 + B1$

После выполнения вычислений была построена диаграмма по значениям диапазона ячеек A2:D2. Укажите получившуюся диаграмму.

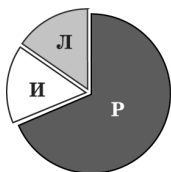


- 14.** На диаграмме показано количество призеров олимпиады по русскому языку (Р), литературе (Л) и истории (И) в трех городах России.

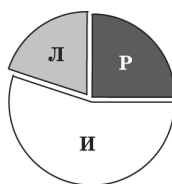


Какая из диаграмм правильно отражает соотношение призеров из всех городов по каждому предмету?

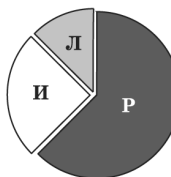
1)



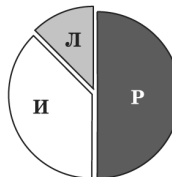
2)



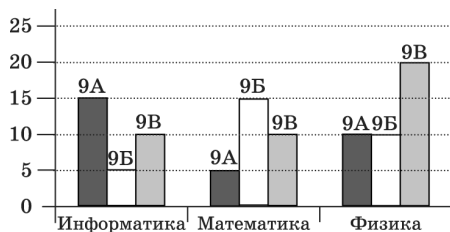
3)



4)

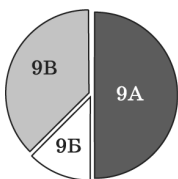


- 15.** На диаграмме показано количество призеров школьной олимпиады по информатике, математике и физике среди 9-х классов:

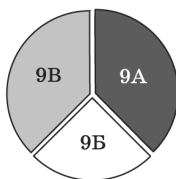


Какая из диаграмм правильно отражает соотношение призеров по классам по всем предметам?

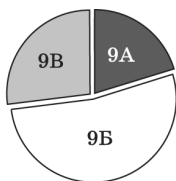
1)



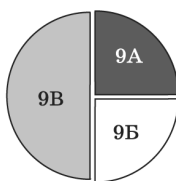
2)



3)



4)



ТЕХНОЛОГИЯ ХРАНЕНИЯ, ПОИСКА И СОРТИРОВКИ В БАЗАХ ДАННЫХ (НА ПРИМЕРЕ ACCESS)

ACCESS является наиболее распространенной программой для работы с базами данных, и только она входит в школьный курс информатики, поэтому все нижеприведенные примеры ориентированы именно на этот программный продукт.

В реляционной базе данных (например, Access) информация хранится в виде таблиц, связанных между собой. Каждая таблица должна иметь свое имя.

Запись — это строка таблицы.

Поле — это столбец таблицы.

Таблица — информационная модель реальной системы.

Запись содержит информацию об одном конкретном объекте.

Поле содержит определенные характеристики объектов.

Основной элемент БД Access — запись.

Главный ключ — это поле или совокупность полей, которые однозначно определяют запись в таблице. Связь между таблицами осуществляется посредством значений одного или нескольких совпадающих полей.

Базы данных, созданные в Access, имеют расширение **mdb**.

Основные объекты базы данных:

Таблица — объект, предназначенный для хранения данных в виде записей и полей.

Форма — объект, предназначенный для ввода данных в таблицу, для открытия других форм и отчетов (кнопочные формы), а также для создания маски.

Запрос — объект, позволяющий получить нужные данные из одной или нескольких таблиц, т.е. для выборки записей, обновления таблиц и включения в них новых записей.

Отчет — объект для отображения итоговых данных из таблиц и запросов в удобном для просмотра виде.

При создании запроса к базе данных используются различные условия отбора данных из заданной таблицы.

Рассмотрим примеры записи условий отбора данных.

Условие отбора данных	Результат выполнения отбора данных
Р?ка	Рука, река
К*т	Кот, канат, красный молот
*ой	132-й, голубой
* анг *	Манго, лангуст
*.10.99	Все даты в октябре 1999 года
>10	Все числа больше 10
Between 10 and 20	Числа от 10 до 20
>= «Иванов»	Все фамилии от Иванова до конца алфавита
«Франция» OR «Канада»	Франция или Канада
Date ()	Отбор записей, содержащих текущую дату
Between Date () AND Date () 5	В течение последних 5 дней

Для осуществления сложного запроса (отбора данных из нескольких полей) условия отбора записываются для каждого поля.

Когда два условия соединены союзом **ИЛИ**, считаются все варианты, удовлетворяющие 1-му и 2-му условиям. Но если одна запись удовлетворяет обоим условиям, тогда ее надо считать только один раз.

Когда два условия соединены союзом И, считаются все варианты, удовлетворяющие обоим условиям.

Например, разберем задачу, где в табличной форме представлен фрагмент базы данных о результатах тестирования учащихся (используется стобалльная шкала):

Фамилия	Пол	Математика	Русский язык	Химия	Информатика	Биология
Аралова	ж	84	66	44	42	75
Сорокин	м	45	64	49	75	33
Павленко	м	59	70	58	79	85
Роднина	ж	81	73	46	83	77
Алексеева	ж	43	35	79	39	41
Чудова	ж	21	91	82	27	67

Сколько записей в данном фрагменте удовлетворяют условию

«Пол='м' ИЛИ Информатика > Математика»?

В этой задаче необходимо найти все строки, удовлетворяющие первому условию, и все строки, удовлетворяющие второму условию, и сложить их, убрав совпадающие.

Условие 1: «Пол='м'»:

Сорокин	м	45	64	49	75	33
Павленко	м	59	70	58	79	85

Условию удовлетворяют две строки.

Условие 2: «Информатика > Математика»:

Сорокин	м	45	64	49	75	33
Роднина	ж	81	73	46	83	77

Условию удовлетворяют две строки.

Строка с фамилией «Сорокин» является совпадающей в обоих случаях, поэтому считается только один раз, поэтому ответ — 3 строки.

Если условие изменить и поставить вместо союза ИЛИ союз И, то условие примет вид:

«Пол='м' И Информатика > Математика».

В этом случае необходимо будет искать строку, в которой удовлетворяются **оба** условия. Это единственная строка:

Сорокин	м	45	64	49	75	33
---------	---	----	----	----	----	----

Поэтому ответ — 1 строка.

Сортировка в БД

Часто используется и сортировка. Для начала нужно уточнить, что такое сортировка по возрастанию и убыванию.

Сортировка по возрастанию для букв — это от А до Я, от А до Z, а для чисел — это сортировка от наименьшего числа до наибольшего.

Сортировка по убыванию для букв — это от Я до А, от Z до А, а для чисел — это сортировка от наибольшего числа до наименьшего.

ЗАДАНИЯ

Часть 1

При выполнении заданий этой части необходимо выбрать один верный ответ из представленных четырех вариантов и поставить знак «х» в клеточку, номер которой соответствует номеру выбранного вами ответа.

1. Ниже в табличной форме представлен фрагмент базы данных о результатах тестирования учащихся (используется столбальная шкала).

Фамилия	Пол	Математика	Русский язык	История	Информатика	Физика
Иванов	м	71	36	28	39	49
Смелов	м	59	64	61	31	54
Асрян	ж	65	71	67	23	65
Смелко	ж	32	55	41	54	71
Петрова	ж	90	69	74	73	84
Федорчук	м	55	58	39	39	60

Сколько записей в данном фрагменте удовлетворяют условию:

«Пол = 'ж' И Физика >= Математика»?

1) 0

3) 3

2) 2

4) 4

2. Ниже в табличной форме представлен фрагмент базы данных о результатах тестирования учащихся (используется стобалльная шкала):

Фамилия	Пол	Математика	Русский язык	История	Информатика	Физика
Иванов	м	71	36	28	39	49
Смелов	м	59	64	61	31	54
Асрян	ж	65	71	67	23	65
Смелко	ж	32	55	41	54	71
Петрова	ж	90	69	74	73	84
Федорчук	м	55	58	39	39	60

Фамилия	Пол	Математика	Русский язык	История	Информатика	Физика
Смелко	ж	32	55	41	54	71
Петрова	ж	90	69	74	73	84
Федорчук	м	55	58	39	39	60

Сколько записей в ней удовлетворяют условию
«Пол = 'м' И Информатика < 39»?

- 1) 1
- 2) 2
- 3) 3
- 4) 0

5. Ниже в табличной форме представлен фрагмент базы данных о результатах тестирования учащихся (используется стобалльная шкала).

Фамилия	Пол	Математика	Русский язык	История	Информатика	Физика
Иванов	м	71	36	28	39	49
Смелов	м	59	64	61	31	54
Асрян	ж	65	71	67	23	65
Смелко	ж	32	55	41	54	71
Петрова	ж	90	69	74	73	84
Федорчук	м	55	58	39	39	60

Сколько записей в ней удовлетворяют условию
«Пол = '?' И Информатика > Русского языка»?

- 1) 0
- 2) 2
- 3) 3
- 4) 4

6. Количество полей в базе данных структуры, представленной таблицей

Фамилия	Пол	Год	Класс	Средний балл
Мишин	м	1990	11	4,62
Ланина	ж	1991	11	4,81
Погосян	м	1991	11	3,72

равно

- 1) 6
- 2) 5
- 3) 3
- 4) 4

7. Количество записей в базе данных структуры, представленной таблицей

Школа	Класс	Фамилия	Балл
445	11	Петрова	64
307	11	Смирнов	72
1495	9	Котов	60

равно

- 1) 5
- 2) 6
- 3) 3
- 4) 4

8. В городской олимпиаде по русскому языку предлагались задачи трех уровней сложности А, В, и С. По итогам олимпиады была составлена таблица, в колонках которой было указано, сколько задач решил участник и общее количество задач. Вот начало таблицы:

Фамилия	А	В	С	Количество задач
Орехов	2	2	1	5

За правильное решение задачи типа А участнику начислялся 1 балл, за решение задачи типа В — 2 балла, за решение задачи типа С — 3 балла. Победитель определялся по сумме баллов, которая у всех участников оказалась разной. Для определения победителя олимпиады достаточно выполнить следующий запрос:

1. Отсортировать таблицу по возрастанию значений поля $A+B+C$ и взять первую строку.
2. Отсортировать таблицу по убыванию значений поля количество задач и взять первую строку.
3. Отсортировать таблицу по возрастанию значений поля $A+B+C$ + количество задач и взять первую строку.
4. Отсортировать таблицу по убыванию значений поля $A+2*B+3*C$ и взять первую строку.

9. В городской олимпиаде по русскому языку предлагались задачи трех уровней сложности А, В, и С. По итогам олимпиады была составлена таблица, в колонках которой было указано, сколько задач решил участник и общее количество задач. Вот начало таблицы:

Фамилия	А	В	С	Количество задач
Орехов	2	2	1	5

За правильное решение задачи типа А участнику начислялся 1 балл, за решение задачи типа В — 2 балла, за решение задачи типа С — 3 балла. Победитель определялся по среднему баллу (то есть общая сумма делилась на количество правильно выполненных задач). Для определения победителя

- 1) отсортировать таблицу по убыванию значений поля $(A+B+C)/(\text{Количество заданий})$ и взять первую строку;
- 2) отсортировать таблицу по убыванию значений поля $(A+B+C)$ и взять последнюю строку;
- 3) отсортировать таблицу по возрастанию значений поля $(A+2*B+3*C)/(\text{Количество заданий})$ и взять первую строку;

- 4) отсортировать таблицу по возрастанию значений поля $(A+2*B+3*C)/(\text{Количество заданий})$ и взять последнюю строку.

10. База данных «Учащиеся» имеет вид:

Имя поля	Тип поля	Примечания
Класс	текстовый	
Фамилия	текстовый	
Имя	текстовый	
Год рождения	числовой	
Пол	текстовый	«М» или «Ж»
Рост	числовой	
Вес	числовой	

Как следует записать условие отбора при фильтрации, которое позволит сформировать список участниц баскетбольной команды?

(Отбираются девушки не ниже 180 см и не моложе 9 класса.)

- 1) «Пол = 'ж' ИЛИ Рост>180 И Класс = 9»
- 2) «Пол = 'ж' И Рост>180 И Класс>= 9»
- 3) «Пол = 'ж' ИЛИ Рост>180 ИЛИ Класс>= 9»
- 4) «Рост>=180 и Класс>9»

11. База данных «Учащиеся» имеет вид:

Имя поля	Тип поля	Примечания
Класс	текстовый	
Фамилия	текстовый	
Имя	текстовый	
Год рождения	числовой	

Имя поля	Тип поля	Примечания
Пол	текстовый	«М» или «Ж»
Рост	числовой	
Вес	числовой	

Как следует записать условие отбора при фильтрации, которое позволит сформировать список для военкомата? (Отбираются школьники 1991 года рождения с ростом от 165 см.)

- 1) «Пол = 'м' ИЛИ Рост>165 И Год рождения >1991»
- 2) «Год рождения >1991 И Пол = 'м' ИЛИ Рост>=165»
- 3) «Рост>=165 И Пол = 'м' И Год рождения =1991»
- 4) «Пол = 'м' И Рост>165 ИЛИ Год рождения =1991»

12. 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	Класс	Вес спортсмена	Взятые рывки	Рывок с попытки	Взятые толчки	Толчок с попытки
Аганян	11	76,5	109,5	3	140,5	2
Колосов	11	78	121,5	3	—	—

Фамилия	Класс	Вес спортсмена	Взятые рывки	Рывок с попытки	Взятые толчки	Толчок с попытки
Махеев	10	77,1	109,5	2	140,5	1
Петров	10	76,5	117	2	123	1
Сидоркин	9	77	114	1	136	2
Шутько	9	79	—	—	139	3

Кто победил в рывке?

- 1) Аганян
- 2) Колосов
- 3) Петров
- 4) Махеев

13. 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	Класс	Вес спортсмена	Взятые рывки	Рывок с попытки	Взятые толчки	Толчок с попытки
Аганян	11	76,5	109,5	3	140,5	2
Колосов	11	78	121,5	3	—	—
Махеев	10	77,1	109,5	2	140,5	1

Фамилия	Класс	Вес спортсмена	Взятые рывки	Рывок с попытки	Взятые толчки	Толчок с попытки
Петров	10	76,5	117	2	123	1
Сидоркин	9	77	114	1	136	2
Шутько	9	79	—	—	139	3

Кто лучше всех толкнул штангу?

- 1) Махеев
- 2) Сидоркин
- 3) Аганян
- 4) Шутько

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	Класс	Вес спортсмена	Взятые рывки	Рывок с попытки	Взятые толчки	Толчок с попытки
Аганян	11	76,5	109,5	3	140,5	2
Колосов	11	78	121,5	3	—	—
Махеев	10	77,1	109,5	2	140,5	1

Фамилия	Класс	Вес спорт- смена	Взя- тые рывки	Рывок с по- пытки	Взя- тые толчки	Толчок с по- пытки
Петров	10	76,5	117	2	123	1
Сидоркин	9	77	114	1	136	2
Шутько	9	79	—	—	139	3

Кто победил в общем зачете (сумме двух упражнений)?

- 1) Петров
- 2) Аганяр
- 3) Махеев
- 4) Сидоркин

15. Какой тип поля целесообразно использовать в поле День рождения в таблице «Мои друзья»

Фамилия	Имя	День рождения	Телефон	E-mail
Иванов	Илья	10.10.1993	88240326462	Ivanov@mail.ru

- 1) Текстовый
- 2) Числовой
- 3) Дата/время
- 4) Счетчик

ТЕЛЕКОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ

Путь к файлу

Адрес файла включает в себя способ доступа к файлу и имя сервера Интернета, на котором находится файл. Чаще всего (но не во всех случаях) схема обращения начинается с имени протокола, после которого следует двоеточие и двойной слэш (только для on-line сервисов, но именно о них идет речь в некоторых заданиях ЕГЭ). Тогда, например, получаем путь:

`http://`

Далее ставим имя сервера (или его IP-адрес) и правый слэш:

`http://ftp.net/`

И, наконец, записываем само имя файла с расширением:

`http://ftp.net /www.txt`

Конечно, возможно, что файл располагается не прямо на самом сервере, а в каталоге на данном сервере, и тогда путь к файлу удлинится и примет, например, такой вид:

`http://ftp.net /catalog/www.txt`

или такой:

`http://ftp.net /catalog/1/index/www.txt`

Запросы к поисковым серверам

Часто в учебниках по информатике для обозначения логической операции «ИЛИ» в запросе к абстрактному поисковому серверу используется символ `|`, а для логической операции «И» — символ `&`.

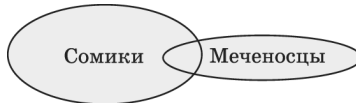
В соответствии с понятиями теории множеств запрос $X \& Y$ будет соответствовать пересечению множеств X и Y , а запрос $X \mid Y$ будет соответствовать объединению множеств X и Y .

Исходя из вышесказанного, можно сделать вывод, что по запросу с логической операцией «ИЛИ» страниц может быть найдено больше, чем по запросу с логической операцией «И». Конечно, существуют примеры, когда вообще нет страниц по одной части запроса или обе части совпадают, но они не рассматриваются в заданиях ЕГЭ.

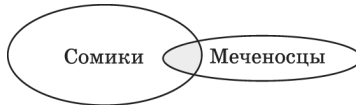
Например, по запросу «сомики | меченосцы» будут найдены страницы, на которых упоминаются сомики и меченосцы вместе, а также страницы, где упомянуты только сомики, и страницы, где упомянуты только меченосцы. По запросу «сомики & меченосцы» будут найдены только те страницы, где упомянуты сомики и меченосцы вместе. По сравнению с предыдущим запросом этот найдет меньше страниц.

Графически это может выглядеть так:

«сомики | меченосцы»



«сомики & меченосцы»



Чем больше условий соединено с помощью логической операции «И», тем меньше страниц найдет поисковый сервер.

IP-адресация в сети

Каждый компьютер, подключенный к Интернету, имеет свой 32-битный (в двоичной системе) IP-адрес.

Система IP-адресации учитывает то, что Интернет является сетью сетей, а не объединением отдельных компьютеров в небольшую локальную сеть. IP-адрес состоит из номера (адреса) сети и номера (адреса) компьютера в данной сети.

Для обеспечения гибкости при распределении **IP**-адресов, в зависимости от количества компьютеров в сети, сети (в данном случае — диапазоны адресов отдельных сетей) разделяются на три класса: А, В, С. Первые биты номера сети отводятся для идентификации класса.

IP-адрес записывают 4 числами (1 байт — одно число), разделенными точками. Каждое число таким образом попадает в диапазон от 0 до 255. Адрес читается справа налево.

По первому числу **IP**-адреса компьютера можно определить его принадлежность к сети того или иного класса.

Адреса класса А — первый октет адреса от 0 до 127;

Адреса класса В — первый октет адреса от 128 до 191;

Адреса класса С — первый октет адреса от 192 до 223.

Например, **IP**-адрес может быть записан как **195.34.32.11**.

ЗАДАНИЯ

Часть 2

Ответом к заданиям этой части является набор символов, которые следует записать справа от номера соответствующего задания начиная с первой клеточки. Каждый символ пишете в отдельной клеточке в соответствии с приведенными образцами (если имеются бланки ЕГЭ).

- 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	(Конституция & Россия) (Конституция & СССР)
2	Конституция & Россия СССР
3	Конституция Россия СССР
4	Конституция (Россия & СССР)

2. 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	(Франция Россия) & Революция
2	Революция & Франция Россия
3	Революция Россия
4	Франция & Россия & Революция

3. 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	Кормление & Собаки
2	(Уход & Кошки) Собаки
3	Кормление & Уход & Кошки & Собаки
4	Уход & Кошки & Собаки

4. 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	Экзамен ЕГЭ Информатика
2	Экзамен & ЕГЭ & Информатика
3	Экзамен ЕГЭ
4	Экзамен ЕГЭ Информатика Литература

5. 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	Сок & Апельсин & Яблоко
2	Сок & Апельсин
3	Сок Нектар
4	Сок

6. 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	Принтеры Сканеры Производители Продавцы
2	(Производители & Принтеры) Сканеры
3	Принтеры & Сканеры & Продавцы
4	Сканеры & Производители

7. 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	Графика Редактор
2	Графика Редактор Векторная
3	Графика Редактор Векторная Растровая
4	Графика & Векторная & Растровая

8. 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	Учебник & Практикум
2	(Программирование Информатика) & (Учебник Практикум)
3	Программирование Информатика Учебник
4	Учебник (Программирование & Информатика)

9. 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	Страна Столица
2	Страна Столица Москва
3	Страна & Столица & Москва
4	Страна & Столица & Москва & Париж

10. 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	Изготовление Торт
2	Украшение Изготовление (Торт & Кекс)
3	(Украшение & Изготовление & Торт) Кекс
4	(Изготовление Кекс) & (Украшение Торт)

- 11.** Миша записал IP-адрес своего компьютера в десятичном коде, а потом переписал в двоичном коде. Его друг Петя решил проверить. В ответе укажите фрагменты IP-адреса в двоичном коде:

204.36.59.192

00100100	00111011	11001100	11000000
А	Б	В	Г

- 12.** Коля, выполняя домашнее задание по ИКТ, записал IP-адрес своего компьютера, но в школе оказалось, что адрес надо было записать в двоичном коде. Помогите восстановить последнюю часть IP-адреса из фрагментов, которые обозначены буквами А, Б, В, Г. В ответе укажите только последовательность букв.

1001010	...	11111011	00110001
74	177	251	49

00	01	10	11
А	Б	В	Г

- 13.** Рома записал IP-адрес своего компьютера на листок бумаги, который оставил на столе. Потом он ушел гулять, а когда вернулся, то увидел, что записку с IP-адресом разорвал младший брат. Фрагменты записки обозначены буквами А, Б, В, Г. Помогите Роме восстановить IP-адрес. В ответе укажите последовательность букв, обозначающих фрагменты в порядке, соответствующем IP-адресу.

7.68	56	51.2	.2
А	Б	В	Г

- 14.** Учитель предложил Пете восстановить IP-адрес школьного компьютера, когда известен только набор двоичных цифр. Буквами А, Б, В, Г обозначены фрагменты IP-адреса:

реса в десятичной системе счисления, но без учета точек, разделяющих части IP-адреса. Исходный набор двоичных цифр выглядел так:

10101100001101100111101110111110

В ответе укажите последовательность букв, обозначающих фрагменты, в порядке, соответствующем IP-адресу без учета точек, которые служат разделителями IP-адреса на части.

123	172	190	54
А	Б	В	Г

15. Костя, Рома, Паша и Андрей записали различные части IP-адреса.

Костя написал: 1011000. Рома — 11011011. Паша — 000011110. Что записал Андрей, если известно, что десятичное значение этого фрагмента IP-адреса — 108. В ответе укажите двоичную последовательность части IP-адреса без учета точки, отделяющей этот фрагмент IP-адреса.

16. Доступ к файлу http.doc, находящемуся на сервере www.org, осуществляется по протоколу http. В таблице фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла

http	/	.doc	://	htp	www	.org
А	Б	В	Г	Д	Е	Ж

17. Доступ к файлу http.txt, находящемуся на сервере edu.com, осуществляется по протоколу ftp. В таблице фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла

://	http	.txt	ftp	edu	/	.com
А	Б	В	Г	Д	Е	Ж

18. Доступ к файлу foto.png, находящемуся на сервере com.net, осуществляется по протоколу http. В таблице фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла

.png	com	/	://	.net	http	foto
А	Б	В	Г	Д	Е	Ж

19. Доступ к файлу ege.ppt, находящемуся на сервере ftpp.ru, осуществляется по протоколу ftp. В таблице фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла

ftp	ege	://	.ru	.ppt	/	ftpp
А	Б	В	Г	Д	Е	Ж

20. Доступ к файлу http.htm, находящемуся на сервере mail.uk, осуществляется по протоколу ftp. В таблице фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла

mail	/	http	.uk	ftp	.htm	://
А	Б	В	Г	Д	Е	Ж

ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

Технологии программирования, т.е. написание программ, требуют глубоких и обширных знаний по теме. В данном сборнике не представляется возможным рассмотреть эту тему подробно, поэтому приводим только список учебников, которыми можно воспользоваться, чтобы научиться программировать.

1. Введение в программирование. Для среднего и старшего школьного возраста / Авт.-сост. В.А. Гольденберг. — Мн.: ООО «Харвест», 1998. — 528 с.

2. *Долинский М.С.* Решение сложных и олимпиадных задач по программированию. — СПб.: Питер, 2007. — 366 с. (<http://www.alleng.ru/d/comp/comp05.htm>)

3. *Златопольский Д.М.* Сборник задач по программированию. — 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2007. — 240 с. (<http://www.alleng.ru/d/comp/comp19.htm>)

4. Информатика и ИКТ. Практикум по программированию. 10—11 классы. Базовый уровень / Под. ред Н.В. Макаровой. — СПб.: Питер, 2008. — 176 с.

5. *Мельникова О.И., Бонюшкина А.Ю.* Начала программирования на языке QBasic: Учебное пособие. — М.: ЭКОМ, 1998. — 304 с.

6. Основы программирования на Visual Basic 6.0 - <http://psbatishev.narod.ru/vb/v000.htm>

7. Сайт «Задачи по информатике» (<http://www.problems.ru/inf/>)

8. *Фаронов В.* Turbo Pascal: Учебное пособие. — СПб.: Питер, 2008. — 367 с.

9. *Фаронов В.* Turbo Pascal 7.0: Начальный курс: Учебное пособие. — СПб.: Питер, 2007. — 576 с.

10. *Уоллес В.* Основы программирования для «чайников». — М.: Изд. дом «Вильямс», 2006. — 384 с.

ЗАДАНИЯ

Часть 3

На бланке запишите сначала номер задания, а затем — полное решение. Ответы записывайте четко и разборчиво.

1. Требовалось написать программу, которая решает уравнение $ax - b > 0$ 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,x); if a = 0 then if b = 0 then write('любое число') else write('нет решения') else if b = 0 then write('x = 0') else write('x =', -b/a, 'или x =', b/a); end.</pre>	<pre>INPUT a, b, x IF a = 0 THEN IF b = 0 THEN PRINT "любое число" ELSE PRINT «нет решения» ENDIF ELSE IF b = 0 THEN PRINT «x = 0» ELSE PRINT «x =», -b/a, «или x =», b/a END IF END IF END</pre>	<pre>void main(void) {float a,b,x; scanf("%f%f%f", &a,&b,&x); if (a==0) if (b==0) printf("любое число"); else printf («нет решения»); else if (b==0) printf («x = 0»); else printf («x=%f или x=%f», -b/a,b/a); }</pre>

Последовательно выполните три задания:

- 1) приведите пример таких чисел a , b , x , при которых программа неверно решает поставленную задачу;

- 2) укажите, какая часть программы является лишней;
- 3) 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.)

2. Требовалось написать программу, которая находит среднее из трех чисел x , y и z , введенных с клавиатуры. Здесь под средним понимается число, которое меньше другого и больше оставшегося из трех. Числа x , y , z являются натуральными и неравными. Входная информация вводится с клавиатуры, а на выходе должно быть сообщение вида «среднее x ». Программист торопился и написал программу неправильно.

ПРОГРАММА НА БЕЙСИКЕ	ПРОГРАММА НА ПАСКАЛЕ	ПРОГРАММА НА СИ
<pre> INPUT x, y, z IF (x>y) AND (y>z) THEN PRINT "среднее y" IF (y>z) AND (x>z) THEN PRINT "среднее z" IF (y>x) AND (x>z) THEN PRINT "среднее x" END </pre>	<pre> var x, y, z: integer; begin readln(x, y, z); if (x>y) and (y>z) then write('среднее y'); if (y>z) and (z>x) then write('среднее z'); if (y>x) and (x>z) then write('среднее x'); end. </pre>	<pre> void main(void) { int x, y, z; scanf("%d%d%d", &x, &y, &z); if (x>y && y>z) printf("среднее y"); if (y>z && z>x) printf("среднее z"); if (y>x && x>z) printf("среднее x"); } </pre>

Последовательно выполните три задания:

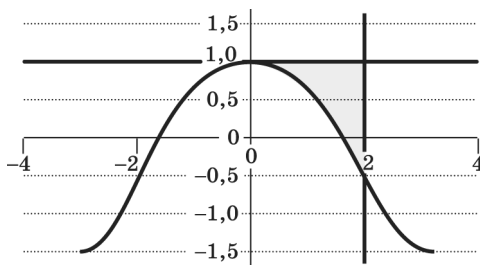
- 1) приведите пример таких чисел x , y , z , при которых программа работает неправильно;
 - 2) укажите, какая часть программы является лишней;
 - 3) укажите, как нужно доработать программу, чтобы не было случаев ее неправильной работы. (Это можно сделать несколькими способами, поэтому можно указать любой способ доработки исходной программы.)
3. Требовалось написать программу, которая решает неравенство $ax+b<0$ относительно x для любых чисел a и b , введенных с клавиатуры. Все числа считаются действительными.

тельными. 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,x); if a=0 then write('любое число') else if a>0 then write('x<','-b/a') else write('x>','-b/a'); end.</pre>	<pre>INPUT a, b, x IF a=0 THEN PRINT "любое число» ELSE IF a>0 THEN PRINT «x<», -b/a ELSE PRINT «x>», -b/a END</pre>	<pre>void main(void) { float a,b,x; scanf("%f%f%f", &a,&b,&x); if (a==0) printf("любое число»); else if (a>0) printf(«x<%f», -b/a); else printf(«x>%f», -b/a); }</pre>

Последовательно выполните три задания:

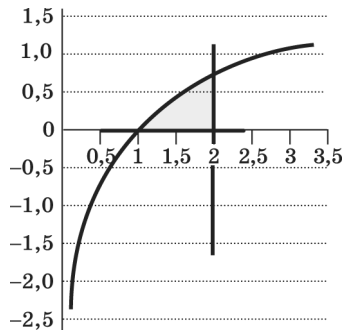
- 1) приведите пример таких чисел a , b , x , при которых программа работает неправильно;
 - 2) укажите, какая часть программы является лишней;
 - 3) укажите, как нужно доработать программу, чтобы не было случаев ее неправильной работы. (Это можно сделать несколькими способами, поэтому можно указать любой способ доработки исходной программы.)
4. Требовалось написать программу, которая вводит с клавиатуры координаты точки на плоскости (x, y — действительные числа) и определяет принадлежность точки заштрихованной 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<=1 then if x>=0 then if y>=cos(x) then write('принадле- жит') else write('не принад- лежит') end. end.</pre>	<pre>INPUT x, y IF y<=1 THEN IF x>=0 THEN IF y>=COS(x) THEN PRINT «принадле- жит» ELSE PRINT «не принад- лежит» ENDIF ENDIF ENDIF END</pre>	<pre>void main(void) { float x,y; scanf("%f%f",&x,&y); if (y<=1) if (x>=0) if (y>=cos(x)) printf(«принадле- жит»); else printf(«не принадле- жит»); } }</pre>

Последовательно выполните следующее:

- 1) приведите пример таких чисел x , y , 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.)
5. Требовалось написать программу, которая вводит с клавиатуры координаты точки на плоскости (x, y — действительные числа) и определяет принадлежность точки заштрихованной области, включая ее границы. Программист торопился и написал программу неправильно.



Последовательно выполните следующее:

- 1) приведите пример таких чисел x , y , 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.)

ПРОГРАММА НА ПАСКАЛЕ	ПРОГРАММА НА БЕЙСИКЕ	ПРОГРАММА НА СИ
<pre>var x,y: real; begin readln(x,y); if y<=1 then if x>=0 then if y>=ln(x) then write('принадле- жит') else write('не принад- лежит') end. end.</pre>	<pre>INPUT x, y IF y<=1 THEN IF x>=0 THEN IF y>=LN(x) THEN PRINT «принадле- жит» ELSE PRINT «не принад- лежит» ENDIF ENDIF ENDIF END</pre>	<pre>void main(void) { float x,y; scanf("%f%f",&x,&y); if (y<=1) if (x>=0) if (y>=ln(x)) printf("принадле- жит"); else printf("не принадле- жит"); } }</pre>

6. Опишите на русском языке или одном из языков программирования алгоритм получения из заданного целочисленного массива размером 30 элементов количества положительных, отрицательных и нулевых значений.
7. Опишите на русском языке или одном из языков программирования алгоритм подсчета суммы элементов, стоящих на четных местах, в массиве из 30 чисел.
8. Опишите на русском языке или одном из языков программирования алгоритм подсчета среднего значения из отрицательных элементов в массиве из 30 чисел.
9. Опишите на русском языке или одном из языков программирования алгоритм подсчета количества элементов, попадающих в заданный интервал, в массиве из 30 чисел.

10. Опишите на русском языке или одном из языков программирования алгоритм подсчета количества элементов, попадающих между номером минимального элемента и номером максимального элемента в массиве из 30 чисел.
11. На вход программе подаются сведения о сдаче экзаменов в формате ЕГЭ учениками 9-х классов некоторой средней школы. В первой строке сообщается количество учеников N , которое не меньше 10, но не превосходит 100, каждая из следующих N строк имеет следующий формат:

`<Фамилия> <Имя> <оценки>`,

где `<Фамилия>` — строка, состоящая не более чем из 20 символов, `<Имя>` — строка, состоящая не более чем из 15 символов, `<оценки>` — через пробел три целых числа, соответствующие оценкам по стобалльной системе. `<Фамилия>` и `<Имя>`, а также `<Имя>` и `<оценки>` разделены одним пробелом. Пример входной строки:

Иванов Петр 64 75 63

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например Borland Pascal 7.0), которая будет выводить на экран фамилии и имена трех худших по среднему баллу учеников. Если среди остальных есть ученики, набравшие тот же средний балл, что и один из трех худших, то следует вывести и их фамилии и имена.

12. На вход программе подаются прописные английские буквы. Ввод этих символов заканчивается точкой (другие символы, отличные от «.» и букв «А...»Z», во входных данных отсутствуют; в программе на языке Бейсик символы можно вводить по одному в строке, пока не будет введена точка). Требуется написать как можно более эффективную программу на языке Паскаль или Бейсик, которая будет определять, можно ли переставить эти буквы так, чтобы полученное сочетание читалось одинаково как слева направо, так и справа налево. Точка при этом не учитывается.

Если это невозможно, то программа должна вывести на экран слово «NO». А если возможно, то в первой строке следует вывести слово «YES», а во второй — искомое сочетание букв. Если таких сочетаний несколько, то программа должна выводить первое в алфавитном порядке возможное сочетание. Например, пусть на вход подаются следующие символы:

МАМА.

В данном случае программа должна вывести

YES

АММА

13. На вход программе подаются сведения об участниках массовки, пришедших на съемки фильма и получающих зарплату пропорционально отработанному времени. В первой строке задано время начала съемки: через двоеточие два целых числа, соответствующие часам (от 00 до 23 — ровно 2 символа) и минутам (от 00 до 59 — ровно 2 символа). Во второй строке сообщается количество участников съемки N , которое не меньше 10, но не превосходит 1000. Каждая из следующих N строк имеет следующий формат: <Фамилия> <время начала съемки>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <время начала съемки> — через двоеточие два целых числа, соответствующие часам (от 00 до 23 — ровно 2 символа) и минутам (от 00 до 59 — ровно 2 символа). <Фамилия> и <время начала съемки> разделены одним пробелом. Сведения отсортированы в порядке времени прихода. Требуется написать программу на языке Паскаль, которая выведет фамилии участников массовки, которые после 6 часов съемок должны освободиться в хронологическом порядке освобождения.

Пример входных данных:

07:00

3

Иванов 14:00

Петров 15:00

Сидоров 11:30

Результат работы программы для этого примера
Петров
Иванов

14. 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 на языке Паскаль или Бейсик, которая будет печатать буквы, встречающиеся во входной последовательности, в порядке уменьшения частоты их встречаемости.

Каждая буква должна быть распечатана один раз. Точка при этом не учитывается.

Если какие-то буквы встречаются одинаковое число раз, то они выводятся в алфавитном порядке. Например, пусть на вход подаются следующие символы:

batat.

В данном случае программа должна вывести atb.

15. На вход программе подаются сведения о номерах школ учащихся, участвовавших в районной олимпиаде по информатике. В первой строке сообщается количество учащихся N, каждая из следующих N строк имеет формат: <Фамилия> <Инициалы> <номер школы>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Инициалы> — строка, состоящая из 4 символов (буква, точка, буква, точка), <номер школы> — не более чем двузначный номер. <Фамилия> и <Инициалы>, а также <Инициалы> и <номер школы> разделены одним пробелом.

Пример входной строки:

Иванов П.С. 57

Требуется написать как можно более эффективную программу, которая будет выводить на экран информацию, из какой школы было меньше всего участников (таких школ может быть несколько). При этом необхо-

димо вывести информацию только по школам, пославшим хотя бы одного участника. Следует учитывать, что $N \leq 1000$.

- 16.** Вступительные испытания в технический вуз состоят из трех экзаменов: физика (максимальный балл — 9), информатика (максимальный балл — 9), литература (максимальный балл — 5). На вход программе подаются сведения о сдаче этих экзаменов абитуриентами. В первой строке вводится количество абитуриентов N , во второй — количество мест K ($K < N$), на которые эти абитуриенты претендуют. Каждая из следующих N строк имеет следующий формат: `<Фамилия> <оценка1> <оценка2> <оценка3>`, где `<Фамилия>` — строка, состоящая не более чем из 20 символов, оценки — числа от 0 до максимальной оценки по предмету соответственно. (Ноль ставится в случае, если экзамен не сдавался, например после полученной на предыдущем экзамене тройки. Все баллы, большие 3, считаются удовлетворительными.) Пример входных строк:

Иванов 8 9 4

Петров 3 0 0

Требуется написать программу на языке Паскаль или Бейсик, которая определяла бы по имеющимся данным количество абитуриентов, набравших полупроходной балл в данный вуз, или сообщала, что такой балл отсутствует. (Полупроходным называется такой балл, что лишь часть абитуриентов, набравших такой балл и не получивших ни одной неудовлетворительной оценки, попадает в K лучших, которые должны быть зачислены на 1-й курс.) Считается, что абитуриенты, получившие только удовлетворительные оценки, обязательно присутствуют.

ОТВЕТЫ

Информация и ее кодирование

Часть 1

№ задания	Ответ	№ задания	Ответ
1	1	16	2
2	3	17	3
3	3	18	4
4	1	19	2
5	4	20	4
6	3	21	2
7	2	22	4
8	4	23	1
9	3	24	1
10	1	25	2
11	2	26	2
12	1	27	3
13	4	28	1
14	4	29	4
15	3	30	2

№ задания	Ответ	№ задания	Ответ
31	4	41	4
32	2	42	2
33	3	43	1
34	1	44	2
35	2	45	4
36	1	46	1
37	3	47	3
38	3	48	2
39	2	49	3
40	3	50	2

Часть 2

№ задания	Ответ	№ задания	Ответ
51	3	58	360
52	4	59	120
53	6	60	117
54	2	61	13
55	3	62	22,38
56	112	63	38,63
57	340	64	71

№ задания	Ответ	№ задания	Ответ
65	70	73	14
66	5	74	128
67	6	75	64
68	5,10	76	5625
69	6	77	2875
70	5,15	78	1250
71	24	79	4800
72	20	80	45

Алгоритмизация и программирование

Часть 1

№ задания	Ответ	№ задания	Ответ
1	3	9	3
2	1	10	1
3	1	11	3
4	3	12	2
5	2	13	4
6	1	14	2
7	2	15	1
8	2	16	1

№ задания	Ответ	№ задания	Ответ
17	1	29	4
18	2	30	3
19	4	31	4
20	4	32	4
21	3	33	1
22	1	34	3
23	3	35	1
24	4	36	64
25	2	37	4
26	3	38	74
27	3	39	2
28	2	40	4

Часть 2

№ задания	Ответ	№ задания	Ответ
41	21122	46	EDCBAA
42	22212	47	127
43	21121 или 22121	48	85
44	2121	49	1
45	12211	50	21123

Часть 3

51.

Содержание верного ответа и указания к оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)				
Выигрывает первый игрок. Выигрышный ход (7,3) или (4,7). Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры				
Стартовая позиция	1-й ход 1-й игрок	2-й ход 2-й игрок	3-й ход 1-й игрок	Примечания
4,3	6,3	8,3	10,3	При безошибочной игре первого игрока на 3-м ходу игры он побеждает. (Подчеркнуты возможные выигрышные варианты.)
			11,3	
			8,7	
		9,3	11,3	
			<u>12,3</u>	
			9,7	
		6,7	8,7	
			9,7	
			<u>6,11</u>	
	7,3	9,3	те же	
		10,3	<u>12,3</u>	
			<u>13,3</u>	
			<u>10,7</u>	
		7,7	9,7	
			<u>10,7</u>	
			<u>7,11</u>	
	4,7	6,7	те же	
		7,7	те же	

Стартовая позиция	1-й ход 1-й игрок	2-й ход 2-й игрок	3-й ход 1-й игрок	Примечания
4,3	4,7	4,11	6,11	
			7,11	
			4,15	

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
<i>Максимальный балл</i>	3

52.

Содержание верного ответа и указания к оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)
Выигрывает первый игрок. Выигрышный ход (7,12). Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры

Стартовая позиция	1-й ход 1-й игрок	2-й ход 2-й игрок	3-й ход 1-й игрок	Примечание
7,15	5,15	3,15	1,15	При безошибочной игре первый игрок выигрывает на 3-м ходу игры. (Подчеркнуты возможные выигрышные варианты.)
			3,13	
			3,12	
		5,13	3,13	
			5,11	
			5,10	
		5,12	3,12	
			5,10	
			<u>5,9</u>	
	7,13	5,13	те же	
		7,11	5,11	
			7,9	
			<u>7,8</u>	
		7,10	5,10	
			<u>7,8</u>	
			<u>7,7</u>	
	7,12	5,12	<u>те же</u>	
		7,10	<u>те же</u>	
		7,9	<u>5,9</u>	
			<u>7,7</u>	
			<u>7,6</u>	

Указания по оцениванию: см. № 51.

53.

Содержание верного ответа и указания к оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)					
Выигрывает второй игрок. Выигрышный ход (8,6), (8,5) и (5,8). Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры					
Стар- товая позиция	1-й ход 1-й игрок	2-й ход 2-й игрок	3-й ход 1-й игрок	4-й ход 2-й игрок	Пояснение
5,3	8,3	11,3	<u>14,3</u>		Вне зави- симости от первого хода первого иг- рока при без- ошибочной игре на 4-м ходу игры выигрывает второй иг- рок. (Под- черкнуты возможные выигрышные варианты.)
			11,6		
			11,5		
		8,6	11,6	<u>14,6</u>	
			8,9	<u>8,12</u>	
			8,8	<u>8,11</u>	
		8,5	11,5	<u>14,5</u>	
			8,8	<u>8,11</u>	
			8,7	<u>11,7</u>	
	8,6	8,6	те же		
		5,9	8,9		
			5,11		
			<u>5,12</u>		
	5,8	8,8	<u>8,11</u>		
		5,11	<u>5,14</u>		

Стар- товая позиция	1-й ход 1-й игрок	2-й ход 2-й игрок	3-й ход 1-й игрок	4-й ход 2-й игрок	Пояснение
			5,10	<u>5,13</u>	
	5,5	8,5	те же		
		5,8	те же		
		5,9	те же		

Указания по оцениванию: см. № 51.

54.

Содержание верного ответа и указания к оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)				
<p>Выигрывает первый игрок. Выигрышный ход (9,1).</p> <p>Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны пары чисел, разделенные запятой. Эти числа соответствуют количеству камней на каждом этапе игры в первой и второй кучах соответственно</p>				
Стар- товая позиция	1-й ход 1-й игрок	2-й ход 2-й игрок	3-й ход 1-й игрок	Пояснения
3,1	9,1	27,1	<u>81,3</u>	При безошибоч- ной игре первый игрок выигры- вает на 3-м ходу игры. (Подчеркнуты возможные выигрышные варианты.)
			<u>29,1</u>	
			<u>27,3</u>	
			<u>27,3</u>	
		9,3	<u>27,3</u>	
			11,3	
			9,9	
			9,5	

Стар- товая позиция	1-й ход 1-й игрок	2-й ход 2-й игрок	3-й ход 1-й игрок	Пояснения
		11,1	<u>33,1</u>	
			13,1	
			11,3	
			11,3	
		9,3	те же	
	3,3	9,3	те же	
		3,9	те же	
		5,3	15,3	
			7,3	
			5,9	
			5,5	
		3,5	те же	
	5,1	15,1	<u>45,1</u>	
			17,1	
			15,3	
			15,3	
		5,3	те же	
		7,1	21,1	
			9,1	
			7,3	
			7,3	
		5,3	те же	
	3,3	те же	те же	

Указания по оцениванию: см. № 51.

55.

Содержание верного ответа и указания к оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)						
Выигрывает второй игрок. Выигрышный ход (6,8). Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны пары чисел, разделенные запятой. Эти числа соответствуют количеству камней на каждом этапе игры в первой и второй кучах соответственно						
Стар- товая пози- ция	1-й ход 1-й игрок	2-й ход 2-й игрок	3-й ход 1-й игрок	4-й ход 2-й игрок	Пояснение	
3,5	6,5	12,5	<u>24,5</u>		При безошибочной игре второй игрок выигрывает на 4-м ходу игры. (Подчеркнуты возможные выигрышные варианты.)	
		6,10	<u>6,20</u>			
		9,5	<u>18,5</u>			
		6,8	12,8	<u>28,8</u>		
			9,8	<u>18,8</u>		
			6,16	<u>6,32</u>		
			6,12	<u>6,24</u>		
	3,10	6,10				
		<u>3,20</u>				
		6,10				
		3,13				
	6,5	те же				
	3,8	6,8				те же
		3,16	<u>3,32</u>			
		6,8	те же			
		3,12	<u>3,24</u>			

Указания по оцениванию: см. № 51.

Основы логики

Часть 1

№ задания	Ответ	№ задания	Ответ
1	4	9	1
2	1	10	4
3	2	11	1
4	1	12	3
5	3	13	2
6	2	14	1
7	3	15	4
8	1		

Часть 2

№ задания	Ответ	№ задания	Ответ
16	2	24	3
17	2	25	3
18	2	26	3124
19	3	27	1423
20	3	28	CAO
21	7	29	M
22	0101	30	A
23	0110		

Моделирование и компьютерный эксперимент

№ задания	Ответ	№ задания	Ответ
1	2	6	4
2	3	7	3
3	2	8	1
4	4	9	3
5	3	10	2

Программные средства ИКТ

№ задания	Ответ	№ задания	Ответ
1	3	9	2
2	1	10	4
3	4	11	1
4	2	12	4
5	3	13	1
6	3	14	3
7	4	15	2
8	2		

Технология обработки графической информации

№ задания	Ответ	№ задания	Ответ
1	1	3	3
2	2	4	2

№ задания	Ответ	№ задания	Ответ
5	3	13	1
6	2	14	2
7	4	15	3
8	1	16	2
9	3	17	3
10	1	18	4
11	3	19	3
12	2	20	3

Технология обработки данных в электронных таблицах

№ задания	Ответ	№ задания	Ответ
1	4	9	1
2	1	10	3
3	3	11	2
4	1	12	4
5	2	13	1
6	3	14	4
7	4	15	2
8	2		

Технология хранения, поиска и сортировки информации в БД

№ задания	Ответ	№ задания	Ответ
1	2	9	4
2	3	10	2
3	2	11	3
4	1	12	2
5	2	13	3
6	2	14	2
7	3	15	3
8	4		

Телекоммуникационные технологии

№ задания	Ответ	№ задания	Ответ
1	1243	11	ВАБГ
2	4213	12	ВГАБ
3	3412	13	БГВА
4	2314	14	БГАВ
5	1243	15	01101100
6	1243	16	АГЕЖБДВ
7	3214	17	ГАДЖЕБВ
8	3241	18	ЕГБДВЖА
9	2134	19	АВЖГЕБД
10	2143	20	ДЖАГБВЕ

Технологии программирования

1. Требовалось написать программу, которая решает уравнение $ax - b > 0$ 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,x); if a = 0 then if b = 0 then write('любое число') else write('нет решений') else if b = 0 then write('x = 0') else write('x =', -b/a, ' или x =', b/a); end.</pre>	<pre>INPUT a, b, x IF a = 0 THEN IF b = 0 THEN PRINT "любое число" ELSE PRINT «нет реше- ний» ENDIF ELSE IF b = 0 THEN PRINT «x = 0» ELSE PRINT «x =», -b/a, «или x =», b/a ENDIF ENDIF END</pre>	<pre>void main(void) {float a,b,x; scanf("%f%f%f", &a,&b,&x); if (a==0) if (b==0) printf("любое чис- ло"); else printf («нет реше- ний»); else if (b==0) printf(«x = 0»); else printf(«x=%f или x=%f», -b/a,b/a); }</pre>

Последовательно выполните три задания:

- 1) приведите пример таких чисел a , b , x , при которых программа неверно решает поставленную задачу;
- 2) укажите, какая часть программы является лишней;
- 3) 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.)

Ответ:

**Содержание верного ответа и указания по оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)**

Элементы ответа:

1) $a = 1$, $b = -1$, $x = 0$ (значение x может быть не указано. Значения a и b могут быть любыми ненулевыми числами с разными знаками. Также допустим ответ, что программа работает неправильно при любых ненулевых a и b , имеющих разные знаки);

2) лишняя часть:

не нужно вводить x с клавиатуры

верно: `readln(a,b);`

3) возможная доработка:

`readln(a,b);`

`if a = 0 then`

`if b = 0 then write('любое число')`

`else write('нет решений')`

`else`

`if b/a > 0 then`

`write('x=',-b/a, ' или x=',b/a)`

`else`

`if b=0 then write('x=0')`

`else write('нет решений');`

(могут быть и другие способы доработки).

При оценке других вариантов доработки программы нужно проверять, что поставленная цель достигается

Указания к оцениванию	Баллы
Правильно выполнены все 3 пункта задания, при этом в работе (во фрагментах программ) допускается не более одной синтаксической ошибки	3
Правильно выполнены 2 пункта задания. При этом в сданной работе допускается не более двух синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)	2
Правильно выполнен только один пункт задания, при этом если это был п. 3), то в нем допускается не более трех	1

Указания к оцениванию	Баллы
синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)	1
Все пункты задания выполнены неверно	0
<i>Максимальный балл</i>	3

2. Требовалось написать программу, которая находит среднее из трех чисел x , y и z , введенных с клавиатуры. Здесь под средним понимается число, которое меньше другого и больше оставшегося из трех. Числа x , y , z являются натуральными и неравными. Входная информация вводится с клавиатуры, а на выходе должно быть сообщение вида «среднее x ». Программист торопился и написал программу неправильно.

ПРОГРАММА НА БЕЙСИКЕ	ПРОГРАММА НА ПАСКАЛЕ	ПРОГРАММА НА СИ
<pre> INPUT x, y, z IF (x>y) AND (y>z) THEN PRINT "среднее y" IF (y>z) AND (x>z) THEN PRINT "среднее z" IF (y>x) AND (x>z) THEN PRINT "среднее x" END </pre>	<pre> var x, y, z: integer; begin readln(x, y, z); if (x>y) and (y>z) then write('среднее y'); if (y>z) and (z>x) then write('среднее z'); if (y>x) and (x>z) then write('среднее x'); end. </pre>	<pre> void main(void) { int x, y, z; scanf("%d%d%d", &x, &y, &z); if (x>y && y>z) printf("среднее y"); if (y>z && z>x) printf("среднее z"); if (y>x && x>z) printf("среднее x"); } </pre>

Последовательно выполните три задания:

- 1) приведите пример таких чисел a , b , x , при которых программа работает неправильно;
- 2) укажите, какая часть программы является лишней;
- 3) укажите, как нужно доработать программу, чтобы не было случаев ее неправильной работы. (Это можно сде-

лать несколькими способами, поэтому можно указать любой способ доработки исходной программы.)

Ответ:

**Содержание верного ответа и указания по оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)**

Элементы ответа:

1) пример: $x = 3, y = 2, z = 2$

2) возможная доработка:

if ($x > y$) and ($y > z$) then

write('среднее y');

if ($y > z$) and ($z > x$) then

write('среднее z');

if ($y > x$) and ($x > z$) then

write('среднее x');

if ($x > z$) and ($z > y$) then

write('среднее z');

if ($z > x$) and ($x > y$) then

write('среднее x');

if ($z > y$) and ($y > x$) then

write('среднее y');

3) возможная доработка без использования логических операций

AND, OR

if $x < y$ then

if $x > z$ then

write('среднее x')

else

if $x < y$ then

if $x > z$ then

write('среднее x')

else

if $y < z$ then

write('среднее y')

else

write('среднее z')

else

if $y > z$ then

write('среднее y')

else

if $x < z$ then

write('среднее x')

else

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)
<p>write('среднее z')</p> <p>(могут быть и другие способы доработки). При оценке других вариантов доработки программы нужно проверять, что поставленная цель достигается</p>

Указания к оцениванию	Баллы
Правильно выполнены п.1)+п.3) задания (т.к. выполнение п.3 «покрывает» и пункт 2), или правильно выполнены все 3 пункта задания, при этом в работе (во фрагментах программ) допускается не более одной синтаксической ошибки	3
Правильно выполнены 2 пункта задания: 1)+2) или 2)+3) (причем способы доработки в п.2 и п.3 различны). При этом в сданной работе допускается не более двух синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)	2
Правильно выполнен только один пункт задания, при этом если это был п.2) или п.3), то в нем допускается не более двух синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)	1
Все пункты задания выполнены неверно	0
<i>Максимальный балл</i>	3

3. 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,x); if a=0 then write('любое число') else if a>0 then write('x<','-b/a') else write('x>','-b/a'); end.</pre>	<pre>INPUT a, b, x IF a=0 THEN PRINT "любое число» ELSE IF a>0 THEN PRINT «x<», -b/a ELSE PRINT «x>», -b/a END</pre>	<pre>void main(void) { float a,b,x; scanf("%f%f%f", &a,&b,&x); if (a==0) printf("любое число»); else if (a>0) printf(«x<%f», -b/a); else printf(«x>%f», -b/a); }</pre>

Последовательно выполните три задания:

- 1) приведите пример таких чисел a , b , x , при которых программа работает неправильно;
- 2) укажите, какая часть программы является лишней;
- 3) 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.)

Ответ:

**Содержание верного ответа и указания по оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)**

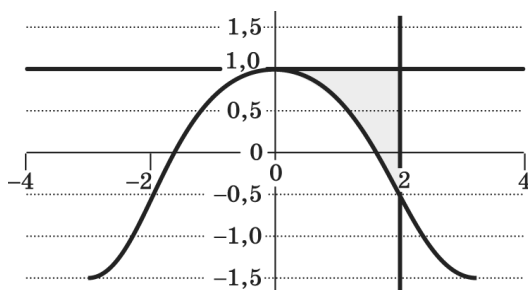
Элементы ответа:

- 1) $a = 1$, $b = -1$, $x = 0$ (значение x может быть не указано. Значения a и b могут быть любыми ненулевыми числами с разными знаками. Также допустим ответ, что программа работает неправильно при любых ненулевых a и b , имеющих разные знаки);
- 2) лишняя часть:
не нужно вводить x с клавиатуры
верно: `readln(a,b);`
- 3) возможная доработка:
`readln(a,b);`
`if a = 0 then`
`if b = 0 then write('любое число')`
`else write('нет решений')`

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)
<pre>else if b/a > 0 then write('x=',-b/a, ' или x=',b/a) else if b=0 then write('x=0') else write('нет решений');</pre> <p>(могут быть и другие способы доработки). При оценке других вариантов доработки программы нужно проверять, что поставленная цель достигается</p>

Указания к оцениванию	Баллы
Правильно выполнены все 3 пункта задания, при этом в работе (во фрагментах программ) допускается не более одной синтаксической ошибки	3
Правильно выполнены 2 пункта задания. При этом в сданной работе допускается не более двух синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)	2
Правильно выполнен только один пункт задания, при этом если это был п.3), то в нем допускается не более трех синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)	1
Все пункты задания выполнены неверно	0
<i>Максимальный балл</i>	3

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



Последовательно выполните следующее:

- 1) приведите пример таких чисел x , y , 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.)

Ответ:

**Содержание верного ответа и указания по оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)**

Элементы ответа:

1) пример: $x=3$, $y=0,5$ (любая пара (x, y) , для которой выполняется: $y > 1$ или $x < 0$ или ($y = \cos x$ и $x > \pi/2$ и $y \leq 1$))

2) возможная доработка (Паскаль):

```
if (y <= 1) and (x >= 0) and (y >= cos(x)) and (x <= 3,14/2) then
  write('принадлежит')
```

```
else
```

```
  write('не принадлежит')
```

(могут быть и другие способы доработки)

Указания по оцениванию

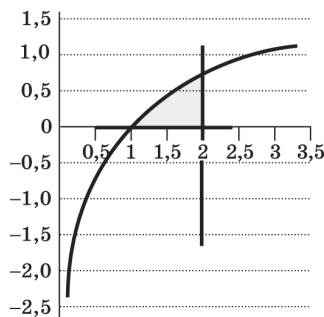
Баллы

Обратите внимание! В задаче требовалось выполнить **три** действия: указать пример входных данных, при которых программа работает неверно, и исправить две ошибки:

Указания по оцениванию	Баллы
<p>1. Неправильное использование условного оператора, в результате чего при невыполнении первого или второго условия программа не выдавала ничего (отсутствуют случаи ELSE).</p> <p>2. Приведенным трем ограничениям удовлетворяют также те точки плоскости, у которых ($y \geq \cos x$ и $x \geq \pi/2$ и $y \leq 1$)</p>	
<p>Правильно выполнены оба пункта задания. Исправлены обе ошибки. Допускается замена числа π на 3,14 или другую константу. В работе (во фрагментах программ) допускается не более одной синтаксической ошибки</p>	3
<p>Правильно выполнены 2 пункта задания из трех (исправлены обе ошибки, но не указан/неправильно указан пример требуемых входных данных, либо правильно указан пример входных данных, программа правильно работает при большем числе случаев, чем исходная, но не при всех). Например, выдает “принадлежит” для точек, у которых ($y \geq \cos x$ и $x > \pi/2$ и $y \leq 1$).</p> <p>Допускается, например, такое решение:</p> <pre> if y<=1 then if x>=0 then if y>=cos(x) then write('принадлежит') else write('не принадлежит') else write('не принадлежит') else write('не принадлежит')</pre> <p>При этом в данной работе допускается не более двух синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)</p>	2
<p>Правильно выполнен только один пункт задания. То есть только приведен пример входных данных, либо он не приведен, но имеется программа, корректно работающая при большем количестве входных данных, чем исходная. При этом если приведена программа, то в ней допускается не</p>	1

Указания по оцениванию	Баллы
более трех синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)	1
Все пункты задания выполнены неверно (пример входных данных не указан или указан неверно, программа не приведена, либо приведенная программа корректно работает в не большем количестве случаев, чем исходная)	0
Максимальный балл	3

5. Требовалось написать программу, которая вводит с клавиатуры координаты точки на плоскости (x, y — действительные числа) и определяет принадлежность точки заштрихованной 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<=1 then if x>=0 then if y>=ln(x) then write('принадле- жит') else</pre>	<pre>INPUT x, y IF y<=1 THEN IF x>=0 THEN IF y>=LN(x) THEN PRINT «принадле- жит» ELSE PRINT «не принад- лежит»</pre>	<pre>void main(void) { float x,y; scanf("%f%f",&x,&y); if (y<=1) if (x>=0) if (y>=ln(x)) printf("принадле- жит"); else</pre>

ПРОГРАММА НА ПАСКАЛЕ	ПРОГРАММА НА БЕЙСИКЕ	ПРОГРАММА НА СИ
write('не принадле- жит') end.	ENDIF ENDIF ENDIF END	printf(«не принадле- жит»); }

Последовательно выполните следующее:

- 1) приведите пример таких чисел x , y , 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.)

Ответ:

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)
<p>Элементы ответа:</p> <p>1) пример: $x=3$, $y=0,5$ (любая пара (x,y), для которой выполняется: $y>1$ или $x<0$ или ($y>=\ln x$ и $x>2$ и $y<=1$))</p> <p>2) возможная доработка (Паскаль):</p> <pre>if (y<=1) and (x>=0) and (y>=ln(x)) and (x<=2) then write('принадлежит') else write('не принадлежит')</pre> <p>(могут быть и другие способы доработки)</p>

Указания по оцениванию	Баллы
<p>Обратите внимание! В задаче требовалось выполнить три действия: указать пример входных данных, при которых программа работает неверно, и исправить две ошибки:</p> <ol style="list-style-type: none"> 1. Неправильное использование условного оператора, в результате чего при невыполнении первого или второго условия программа не выдавала ничего (отсутствуют случаи ELSE). 2. Приведенным трем ограничениям удовлетворяют также те точки плоскости, у которых ($y>=\ln x$ и $x>2$ и $y<=1$) 	

Указания по оцениванию	Баллы
Правильно выполнены оба пункта задания. Исправлены обе ошибки. В работе (во фрагментах программ) допускается не более одной синтаксической ошибки	3
<p>Правильно выполнены 2 пункта задания из трех (исправлены обе ошибки, но не указан/неправильно указан пример требуемых входных данных, либо правильно указан пример входных данных, программа правильно работает при большем числе случаев, чем исходная, но не при всех).</p> <p>Например, выдает “принадлежит” для точек, у которых ($y \geq \ln x$ и $x > 2$ и $y \leq 1$)</p>	2
<p>Допускается, например, такое решение:</p> <pre>if y<=1 then if x>=0 then if y>=ln(x) then write('принадлежит') else write('не принадлежит')</pre> <pre>else write('не принадлежит') else write('не принадлежит')</pre> <p>При этом в сданной работе допускается не более двух синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)</p>	2
Правильно выполнен только один пункт задания. То есть только приведен пример входных данных, либо он не приведен, но имеется программа, корректно работающая при большем количестве входных данных, чем исходная. При этом если приведена программа, то в ней допускается не более трех синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)	1

Указания по оцениванию	Баллы
Все пункты задания выполнены неверно (пример входных данных не указан или указан неверно, программа не приведена, либо приведенная программа корректно работает в не большем количестве случаев, чем исходная)	0
<i>Максимальный балл</i>	3

6. Опишите на русском языке или одном из языков программирования алгоритм получения из заданного целочисленного массива размером 30 элементов количества положительных, отрицательных и нулевых значений.

Ответ:

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)
В цикле от первого элемента до последнего определяем знак элементов исходного массива и суммируем в двух ячейках количество положительных и отрицательных элементов. Количество нулей определяем как разность между общим количеством элементов и суммой положительных и отрицательных элементов. Эти три числа выдаем на печать

Пример правильной и эффективной программы (на основе алгоритма, использующего однократный проход по массиву):

ПРОГРАММА НА БЕЙСИКЕ	ПРОГРАММА НА ПАСКАЛЕ
N=30 DIM X(N) AS INTEGER KP=0 KO=0 FOR I=1 TO N IF X(I)>0 THEN	const N=30; var x:array[1..N] of integer; kp,ko,kn,i:integer; kp:=0;ko:=0; for i:=1 to N do begin

ПРОГРАММА НА БЕЙСИКЕ	ПРОГРАММА НА ПАСКАЛЕ
<pre> KP=KP+1 ELSE KO=KO+1 ENDIF NEXT I KN=N-KP-KO PRINT KP,KO,KN END </pre>	<pre> if x[I]>0 then kp:=kp+1 else ko:=ko+1end; kn:=n-kp-ko; writeln(kp,ko,kn) end. </pre>

Указания по оцениванию	Баллы
Предложен правильный алгоритм, выдающий верное значение. Возможно использование числа 30 вместо константы. Возможно наличие отдельных синтаксических ошибок (пропущенные «;», неверная запись оператора присваивания и т.п.), не искажающих замысла автора программы	2
<p>Имеется не более двух ошибок из числа следующих:</p> <ol style="list-style-type: none"> 1) Не заданы или неверно заданы начальные значения <i>kp</i>, <i>ko</i> 2) Не задано или неверно задано начальное значение индексной переменной в цикле 3) Не указано или неверно указано условие завершения цикла 4) Индексная переменная в цикле не меняется 5) Неверно расставлены операторские скобки 6) Программа не выводит или не возвращает результат 	1
Ошибок, перечисленных выше, больше двух, или алгоритм сформулирован неверно	0
<i>Максимальный балл</i>	2

7. Опишите на русском языке или одном из языков программирования алгоритм подсчета суммы элементов, стоящих на четных местах, в массиве из 30 чисел.

Ответ:

**Содержание верного ответа и указания по оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)**

В цикле от первого элемента до последнего суммируем элементы исходного массива, стоящие на четных местах. При организации цикла на Бейсике используем оператор STEP, а на Паскале используем цикл **while i<=N do**, в котором переменную цикла начинаем с 2 и шаг изменения задаем 2. Полученное число выдаем на печать

Пример правильной и эффективной программы (на основе алгоритма, использующего однократный проход по массиву):

БЕЙСИК	ПАСКАЛЬ
<pre> N = 30 DIM a(N) AS INTEGER S = 0 FOR i = 2 TO N STEP 2 S = S + a(i) NEXT i PRINT S END </pre>	<pre> const N=30; var a:array[1..N] of integer; i, s: integer; begin s:=0; i:=2; while i<=N do begin s:=s+ a[i]; i:=i+2; end; writeln(s); end. </pre>

Указания по оцениванию	Баллы
Предложен правильный алгоритм, выдающий верное значение. Возможно использование числа 30 вместо константы. Возможно наличие отдельных синтаксических ошибок (пропущенные «;», неверная запись оператора присваивания и т.п.), не искажающих замысла автора программы	2
Имеется не более двух ошибок из числа следующих: 1) Не задано или неверно задано начальное значение s 2) Не задано или неверно задано начальное значение индексной переменной в цикле	1

Указания по оцениванию	Баллы
3) Не указано или неверно указано условие завершения цикла 4) Индексная переменная в цикле не меняется 5) Неверно расставлены операторские скобки 6) Программа не выводит или не возвращает результат	1
Ошибок, перечисленных выше, больше двух, или алгоритм сформулирован неверно	0
<i>Максимальный балл</i>	2

8. Опишите на русском языке или одном из языков программирования алгоритм подсчета среднего значения из отрицательных элементов в массиве из 30 чисел.

Ответ:

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)
В цикле от первого элемента до последнего суммируем отрицательные элементы исходного массива, одновременно подсчитываем их количество. По окончании цикла вычисляем среднее значение отрицательных элементов ряда. Полученное число выдаем на печать

Пример правильной и эффективной программы (на основе алгоритма, использующего однократный проход по массиву):

БЕЙСИК	ПАСКАЛЬ
N = 30 DIM a(N) AS INTEGER S = 0 K = 0 FOR i = 1 TO N IF a(i) < 0 THEN	const N=30; var a:array[1..N] of integer; i,k, s: integer; begin s:=0; k:=0; for i:=1 to N do

БЕЙСИК	ПАСКАЛЬ
<pre> S = S + a(i) K=K+1 NEXT i PRINT S/K END </pre>	<pre> if a[i]<0 then begin s:=s+ a[i]; k:=k+1; end; writeln(s/k); end. </pre>

Указания по оцениванию	Баллы
Предложен правильный алгоритм, выдающий верное значение. Возможно использование числа 30 вместо константы. Возможно наличие отдельных синтаксических ошибок (пропущенные «;», неверная запись оператора присваивания и т.п.), не искажающих замысла автора программы	2
<p>Имеется не более двух ошибок из числа следующих:</p> <ul style="list-style-type: none"> 1) Не заданы или неверно заданы начальные значения s,k 2) Не задано или неверно задано начальное значение индексной переменной в цикле 3) Не указано или неверно указано условие завершения цикла 4) Индексная переменная в цикле не меняется 5) Неверно расставлены операторские скобки 6) Программа не выводит или не возвращает результат 	1
Ошибок, перечисленных выше, больше двух, или алгоритм сформулирован неверно	0
<i>Максимальный балл</i>	2

9. Опишите на русском языке или одном из языков программирования алгоритм подсчета количества элементов, попадающих в заданный интервал, в массиве из 30 чисел.

Ответ:

**Содержание верного ответа и указания по оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)**

В цикле от первого элемента до последнего подсчитываем количество элементов исходного массива, попадающих в заданный интервал. Полученное число выдаем на печать

Пример правильной и эффективной программы (на основе алгоритма, использующего однократный проход по массиву):

БЕЙСИК	ПАСКАЛЬ
<pre> N = 30 DIM a(N) AS INTEGER K=0 Nint=0 Kint=0 FOR i = 1 TO N IF (a(i)<=Kint) AND (a(i)>=Nint) THEN K=K+1 NEXT i PRINT K END </pre>	<pre> const N=30; var a:array[1..N] of integer; i,k,Kint,Nint: integer; begin Nint:=0; Kint:=0; k:=0; for i:=1 to N do if (a[i]<=Kint) and (a[i]>=Nint) then k:=k+1; writeln(k); end. </pre>

Указания по оцениванию	Баллы
Предложен правильный алгоритм, выдающий верное значение. Возможно использование числа 30 вместо константы. Возможно наличие отдельных синтаксических ошибок (пропущенные «;», неверная запись оператора присваивания и т.п.), не искажающих замысла автора программы	2
Имеется не более двух ошибок из числа следующих: 1) Не заданы или неверно заданы начальные значения k, Nint, Kint 2) Не задано или неверно задано начальное значение индексной переменной в цикле	1

Указания по оцениванию	Баллы
3) Не указано или неверно указано условие завершения цикла 4) Индексная переменная в цикле не меняется 5) Неверно расставлены операторские скобки 6) Программа не выводит или не возвращает результат	1
Ошибок, перечисленных выше, больше двух, или алгоритм сформулирован неверно	0
<i>Максимальный балл</i>	2

- 10.** Опишите на русском языке или одном из языков программирования алгоритм подсчета количества элементов, попадающих между номером минимального элемента и номером максимального элемента в массиве из 30 чисел.

Ответ:

<p align="center">Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)</p>
<p>В цикле от первого элемента до последнего подсчитываем количество элементов исходного массива, попадающих в заданный интервал. Полученное число выдаем на печать</p>

Пример правильной и эффективной программы (на основе алгоритма, использующего однократный проход по массиву):

БЕЙСИК	ПАСКАЛЬ
<pre> N = 30 DIM a(N) AS INTEGER Max=a(1) Min=a(1) Nmax=1 Nmin=1 FOR i = 1 TO N IF a(i)>Max THEN </pre>	<pre> const N=30; var a:array[1..N] of integer; i, Max, Min, Nmax, Nmin : integer; begin Max:=a[1]; Min:=a[1]; Nmax:=1; Nmin:=1; for i:=1 to N do </pre>

БЕЙСИК	ПАСКАЛЬ
<pre> Max=a(i) Nmax=i ENDIF IF a(i)<Min THEN Min=a(i) Nmin=i ENDIF NEXT i PRINT Nmax-Nmin-1 END </pre>	<pre> begin if a[i]>Max then begin Max:=a[i]; Nmax:=i end; for i:=1 to N do begin if a[i]>Max then begin Max:=a[i]; Nmax:=i end end; writeln(Nmax-Nmin-1); end. </pre>

Указания по оцениванию	Баллы
Предложен правильный алгоритм, выдающий верное значение. Возможно использование числа 30 вместо константы. Возможно наличие отдельных синтаксических ошибок (пропущенные «;», неверная запись оператора присваивания и т.п.), не искажающих замысла автора программы	2
<p>Имеется не более двух ошибок из числа следующих:</p> <p>Не заданы или неверно заданы начальные значения Max, Nmax, Min, Nmin</p> <p>Не задано или неверно задано начальное значение индексной переменной в цикле</p> <p>Не указано или неверно указано условие завершения цикла</p> <p>Индексная переменная в цикле не меняется</p> <p>Неверно расставлены операторские скобки</p> <p>Программа не выводит или не возвращает результат</p>	1
Ошибок, перечисленных выше, больше двух, или алгоритм сформулирован неверно	0
<i>Максимальный балл</i>	2

11. На вход программе подаются сведения о сдаче экзаменов учениками 9-х классов некоторой средней школы. В первой строке сообщается количество учеников N , которое не меньше 10, но не превосходит 100, каждая из следующих N строк имеет следующий формат:

`<Фамилия> <Имя> <оценки>`,

где `<Фамилия>` — строка, состоящая не более чем из 20 символов, `<Имя>` — строка, состоящая не более чем из 15 символов, `<оценки>` — через пробел три целых числа, соответствующие оценкам по пятибалльной системе. `<Фамилия>` и `<Имя>`, а также `<Имя>` и `<оценки>` разделены одним пробелом. Пример входной строки:

Иванов Петр 4 5 3

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например Borland Pascal 7.0), которая будет выводить на экран фамилии и имена трех худших по среднему баллу учеников. Если среди остальных есть ученики, набравшие тот же средний балл, что и один из трех худших, то следует вывести и их фамилии и имена.

Ответ:

Содержание верного ответа и указания к оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)

Программа верно читает входные данные, запоминая фамилии, имена и сумму баллов в массиве записей (или в нескольких массивах), сразу или за дополнительный просмотр подсчитывая три худших по величине суммы баллов (так как количество экзаменов у всех учеников одинаковое, лучший средний балл соответствует лучшей сумме баллов). Затем за дополнительный просмотр этого массива распечатывается информация о тех учениках, которые набрали в сумме баллов не больше третьей по величине суммы. Баллы начисляются только за программу, которая решает задачу хотя бы для частного случая (например, все ученики набрали различный средний балл)

Пример правильной и эффективной программы на языке Паскаль:

```
var p:array[1..100] of record
  name:string;
  sum:integer;
end;
c:char;
i,j,N,s1,s2,s3,m:integer;
begin
  readln(N);
  for i:=1 to N do
  begin
    p[i].name:='';
    repeat
      read(c);
      p[i].name:=p[i].name+c
    until c=' '; {считана фамилия}
    repeat
      read(c);
      p[i].name:=p[i].name+c
    until c=' '; {считано имя}
    p[i].sum:=0;
    for j:=1 to 3 do
    begin
      read(m);
      p[i].sum:=p[i].sum+m
    end; {подсчитана сумма баллов}
    readln;
  end;
  s1:=20; s2:=20; s3:=20;
  for i:=1 to N do
  begin
    if p[i].sum<s1 then
    begin
      s3:=s2; s2:=s1;
      s1:=p[i].sum
    end else
    if p[i].sum<s2 then
    begin
      s3:=s2; s2:=p[i].sum
    end else
```

```

if p[i].sum<s3 then s3:=p[i].sum;
end;
for i:=1 to N do
if p[i].sum<=s3 then writeln(p[i].name);
end.

```

Указания к оцениванию	Баллы
<p>Программа работает верно, т.е. корректно выделяет из входных данных оценки, ищет три худшие суммы баллов и распечатывает учеников, набравших эти суммы. Допускается наличие в тексте программы одной синтаксической ошибки</p>	4
<p>Программа работает в целом верно, но содержит по крайней мере две из следующих неточностей (нерацональностей): сохраняются не суммы баллов (средние баллы), а сами баллы, и суммы перевычисляются несколько раз заново; явно вычисляются средние баллы, что приводит к сравнению вещественных чисел; при нахождении трех минимальных значений элементы массива переставляются местами; при печати сравнения производятся с каждым из трех минимальных элементов. Допускается наличие от одной до трех синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных</p>	3
<p>Программа работает в целом верно, но выводит только трех худших учеников, даже если кто-то еще сдал экзамены так же. Возможно, в реализации алгоритма содержится 1–2 ошибки (используется знак “<” вместо “>”, “or” вместо “and” и т.п.). Возможно, некорректно организовано считывание входных данных. Допускается наличие до пяти синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных</p>	2

Указания к оцениванию	Баллы
Программа неверно работает при некоторых входных данных и, возможно, содержит ошибку в алгоритме поиска трех минимальных элементов. Допускается до 4 различных ошибок в ходе решения задачи, в том числе описанных в критериях присвоения двух баллов. Допускается наличие от одной до семи синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных	1
Задание выполнено неверно	0
<i>Максимальный балл</i>	4

12. На вход программе подаются прописные английские буквы. Ввод этих символов заканчивается точкой (другие символы, отличные от «.» и букв «А»...«Z», во входных данных отсутствуют; в программе на языке Бейсик символы можно вводить по одному в строке, пока не будет введена точка). Требуется написать как можно более эффективную программу на языке Паскаль или Бейсик, которая будет определять, можно ли переставить эти буквы так, чтобы полученное сочетание читалось одинаково как слева направо, так и справа налево. Точка при этом не учитывается.

Если это невозможно, то программа должна вывести на экран слово «NO». А если возможно, то в первой строке следует вывести слово «YES», а во второй — искомое сочетание букв. Если таких сочетаний несколько, то программа должна выводить первое в алфавитном порядке возможное сочетание. Например, пусть на вход подаются следующие символы:

МАМА.

В данном случае программа должна вывести

YES

АММА

Ответ:

**Содержание верного ответа и указания по оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)**

Программа читает все входные символы до точки один раз, подсчитывая в массиве, хранящем 26 целых чисел, количество каждой из букв. Сами входные символы при этом не запоминаются. Затем проверяется, сколько в этом массиве нечетных элементов. Если больше одного, то задача решения не имеет. При наличии решения сначала печатается половина имеющихся букв А (если таковые имеются и их количество четно), затем В и т.д. до Z, потом печатается буква, которая встречается во входных данных нечетное число раз (столько же раз она и печатается), а затем — оставшаяся половина букв Z (если таковые имеются и их количество четно), Y, и т.д. до А.

Баллы начисляются только за программу, которая решает задачу хотя бы для одного частного случая (например, для строк, состоящих не более чем из 255 символов) или которая только умеет определять, имеет ли задача решение

Пример правильной и эффективной программы на языке Паскаль:

```
var a:array['A'..'Z'] of integer;
c, c_odd: char;
i, k: integer;
begin
  for c:='A' to 'Z' do a[c]:=0;
  read(c);
  while c<> '.' do
    begin
      a[c] := a[c] + 1;
      read(c);
    end;
  k := 0; {количество букв, встречающихся нечетное число раз}
  for c := 'A' to 'Z' do
    if a[c] mod 2 = 1 then
      begin
        k := k + 1;
        c_odd := c
      end;
    if k > 1 then writeln('NO') else
```

```

begin
writeln('YES');
for c := 'A' to 'Z' do
if a[c] mod 2 = 0 then
for i := 1 to a[c] div 2 do
write(c);
if k = 1 then
for i := 1 to a[c_odd] do
write(c_odd);
for c := 'Z' downto 'A' do
if a[c] mod 2 = 0 then
for i := 1 to a[c] div 2 do
write(c);
end
end.

```

Указания по оцениванию	Баллы
<p>Программа работает верно, т.е. определяет, имеет ли задача решение для любых входных данных произвольного размера, и строит первое по алфавиту решение, не сохраняя входные данные в строке или массиве символов. Программа просматривает входные данные один раз, в тексте программы не анализируется каждая английская буква в отдельности. Допускается наличие в тексте программы одной пунктуационной ошибки</p>	4
<p>Программа работает верно, но входные данные запоминаются в массиве символов или строке или входные данные считываются несколько раз. Возможно, каждая буква обрабатывается явным образом (26 операторов IF с использованием многоточия при записи программы или оператор CASE, содержащий 26 вариантов). Возможно, сохраненные входные данные сортируются одним из стандартных алгоритмов сортировки путем перестановки входных символов или ответ формируется путем перестановки входных символов. Допускается наличие от одной до трех синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных</p>	3

Указания по оцениванию	Баллы
<p>Программа работает в целом верно, но, возможно, выводит значение не первого по алфавиту искомого слова. Возможно, в реализации алгоритма содержатся 1–2 ошибки (используется знак “/” вместо “\”, “div” вместо “mod” и т.п.). Возможно, некорректно организовано считывание входных данных. Допускается наличие от одной до пяти синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных</p>	2
<p>Программа, возможно, неверно работает при некоторых входных данных, например при наличии буквы, которая встречается нечетное число раз. Возможно, выводит только “NO” или “YES” и не выводит искомое слово или выводит его неверно. Допускается наличие от одной до семи синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных</p>	1
Задание выполнено неверно	0
<i>Максимальный балл</i>	4

13. На вход программе подаются сведения об участниках массовки, пришедших на съемки фильма и получающих зарплату пропорционально отработанному времени. В первой строке задано текущее время начала съемки: через двоеточие два целых числа, соответствующие часам (от 00 до 23 — ровно 2 символа) и минутам (от 00 до 59 — ровно 2 символа). Во второй строке сообщается количество участников съемки N , которое не меньше 10,

но не превосходит 1000. Каждая из следующих N строк имеет следующий формат: <Фамилия> <время начала съемки>, где <Фамилия> – строка, состоящая не более, чем из 20 символов, <время начала съемки> — через двоеточие два целых числа, соответствующие часам (от 00 до 23 — ровно 2 символа) и минутам (от 00 до 59 — ровно 2 символа). <Фамилия> и <время начала съемки> разделены одним пробелом. Сведения отсортированы в порядке времени начала съемки. Требуется написать программу на языке Паскаль или Бейсик, которая выведет фамилии участников массовки, которые после 6 часов съемок должны освободиться в хронологическом порядке.

Пример входных данных:

7:00

3

Иванов 14:00

Петров 15:00

Сидоров 11:30

Результат работы программы для этого примера

Петров

Иванов

Ответ:

**Содержание верного ответа и указания по оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)**

Программа верно читает входные данные, сразу запоминая в массиве только фамилии и времена окончания работы тех участников массовки, которые должны освободиться после 6 часов работы. Время при считывании удобно перевести в минуты и в этом же виде хранить и сравнивать. Затем полученный массив времен сортируется по неубыванию любым алгоритмом сортировки, параллельно переставляются и элементы массива с фамилиями (возможно использование одного массива записей, состоящих из двух полей). Печатаются элементы массива фамилий в полученном в результате сортировки порядке

Пример правильной и эффективной программы на языке Паскаль:

```

type pp=record
name:string[20];
time:integer;
end;
var
p:array[1..1000]of pp;
q:pp;
c,c1:char;
i,j,N,time1:integer;
begin
read(c,c1); {читаны часы текущего времени}
time1:=60*((ord(c)-ord('0'))*10+ ord(c1)-ord('0'));
readln(c,c,c1); {пропущено двоеточие, и считаны минуты}
time1:=time1+(ord(c)-ord('0'))*10+ord(c1)-ord('0');
readln(N);
j:=1;
for i:=1 to N do
begin
p[j].name:='';
repeat
read(c);
p[j].name:=p[j].name+c
until c=' '; {считана фамилия}
read(c,c1); {читаны часы начала съемки}
p[j].time:=60*((ord(c)-ord('0'))*10+ ord(c1)-ord('0'));
readln(c,c,c1); {пропущено двоеточие, и считаны минуты}
p[j].time:=p[j].time+(ord(c)-ord('0'))*10+ord(c1)-
ord('0');
if (p[j].time>=time1)and(p[j].time<=time1+120)then
j:=j+1; {данные занесены в массив}
end;
N:=j-1;
for i:=1 to N-1 do {сортируем данные}
for j:=1 to N-i do
if p[j].time>p[j+1].time then
begin
q:=p[j];
p[j]:=p[j+1];
p[j+1]:=q;
end;
for i:=1 to N do
writeln(p[i].name)
end.

```

Указания по оцениванию	Баллы
<p>Программа работает верно и эффективно, т.е. корректно выделяет из входных данных время, запоминает фамилии только тех участников съемки, которые должны освободиться первыми через 6 часов съемок. Фамилии этих участников верно сортируются согласно отработанному времени, а затем печатаются. Допускается наличие в тексте программы одной пунктуационной ошибки</p>	4
<p>Программа работает в целом верно, но содержит по крайней мере две из следующих нерациональностей: сохраняются фамилии и времена для всех участников массовой, время не переводится в минуты, сортируются все фамилии, а при печати анализируется, какие из них допустимые. Допускается наличие от одной до трех синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных</p>	3
<p>Программа работает в целом верно, но не всегда верно определяет допустимость времени или некорректно работает в случае отсутствия допустимых времен. Возможно, в реализации алгоритма содержатся 1—2 ошибки (используется знак “<” вместо “>”, “or” вместо “and” и т.п.). Возможно, некорректно организовано считывание входных данных. Допускается наличие до пяти синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных</p>	2
<p>Программа неверно работает при некоторых входных данных и, возможно, содержит ошибку в сортировке, или времена сортируются верно, а соответствующие им фамилии — нет. Допускается до 4 различных ошибок в ходе решения задачи, в том числе описанных в критериях присвоения двух баллов. Допускается наличие от одной до семи синтаксических ошибок: пропущен или</p>	1

Указания по оцениванию	Баллы
неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных	1
Задание выполнено неверно	0
Максимальный балл	4

14. 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 на языке Паскаль или Бейсик, которая будет печатать буквы, встречающиеся во входной последовательности, в порядке уменьшения частоты их встречаемости.

Каждая буква должна быть распечатана один раз. Точка при этом не учитывается.

Если какие-то буквы встречаются одинаковое число раз, то они выводятся в алфавитном порядке. Например, пусть на вход подаются следующие символы:

batat.

В данном случае программа должна вывести atb.

Ответ:

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)
Программа читает все входные символы до точки один раз, подсчитывая в массиве, хранящем 26 целых чисел, количество каждой из букв. Сами входные символы при этом не запоминаются. В дополнительный массив, состоящий из 26 символов, заносятся буквы от

**Содержание верного ответа и указания по оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)**

“а” до “z”. Затем элементы первого массива сортируются по невозрастанию любым алгоритмом сортировки, параллельно переставляются и элементы второго массива (возможно использование одного массива записей, состоящих из двух полей). При этом элементы с равным числом вхождений символов местами не меняются. Печатаются элементы второго из отсортированных массивов, количество которых больше 0. Баллы начисляются только за программу, которая решает задачу хотя бы для одного частного случая (например, для строк, состоящих не более чем из 255 символов).

Пример правильной и эффективной программы на языке Паскаль:

```
var a:array[0..25] of integer;
m:array[0..25] of 'a'..'z';
c: char;
i, j, k: integer;
begin
  for i:=0 to 25 do
    begin
      a[i]:=0;
      m[i]:=chr(ord('a')+i)
    end;
  read(c);
  while c<>'.' do
    begin
      a[ord(c)-ord('a')] := a[ord(c)-ord('a')] + 1;
      read(c);
    end;
  for i:=1 to 25 do
    for j := 0 to 24 do
      if a[j] < a[j+1] then
        begin
          k:=a[j]; c:=m[j];
          a[j]:=a[j+1]; m[j]:=m[j+1];
          a[j+1]:=k; m[j+1]:=c
        end;
      i:=0;
    while (i<26)and (a[i]>0) do
```

```

begin
write(m[i]);
i:=i+1
end;
writeln
end.

```

Указания по оцениванию	Баллы
<p>Программа работает верно для любых входных данных произвольного размера и строит решение, не сохраняя входные данные в строке или массиве символов. Программа просматривает входные данные один раз, в тексте программы не анализируется каждая английская буква в отдельности. Допускается наличие в тексте программы одной пунктуационной ошибки</p>	4
<p>Программа работает верно, но входные данные запоминаются в массиве символов или строке, или входные данные считываются несколько раз. Возможно, каждая буква обрабатывается явным образом (26 операторов IF с использованием многоточия при записи программы или оператор CASE, содержащий 26 вариантов). Возможно, после сохранения входных данных для каждой буквы от “a” до “z” они просматриваются заново и подсчитывается количество вхождений соответствующей буквы. В программе присутствуют вложенные циклы (один по входным данным, второй — по буквам алфавита или их номерам, он может быть заменен оператором CASE или 26 операторами IF). Допускается наличие от одной до трех синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных</p>	3
<p>Программа работает в целом верно, эффективно или нет, но в реализации алгоритма содержатся 1–2 ошибки (используется знак “>” вместо “<”, выход за границу массива и т.п.), в том числе в алгоритме сортировки или при распечатке результата. Возможно, буквы, встречающиеся с одинаковой частотой, выводятся не по алфавиту. Возможно, некорректно организовано считывание входных данных.</p>	

Указания по оцениванию	Баллы
Допускается наличие от одной до пяти синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных	2
Программа, возможно, неверно работает при некоторых входных данных. Возможно, программа выводит в том числе и буквы, которые во входных данных не встречаются, или содержит другие ошибки в выводе ответа. Допускается до 4 различных ошибок в реализации алгоритма, в том числе описанных в критериях присвоения двух баллов. Допускается наличие от одной до семи синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, не допустимая для соответствующего типа данных	1
Задание выполнено неверно	0
<i>Максимальный балл</i>	4

15. На вход программе подаются сведения о номерах школ учащихся, участвовавших в районной олимпиаде по информатике. В первой строке сообщается количество учащихся N , каждая из следующих N строк имеет формат: <Фамилия> <Инициалы> <номер школы>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Инициалы> — строка, состоящая из 4 символов (буква, точка, буква, точка), <номер школы> — не более чем двузначный номер. <Фамилия> и <Инициалы>, а также <Инициалы> и <номер школы> разделены одним пробелом. Пример входной строки:

Иванов П.С. 57

Требуется написать как можно более эффективную программу, которая будет выводить на экран информацию, из какой школы было меньше всего участников (таких

школ может быть несколько). При этом необходимо вывести информацию только по школам, пославшим хотя бы одного участника.

Следует учитывать, что $N \leq 1000$.

Ответ:

Содержание верного ответа и указания к оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)

Программа верно читает входные данные, запоминая фамилии, имена и номер школы в массиве записей (или в нескольких массивах), сразу или за дополнительный просмотр запоминая номера школ, приславших хотя бы одного участника. Баллы начисляются только за программу, которая решает задачу хотя бы для частного случая (например, из всех школ участвовали по одному участнику)

Пример правильной и эффективной программы на языке Паскаль:

```
var p:array[1..1000] of record
  name:string;
  sum:integer;
end;
c:char;
i,j,N,s1,m:integer;
begin
  readln(N);
  for i:=1 to N do
    begin
      p[i].name:='';
      repeat
        read(c);
        p[i].name:=p[i].name+c
      until c=' '; {считана фамилия}
      repeat
        read(c);
        p[i].name:=p[i].name+c
      until c=' '; {считано имя}
      p[i].sum:=0;
      for j:=1 to 3 do
```

```

begin
read(ns); {считан номер школы}
p[i].ns:= ns
end;
readln;
end;
s1:=1000;
for i:=1 to N do
begin
if p[i].sum<s1 then
begin
s3:=s2; s2:=s1;
s1:=p[i].sum
end else
if p[i].sum<s2 then
begin
s3:=s2; s2:=p[i].sum
end else
if p[i].sum<s3 then s3:=p[i].sum;
end;
for i:=1 to N do
if p[i].sum<=s3 then writeln(p[i].name);
end.

```

Указания к оцениванию	Баллы
Программа работает верно, т.е. корректно выделяет из входных данных оценки, ищет три худшие суммы баллов и распечатывает учеников, набравших эти суммы. Допускается наличие в тексте программы одной синтаксической ошибки	4
Программа работает в целом верно, но содержит по крайней мере две из следующих неточностей (нерациональностей): сохраняются не суммы баллов (средние баллы), а сами баллы, и суммы перевычисляются несколько раз заново; явно вычисляются средние баллы, что приводит к сравнению вещественных чисел; при нахождении трех минимальных значений элементы массива переставляются местами; при печати сравнения производятся с каждым из трех минимальных элементов. Допускается наличие от одной до трех синтаксических ошибок:	3

Указания к оцениванию	Баллы
пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, недопустимая для соответствующего типа данных	3
Программа работает в целом верно, но выводит только трех худших учеников, даже если кто-то еще сдал экзамены так же. Возможно, в реализации алгоритма содержатся 1–2 ошибки (используется знак “<” вместо “>”, “or” вместо “and” и т.п.). Возможно, некорректно организовано считывание входных данных. Допускается наличие до пяти синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, недопустимая для соответствующего типа данных	2
Программа неверно работает при некоторых входных данных и, возможно, содержит ошибку в алгоритме поиска трех минимальных элементов. Допускается до 4 различных ошибок в ходе решения задачи, в том числе описанных в критериях присвоения двух баллов. Допускается наличие от одной до семи синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, недопустимая для соответствующего типа данных	1
Задание выполнено неверно	0
<i>Максимальный балл</i>	4

16. Вступительные испытания в технический вуз состоят из трех экзаменов: физика (максимальный балл — 9), информатика (максимальный балл — 9), литература (максимальный балл — 5). На вход программе подаются сведения о сдаче этих экзаменов абитуриентами. В первой строке вводится количество абитуриентов N, во второй — количество

мест K ($K < N$), на которые эти абитуриенты претендуют. Каждая из следующих N строк имеет следующий формат: $\langle \text{Фамилия} \rangle \langle \text{оценка1} \rangle \langle \text{оценка2} \rangle \langle \text{оценка3} \rangle$, где $\langle \text{Фамилия} \rangle$ — строка, состоящая не более чем из 20 символов, оценки — числа от 0 до максимальной оценки по предмету соответственно. (Ноль ставится в случае, если экзамен не сдавался, например после полученной на предыдущем экзамене тройки. Все баллы, большие 3, считаются удовлетворительными). Пример входных строк:

Иванов 8 9 4

Петров 3 0 0

Требуется написать программу на языке Паскаль или Бейсик, которая определяла бы по имеющимся данным количество абитуриентов, набравших полупроходной балл в данный вуз, или сообщала, что такой балл отсутствует. (Полупроходным называется такой балл, что лишь часть абитуриентов, набравших такой балл и не получивших ни одной неудовлетворительной оценки, попадает в K лучших, которые должны быть зачислены на 1-й курс.) Считается, что абитуриенты, получившие только удовлетворительные оценки, обязательно присутствуют.

16

**Содержание верного ответа и указания по оцениванию
(допускаются иные формулировки ответа,
не искажающие его смысла)**

Программа верно читает входные данные, не запоминая их все, а сразу подсчитывая в массиве, хранящем 24 целых числа, количество абитуриентов, набравших тот или иной балл (от 0 до 23). Если при этом абитуриент получил хотя бы одну двойку, то удобно считать, что его общий балл равен 0. Затем вычисляется сумма элементов этого массива начиная с 23-го, до тех пор пока она не превосходит K . Индекс первого элемента массива, который не войдет в эту сумму и будет искомым полупроходным баллом. Если проходной балл набрали ровно K абитуриентов, то программа сообщает, что полупроходной балл отсутствует. Баллы начисляются только за программу, которая решает задачу хотя бы для частного случая (например, проходной балл набрали строго меньше K абитуриентов)

Пример правильной и эффективной программы на языке Паскаль:

```

var m:array[0..23] of integer;
    c:char;
    i, K, N, S, m1, m2, m3:integer;
begin
  readln(N); readln(K);
  for i:=0 to 23 do m[i]:=0;
  for i:=1 to N do
    begin
      repeat
        read(c)
      until c=' '; {считана фамилия абитуриента}
      readln(m1, m2, m3);
      if (m1<4)or(m2<4)or(m3<4) then s:=0
      else s:=m1+m2+m3;
      m[s]:=m[s]+1 {учитываем абитуриента в элементе массива, соответ-
        ствующем его баллам}
    end;
    s:=m[23]; i:=23;
    while s+m[i-1]<=K and
      (i>9) {9 - минимально возможный балл} do
      begin
        i:=i-1;
        s:=s+m[i]
      end;
      if (s<K)and(i>9) then

        writeln('полупроходной балл набрали', m[i-1], ' человек')
        else writeln('полупроходной балл отсутствует');
      readln
    end.

```

Указания по оцениванию	Баллы
Программа работает верно, т.е. корректно выделяет из входных данных оценки абитуриентов, верно учитывает результаты абитуриентов, получивших двойки, не содержит вложенных циклов (от 1 до N и от 0 до 23). Допускается наличие в тексте программы одной пунктуационной ошибки	4

Указания по оцениванию	Баллы
<p>Программа работает верно, но содержит вложенные циклы (от 0 до 23 и от 1 до N) или несколько операторов IF (по количеству возможных баллов у абитуриента) или оператор CASE, обрабатывающий различные варианты количества баллов абитуриента. Возможно, сохраняет все входные данные в массиве абитуриентов. Допускается наличие от одной до трех различных синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, недопустимая для соответствующего типа данных</p>	3
<p>Программа не учитывает случай, когда ровно K абитуриентов набрали проходной балл или что количество абитуриентов, получивших удовлетворительные оценки, может оказаться меньше K. Возможно, в реализации алгоритма содержатся 1–2 ошибки (используется знак “<” вместо “<=”, “or” вместо “and”, выражение на 1 отличается от верного и т.п.). Допускается наличие от одной до пяти различных синтаксических ошибок</p>	2
<p>Программа неверно работает при некоторых входных данных, возможно, содержит ошибку при учете баллов абитуриентов, получивших неудовлетворительные оценки или в выделении оценок из строки входных данных, или в логике определения полупроходного балла. Допускается наличие от одной до семи различных синтаксических ошибок</p>	1
<p>Задание выполнено неверно</p>	0
<p><i>Максимальный балл</i></p>	4

Издание для дополнительного образования

Для старшего школьного возраста

ЕГЭ. СБОРНИК ЗАДАНИЙ

**Зорин Михаил Вячеславович
Зорина Елена Михайловна**

ЕГЭ 2012

Информатика

Сборник заданий

Ответственный редактор *А. Жилинская*

Ведущий редактор *Т. Судакова*

Художественный редактор *Е. Брынчик*

Компьютерная верстка *А. Григорьев*

ООО «Издательство «Эксмо»

127299, Москва, ул. Клары Цеткин, д. 18/5. Тел. 411-68-86, 956-39-21.

Home page: **www.eksmo.ru** E-mail: **info@eksmo.ru**

Подписано в печать 19.07.2011.

Формат 60×90^{1/16}. Гарнитура «Школьная». Печать офсетная.

Бум. тип. Усл. печ. л. 13,0.

Тираж экз. Заказ №

ISBN 978-5-699-50665-1



9 785699 506651 >

Оптовая торговля книгами «Эксмо»:

ООО «ТД «Эксмо», 142702, Московская обл., Ленинский р-н, г. Видное,
Белокаменное ш., д. 1, многоканальный тел. 411-50-74.

E-mail: reception@eksmo-sale.ru

По вопросам приобретения книг «Эксмо»

зарубежными оптовыми покупателями

обращаться в отдел зарубежных продаж ТД «Эксмо»

E-mail: international@eksmo-sale.ru

International Sales: International wholesale customers should contact

Foreign Sales Department of Trading House «Eksmo» for their orders.

international@eksmo-sale.ru

По вопросам заказа книг корпоративным клиентам,

в том числе в специальном оформлении,

обращаться по тел. 411-68-59, доб. 2115, 2117, 2118, 411-68-99, доб. 2762, 1234.

E-mail: vipzakaz@eksmo.ru

Оптовая торговля бумажно-беловыми

и канцелярскими товарами для школы и офиса «Канц-Эксмо»:

Компания «Канц-Эксмо»: 142700, Московская обл., Ленинский р-н,
г. Видное-2, Белокаменное ш., д. 1, а/я 5.

Тел./факс +7 (495) 745-28-87 (многоканальный).

e-mail: kanc@eksmo-sale.ru, сайт: www.kanc-eksmo.ru

Полный ассортимент книг издательства «Эксмо» для оптовых покупателей:

В Санкт-Петербурге: ООО СЗКО, пр-т Обуховской Обороны, д. 84Е.

Тел. (812) 365-46-03/04.

В Нижнем Новгороде: ООО ТД «Эксмо НН», ул. Маршала Воронова, д. 3.

Тел. (8312) 72-36-70.

В Казани: Филиал ООО «РДЦ-Самара», ул. Фрезерная, д. 5.

Тел. (843) 570-40-45/46.

В Самаре: ООО «РДЦ-Самара», пр-т Кирова, д. 75/1, литера «Е».

Тел. (846) 269-66-70.

В Ростове-на-Дону: ООО «РДЦ-Ростов», пр. Стачки, 243А.

Тел. (863) 220-19-34.

В Екатеринбурге: ООО «РДЦ-Екатеринбург», ул. Прибалтийская, д. 24а.

Тел. +7 (343) 272-72-01/02/03/04/05/06/07/08.

В Новосибирске: ООО «РДЦ-Новосибирск», Комбинатский пер., д. 3.

Тел. +7 (383) 289-91-42. E-mail: eksmo-nsk@yandex.ru

В Киеве: ООО «РДЦ Эксмо-Украина», Московский пр-т, д. 9.

Тел./факс (044) 495-79-80/81.

Во Львове: ТП ООО «Эксмо-Запад», ул. Бузкова, д. 2.

Тел./факс: (032) 245-00-19.

В Симферополе: ООО «Эксмо-Крым», ул. Киевская, д. 153.

Тел./факс (0652) 22-90-03, 54-32-99.

В Казахстане: ТОО «РДЦ-Алматы», ул. Домбровского, д. 3а.

Тел./факс (727) 251-59-90/91. rdc-almaty@mail.ru

Полный ассортимент продукции издательства «Эксмо»

можно приобрести в магазинах «Новый книжный» и «Читай-город».

Телефон единой справочной: 8 (800) 444-8-444.

Звонок по России бесплатный.

В Санкт-Петербурге в сети магазинов «Буквоед»:

«Парк культуры и чтения», Невский пр-т, д. 46. Тел. (812) 601-0-601

www.bookvoed.ru

ДЛЯ ЗАМЕТОК

ПОДГОТОВКА К ЕГЭ ВЫСШИЙ УРОВЕНЬ КАЧЕСТВА



СБОРНИК ЗАДАНИЙ ИНФОРМАТИКА

Выпускникам школ, абитуриентам и педагогам для повышения качества подготовки к ЕГЭ издательство «Эксмо» предлагает учебно-методические комплекты по подготовке к единому государственному экзамену по всем предметам: русскому языку, литературе, математике, истории, обществознанию, биологии, географии, химии, физике, информатике, иностранным языкам.

В каждый предметный комплект входят:

- Тренировочные задания
- Тематические тренировочные задания
- Сборник заданий
- Универсальный справочник
- 100 дней до ЕГЭ
- Сдаем без проблем

Настоящее издание содержит:

- ✓ более 300 заданий частей А, В и С;
- ✓ методические рекомендации;
- ✓ ответы и комментарии.

ISBN 978-5-699-50665-1



9 785699 506651 >

