

**Единый государственный экзамен  
по ИНФОРМАТИКЕ и ИКТ**

**Инструкция по выполнению работы**

Экзаменационная работа состоит из двух частей, включающих в себя 27 заданий. Часть 1 содержит 23 задания с кратким ответом, часть 2 содержит 4 задания с развёрнутым ответом.

На выполнение экзаменационной работы по информатике и ИКТ отводится 3 часа 55 минут (235 минут).

Ответы к заданиям 1–23 записываются в виде числа, последовательности букв или цифр. Ответы запишите в поле ответа в тексте работы, а затем перенесите в бланк ответов № 1.

Ответ: 23.

1 2 3

Задания 24–27 требуют развёрнутого ответа. В бланке ответов № 2 укажите номер задания и запишите его полное решение.

Все бланки ЕГЭ заполняются яркими чёрными чернилами. Допускается использование гелевой, капиллярной или перьевой ручек.

При выполнении заданий можно пользоваться черновиком. Записи в черновике не учитываются при оценивании работы.

Баллы, полученные Вами за выполненные задания, суммируются. Постарайтесь выполнить как можно больше заданий и набрать наибольшее количество баллов.

***Желаем успеха!***

В экзаменационных заданиях используются следующие соглашения.

1. Обозначения для логических связок (операций):
  - a) отрицание (инверсия, логическое НЕ) обозначается  $\neg$  (например,  $\neg A$ );
  - b) конъюнкция (логическое умножение, логическое И) обозначается  $\wedge$  (например,  $A \wedge B$ ) либо  $\&$  (например,  $A \& B$ );
  - c) дизъюнкция (логическое сложение, логическое ИЛИ) обозначается  $\vee$  (например,  $A \vee B$ ) либо  $|$  (например,  $A | B$ );
  - d) следование (импликация) обозначается  $\rightarrow$  (например,  $A \rightarrow B$ );
  - e) тождество обозначается  $\equiv$  (например,  $A \equiv B$ ). Выражение  $A \equiv B$  истинно тогда и только тогда, когда значения  $A$  и  $B$  совпадают (либо они оба истинны, либо они оба ложны);
  - f) символ 1 используется для обозначения истины (истинного высказывания); символ 0 – для обозначения лжи (ложного высказывания).

2. Два логических выражения, содержащих переменные, называются равносильными (эквивалентными), если значения этих выражений совпадают при любых значениях переменных. Так, выражения  $A \rightarrow B$  и  $(\neg A) \vee B$  равносильны, а  $A \vee B$  и  $A \wedge B$  неравносильны (значения выражений разные, например, при  $A = 1, B = 0$ ).

3. Приоритеты логических операций: инверсия (отрицание), конъюнкция (логическое умножение), дизъюнкция (логическое сложение), импликация (следование), тождество. Таким образом,  $\neg A \wedge B \vee C \wedge D$  означает то же, что и  $((\neg A) \wedge B) \vee (C \wedge D)$ .

Возможна запись  $A \wedge B \wedge C$  вместо  $(A \wedge B) \wedge C$ . То же относится и к дизъюнкции: возможна запись  $A \vee B \vee C$  вместо  $(A \vee B) \vee C$ .

4. Обозначения Мбайт и Кбайт используются в традиционном для информатики смысле – как обозначения единиц измерения, чьё соотношение к единице «байт» выражается степенью двойки.

КИМ

Бланк



**Часть 1**

*Ответами к заданиям 1–23 являются число, последовательность букв или цифр, которые следует записать в БЛАНК ОТВЕТОВ № 1 справа от номера соответствующего задания, начиная с первой клеточки, без пробелов, запятых и других дополнительных символов. Каждый символ пишете в отдельной клеточке в соответствии с приведёнными в бланке образцами.*

**1** Сколько существует натуральных чисел, для которых выполнено неравенство  $70_8 < x < 70_{16}$

Ответ: \_\_\_\_\_.

**2** Логическая функция F задаётся выражением  $y \wedge \neg z \wedge (\neg x \vee w)$ . На рисунке приведен фрагмент таблицы истинности функции F, содержащий все наборы аргументов, при которых функция F истинна. Определите, какому столбцу таблицы истинности функции F соответствует каждая из переменных w, x, y, z.

Перем. 1	Перем. 2	Перем. 3	Перем. 4	Функция
???	???	???	???	F
1	0	0	0	1
1	0	0	1	1
1	1	0	1	1

В ответе напишите буквы x, y, z в том порядке, в котором идут соответствующие им столбцы (сначала – буква, соответствующая 1-му столбцу; затем – буква, соответствующая 2-му столбцу; затем – буква, соответствующая 3-му столбцу). Буквы в ответе пишите подряд, никаких разделителей между буквами ставить не нужно.

*Пример.* Пусть задано выражение  $x \rightarrow y$ , зависящее от двух переменных x и y, и таблица истинности:

Перем. 1	Перем. 2	Функция
???	???	F
0	0	1
0	1	1
1	0	0
0	1	1

Тогда 1-му столбцу соответствует переменная y, а 2-му столбцу соответствует переменная x. В ответе нужно написать: yx.

Ответ: \_\_\_\_\_.

**3** Между населёнными пунктами A, B, C, D, E, F построены дороги, протяжённость которых приведена в таблице. (Отсутствие числа в таблице означает, что прямой дороги между пунктами нет.)

	A	B	C	D	E	F
A		3	5	9		20
B	3			6		
C	5			6		
D	9	6	6		4	5
E				4		2
F	20			5	2	

Определите длину кратчайшего пути между пунктами A и F (при условии, что передвигаться можно только по построенным дорогам).

Ответ: \_\_\_\_\_.





4

В фрагменте базы данных представлены сведения о родственных отношениях. На основании приведённых данных определите, сколько прямых потомков (т.е. детей и внуков) Пригула А.К. упомянуты в таблице 1.

ID	Фамилия И.О.	Пол
2294	Решко Л.П.	Ж
3039	Пригула А.К.	М
3043	Вирченко В.А.	Ж
2232	Плиев Г.А.	М
2659	Пригула Е.А.	Ж
2144	Вирченко Н.А.	Ж
2278	Пригула И.А.	М
2849	Ложкина Т.Х.	Ж
2158	Король А.П.	М
2487	Брик А.И.	Ж
3021	Пригула П.И.	М
2494	Пригула А.И.	Ж
2565	Мунтян С.А.	Ж
2876	Король П.А.	М

ID_Родителя	ID_Ребёнка
2294	2659
2294	2278
3039	2659
3039	2278
2659	3043
2659	2565
2659	2876
2278	3021
2278	2487
2849	3021
2849	2487
2158	3043
2158	2565
2158	2876

Ответ: \_\_\_\_\_.

5

Для кодирования некоторой последовательности, состоящей из букв К, Л, М, Н, решили использовать неравномерный двоичный код, удовлетворяющий условию Фано. Для буквы Н использовали кодовое слово 1, для буквы К — кодовое слово 01. Какова наименьшая возможная суммарная длина всех четырёх кодовых слов?

Ответ: \_\_\_\_\_.

6

На вход алгоритма подаётся натуральное число N. Алгоритм строит по нему новое число R следующим образом.

1. Строится двоичная запись числа N.
2. К этой записи дописываются справа ещё два разряда по следующему правилу:
  - а) складываются все цифры двоичной записи, и остаток от деления суммы на 2 дописывается в конец числа (справа). Например, запись 11100 преобразуется в запись 111001;
  - б) над этой записью производятся те же действия – справа дописывается остаток от деления суммы цифр на 2.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью искомого числа R. Укажите такое наименьшее число N, для которого результат работы алгоритма больше 100. В ответе это число запишите в десятичной системе счисления.

Ответ: \_\_\_\_\_.

7

Дан фрагмент электронной таблицы. Из ячейки A1 в ячейку B2 была скопирована формула. При копировании адреса ячеек в формуле автоматически изменились. Каким стало числовое значение в ячейке B2?

	A	B	C	D
1	=B\$1+\$C2	50	10	1
2	2		20	2
3	3	70	30	3
4	4	80	40	4

Примечание: знак \$ обозначает абсолютную адресацию.

Ответ: \_\_\_\_\_.

8

Запишите число, которое будет напечатано в результате выполнения следующей программы:

**Паскаль**

```
var n, s: integer;
begin
  n := 1;
  s := 0;
  while n <= 250 do begin
    s := s + 1;
    n := n * 3
  end;
  write(s)
end.
```

Ответ: \_\_\_\_\_.

9

Для хранения произвольного растрового изображения размером 512x128 пикселей отведено 64 Кбайт памяти. Для каждого пикселя для хранения кода отведено одинаковое количество бит. Какое максимально количество цветов можно использовать в изображении?

Ответ: \_\_\_\_\_.

10

Сколько слов длины 4 можно составить из букв Е, Г, Э, С, Т, Р, А, Ш, Н, О, при условии, что первая и последняя буквы - только гласные, а остальные - только согласные? Каждая буква может входить в слово несколько раз. Слова не обязательно должны быть осмысленными словами русского языка.

Ответ: \_\_\_\_\_.

11

Ниже на записан рекурсивный алгоритм F:

**Паскаль**

```
procedure F(n: integer);
begin
  writeln(n);
  if n > 0 then begin
    writeln(n);
    F(n - 2);
    F(n - 3)
  end
end;
```

Чему равна сумма напечатанных на экране чисел при выполнении вызова F(7)?

Ответ: \_\_\_\_\_.

12

В терминологии сетей TCP/IP маска сети — это двоичное число, меньшее  $2^{32}$ ; в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места нули. Маска определяет, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес – в виде четырёх байт, причём каждый байт записывается в виде десятичного числа. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. Например, если IP-адрес узла равен 231.32.255.131, а маска равна 255.255.240.0, то адрес сети равен 231.32.240.0.

Для узла с IP-адресом 192.168.54.10 адрес сети равен 192.168.0.0. Какое наибольшее число единиц может быть в разрядах маски?

Ответ: \_\_\_\_\_.



**13** При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 13 символов и содержащий только символы латинского алфавита (26 символов) и восьмеричные цифры. Каждый такой пароль в компьютерной программе записывается минимально возможным и одинаковым целым количеством байт, при этом используют посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством бит. Определите, сколько байт необходимо для хранения 60 паролей.

Ответ: \_\_\_\_\_.

**14** Исполнитель Редактор получает на вход строку цифр и преобразовывает её. Редактор может выполнять две команды, в обеих командах v и w обозначают цепочки цифр.

А) **заменить** (v, w).

Эта команда заменяет в строке первое слева вхождение цепочки v на цепочку w. Например, выполнение команды **заменить** (111, 27)

преобразует строку 05111150 в строку 0527150.

Если в строке нет вхождений цепочки v, то выполнение команды **заменить** (v, w) не меняет эту строку.

Б) **нашлось** (v).

Эта команда проверяет, встречается ли цепочка v в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь». Строка исполнителя при этом не изменяется.

Цикл

ПОКА условие  
    последовательность команд  
КОНЕЦ ПОКА

выполняется, пока условие истинно.

В конструкции

ЕСЛИ условие  
    ТО команда1  
    ИНАЧЕ команда2  
КОНЕЦ ЕСЛИ

выполняется команда1 (если условие истинно) или команда2 (если условие

ложно).

Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 70 идущих подряд цифр 5? В ответе запишите полученную строку.

НАЧАЛО  
ПОКА **нашлось** (5555) ИЛИ **нашлось** (7777)  
    ЕСЛИ **нашлось** (5555)  
        ТО **заменить** (5555, 7)  
        ИНАЧЕ **заменить** (7777, 5)  
    КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

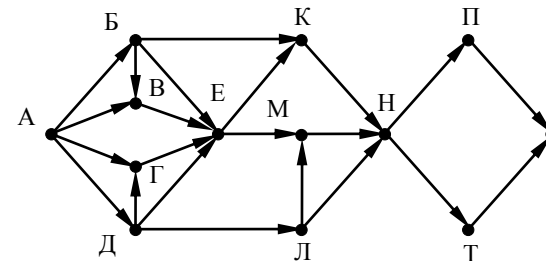
КОНЕЦ

Ответ: \_\_\_\_\_.

**15** На рисунке – схема дорог, связывающих города А, Б, В, Г, Д, Е, К, Л, М, Н, П, Р, Т.

По каждой дороге можно двигаться только в одном направлении, указанном стрелкой.

Сколько существует различных путей, ведущих из города А в город Т и не проходящих через город М?



Ответ: \_\_\_\_\_.



**16** Значение арифметического выражения  $16^3 + 4^{10} - 26$  записали в системе счисления с основанием 4. Какова сумма цифр, которые содержатся в этой записи?

Ответ: \_\_\_\_\_.

**17** В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» — символ «&». В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

Запрос	Найдено страниц (в тысячах)
<i>Гибсон</i>	350
<i>Гибсон   Фендер   Ямаха</i>	600
<i>Гибсон &amp; Фендер &amp; Ямаха</i>	0
<i>Гибсон &amp; Ямаха</i>	50
<i>Гибсон &amp; Фендер</i>	110
<i>Фендер &amp; Ямаха</i>	160

Какое количество страниц (в тыс.) будет найдено по запросу  
*Фендер | Ямаха*

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Ответ: \_\_\_\_\_.

**18** Обозначим через  $m \& n$  поразрядную конъюнкцию неотрицательных целых чисел  $m$  и  $n$ . Так, например,  $14 \& 5 = 1110_2 \& 0101_2 = 0100_2 = 4$ .

Для какого наименьшего неотрицательного целого числа  $A$  формула

$$x \& 30 = 0 \vee x \& 50 \neq 0 \vee x \& A \neq 0$$

тождественно истинна (т.е. принимает значение 1 при любом неотрицательном целом значении переменной  $x$ )

Ответ: \_\_\_\_\_.

**19** В программе используется одномерный целочисленный массив  $A$  с индексами от 1 до 10. Значения элементов равны 5, 8, 10, 4, 6, 12, 7, 3, 8, 2 соответственно, т.е.  $A[1] = 5$ ;  $A[2] = 8$  и т.д. Определите значение переменной  $s$  после выполнения следующего фрагмента программы:

```

Паскаль
s := 0;
for i := 1 to 9 do
  if A[i] > A[i+1] then begin
    s := s + 1;
    t := A[i];
    A[i] := A[i+1];
    A[i+1] := t;
  end;
    
```

Ответ: \_\_\_\_\_.

**20** Ниже записан алгоритм. При каком наименьшем  $x$  алгоритм печатает сначала 3, а потом 16?

```

Паскаль
var x, a, b: integer;
begin
  readln(x);
  a := 0; b := 0;
  while x > 0 do begin
    a := a + 1;
    b := b + (x mod 8);
    x := x div 8;
  end;
  writeln(a); write(b);
end.
    
```

Ответ: \_\_\_\_\_.



ТРЕНИРОВОЧНЫЙ КИМ №051602



21

Определите, какое число будет напечатано в результате выполнения следующего алгоритма.

**Паскаль**

```
var a,b,t,M: integer;
Function F(x: integer):integer;
begin
  F := -(x+50)*(x+10);
end;
BEGIN
  a := -50; b := 50;
  M := 0;
  for t := a to b do begin
    if (F(t) > 0) then begin
      M := M + 1;
    end
  end;
  write(M)
END.
```

Ответ: \_\_\_\_\_.

22

Исполнитель Тренер преобразует целое число, записанное на экране. У исполнителя три команды, каждой команде присвоен номер:

**1. Прибавь 1**

**2. Умножь на 2**

Первая команда увеличивает число на экране на 1, вторая умножает его на 2. Сколько существует программ, которые число 5 преобразуют в число 30 при условии, что траектория вычислений содержит число 10?

Ответ: \_\_\_\_\_.

23

Сколько различных решений имеет система уравнений?

$$\begin{aligned} &((\neg x_1 \vee x_2) \wedge (\neg x_1 \vee y_1)) \wedge (y_1 \vee \neg y_2) = 1 \\ &((\neg x_2 \vee x_3) \wedge (\neg x_2 \vee y_2)) \wedge (y_2 \vee \neg y_3) = 1 \\ &((\neg x_3 \vee x_4) \wedge (\neg x_3 \vee y_3)) \wedge (y_3 \vee \neg y_4) = 1 \\ &((\neg x_4 \vee x_5) \wedge (\neg x_4 \vee y_4)) \wedge (y_4 \vee \neg y_5) = 1 \\ &((\neg x_5 \vee x_6) \wedge (\neg x_5 \vee y_5)) \wedge (y_5 \vee \neg y_6) = 1 \\ &\neg x_6 \rightarrow y_6 = 1 \end{aligned}$$

где  $x_1, x_2, \dots, x_6, y_1, y_2, \dots, y_6$  – логические переменные?

В ответе не нужно перечислять все различные наборы значений переменных, при которых выполнено данное равенство.

В качестве ответа нужно указать количество таких наборов.

Ответ: \_\_\_\_\_.

**Не забудьте перенести все ответы в бланк ответов № 1 в соответствии с инструкцией по выполнению работы.**



## Часть 2

Для записи ответов на задания этой части (24–27) используйте БЛАНК ОТВЕТОВ № 2. Запишите сначала номер задания (24, 25 и т. д.), а затем полное решение. Ответы записывайте чётко и разборчиво.

24

- 1) Требовалось написать программу, которая решает уравнение " $a|x| = b$ " относительно  $x$  для любых чисел  $a$  и  $b$ , введенных с клавиатуры. Все числа считаются действительными. Программист торопился и написал программу неправильно:

## Паскаль

```
#include <stdio.h>
int 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 решений");
else
if (b == 0)
printf("x = 0");
else printf("x = %f или x = %f", b/a, -b/a);
}
```

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.)

25

Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от  $-10\,000$  до  $10\,000$  включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, сумма которых нечётна и положительна. Под парой подразумевается два подряд идущих элемента массива.

Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но использовать все описанные переменные не обязательно.

## Паскаль

```
const N = 20;
var a: array [1..N] of integer;
    i, j, k: integer;
begin
for i := 1 to N do
    readln(a[i]);
...
end.
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия.





26

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может

- а) добавить в одну из куч (по своему выбору) один камень или
- б) увеличить количество камней в куче в два раза.

Игра завершается в тот момент, когда количество камней в куче становится не менее 24. Если при этом в куче оказалось не более 38 камней, то победителем считается игрок, сделавший последний ход. В противном случае победителем становится его противник. Например, если в куче был 21 камень и Петя удвоит количество камней в куче, то игра закончится и победителем будет Ваня. В начальный момент в куче было  $S$  камней,  $1 < S < 23$ .

Задание 1. а) При каких значениях числа  $S$  Петя может выиграть в один ход?

Укажите все такие значения и соответствующие ходы Пети.

б) У кого из игроков есть выигрышная стратегия при  $S = 22, 21, 20$ ? Опишите выигрышные стратегии для этих случаев.

Задание 2. У кого из игроков есть выигрышная стратегия при  $S = 11, 10$ ?

Опишите соответствующие выигрышные стратегии.

Задание 3. У кого из игроков есть выигрышная стратегия при  $S = 9$ ? Постройте дерево всех партий, возможных при этой выигрышной стратегии (в виде рисунка или таблицы). На рёбрах дерева указывайте, кто делает ход; в узлах – количество камней в позиции.

27

На вход программе подаются сведения о номерах школ учащихся, участвовавших в олимпиаде. 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-х 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 (укажите используемую версию языка программирования, например, Borland Pascal 7.0), 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  $N \geq 1000$ .



## Система оценивания экзаменационной работы по информатике и ИКТ

## Часть 1

За правильный ответ на задания 1–23 ставится 1 балл; за неверный ответ или его отсутствие – 0 баллов.

№ задания	Ответ
1	55
2	yxzw
3	14
4	7
5	9
6	25
7	40
8	6
9	256
10	576
11	42
12	18
13	600
14	7755
15	16
16	15
17	410
18	30
19	6
20	191
21	39
22	14
23	13

24

Требовалось написать программу, которая решает уравнение " $a|x| = b$ " относительно  $x$  для любых чисел  $a$  и  $b$ , введенных с клавиатуры. Все числа считаются действительными. Программист торопился и написал программу неправильно:

Паскаль

```
#include <stdio.h>
int 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 решений");
    else
    if (b == 0)
        printf("x = 0");
    else printf("x = %f или x = %f", b/a, -b/a);
}
```

Последовательно выполните три задания:

- 4) Приведите пример таких чисел  $a$ ,  $b$ ,  $x$ , при которых программа неверно решает поставленную задачу.
- 5) Укажите, какая часть программы является лишней.

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)





**Содержание верного ответа**

(допускаются иные формулировки ответа, не искажающие его смысла)

Решение использует запись программы на Паскале. Допускается использование программы на других языках.

Программа работает неправильно, если  $a$  и  $b$  не равны нулю и имеют разные

знаки: в этом случае уравнение  $a|x| = b$  не имеет решений (поскольку модуль – неотрицательная величина), а программа выдаст два решения. Хотя в задании сказано «Приведите пример таких чисел  $a, b, x, \dots$ », значение  $x$  ни на что не влияет (см. далее), в ответе можно указать любое число  $x$ . Например,  $a = 1, b = -1, x = 0$

Лишняя часть программы – ввод  $x$ , поскольку это не исходные данные, а результат. Поэтому вместо оператора

```
readln(a,b,x);
```

правильнее написать

```
readln(a,b);
```

Переменная  $x$  далее не используется, поэтому ее не нужно объявлять:

```
var a,b: real;
```

Возможная доработка программы – добавить еще один условный оператор, обрабатывающий неучтенный случай ( $a$  и  $b$  не равны нулю и имеют разные знаки), при котором нет решений:

```
var a,b: real;
```

```
begin
```

```
readln(a,b);
```

```
if a = 0 then
```

```
  if b = 0 then
```

```
    write ('любое число')
```

```
  else write ('нет решений')
```

```
else
```

```
  if b = 0 then
```

```
    write('x = 0')
```

```
  else
```

```
    if a*b < 0 then
```

```
      write('нет решений')
```

```
    else write('x =',b/a,' или x =',-b/a);
```

```
end.
```

обратите внимание, что для проверки условия « $a$  и  $b$  имеют разные знаки»

использовано произведение  $a*b$ , которое больше нуля, когда два значения имеют одинаковые знаки, и меньше нуля – когда разные

25

Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от  $-10\,000$  до  $10\,000$  включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, сумма которых нечётна и положительна. Под парой подразумевается два подряд идущих элемента массива. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но использовать все описанные переменные обязательно.

Паскаль

```
const N = 20;
var a: array [1..N] of integer;
    i, j, k: integer;
begin
  for i := 1 to N do
    readln(a[i]);
  ...
end.
```

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия.

**Содержание верного ответа**

(допускаются иные формулировки ответа, не искажающие его смысла)

Записать значение 0 в переменную  $k$ . Затем в цикле перебрать все пары соседних элементов, начиная с пары  $(a[1], a[2])$  до пары  $(a[N-1], a[N])$ . Для каждой пары вычислить сумму, если эта сумма положительна и нечётна (остаток от её деления на 2 не равен нулю), увеличить значение  $k$  на единицу. После завершения цикла вывести значение  $k$ .

26

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может

а) добавить в одну из куч (по своему выбору) один камень или

б) увеличить количество камней в куче в два раза.

Игра завершается в тот момент, когда количество камней в куче становится не

менее 24. Если при этом в куче оказалось не более 38 камней, то победителем

считается игрок, сделавший последний ход. В противном случае победителем становится его противник. Например, если в куче был 21 камень и Петя удвоит количество камней в куче, то игра закончится и победителем будет Ваня. В начальный момент в куче было  $S$  камней,  $1 < S < 23$ .

Задание 1. а) При каких значениях числа  $S$  Петя может выиграть в один ход? Укажите все такие значения и соответствующие ходы Пети.

б) У кого из игроков есть выигрышная стратегия при  $S = 22, 21, 20$ ? Опишите выигрышные стратегии для этих случаев.

Задание 2. У кого из игроков есть выигрышная стратегия при  $S = 11, 10$ ? Опишите соответствующие выигрышные стратегии.

Задание 3. У кого из игроков есть выигрышная стратегия при  $S = 9$ ? Постройте дерево всех партий, возможных при этой выигрышной стратегии (в виде рисунка или таблицы). На рёбрах дерева указывайте, кто делает ход; в узлах – количество камней в позиции.

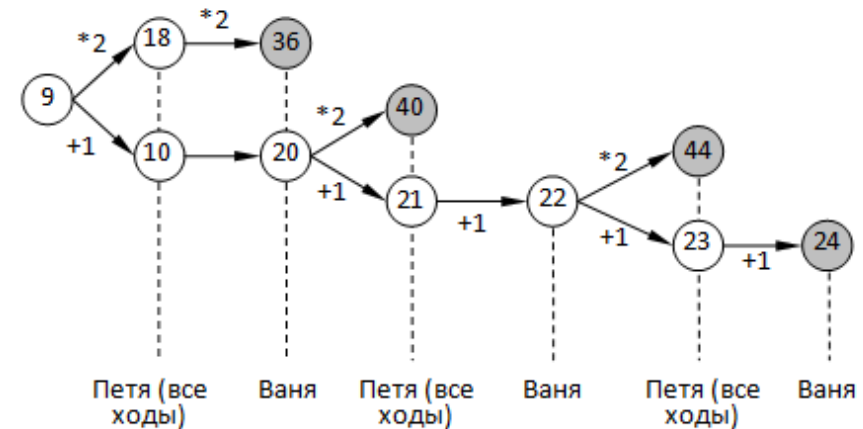
**Содержание верного ответа**

(допускаются иные формулировки ответа, не искажающие его смысла)

- Задание 1а.** Сложность состоит в том, что Петя проиграет, если в результате его хода количество камней станет больше, чем 38. Он может сделать ход «+1» или «\*2». Ходом «+1» он сможет получить 24 камня в куче (и таким образом выиграет!) из позиции  $S = 23$ . Теперь проверим ход «\*2». Для выигрыша Пети количество камней в результате этого хода должно стать от 24 до 38, поэтому Петя выиграет этим ходом при  $S$  от 12 до 19.
- Задание 1б.** При  $S = 22$  возможные ходы дают кучи в 23 и 44 камня. В первом случае ( $S = 23$ ) противник оказывается в выигрышной позиции (см. предыдущий пункт), во втором случае тот, кто ходит, проигрывает, потому что  $44 > 38$ . Поэтому позиция  $S = 22$  – проигрышная, Петя проиграет, у Вани есть выигрышная стратегия: в случае  $S = 23$  сделать ход «+1». При  $S = 21$  Петя может перевести игру в позицию  $S = 22$ , она, как мы только что показали, проигрышная для Вани. Поэтому у Пети есть выигрышная стратегия. При  $S = 20$  ходом «+1» Петя переведет игру в выигрышную (для Вани) позицию, а при ходе «\*2» он сразу проиграет, получив  $40 > 38$  камней. Поэтому выигрышная стратегия есть у Вани.
- Задание 2.** При  $S = 11$  или  $S = 10$  Петя может ходом «\*2» перевести игру в позиции  $S = 22$  и  $S = 20$ , обе они, как мы показали в

предыдущем пункте, проигрышные. Поэтому выигрышную стратегию имеет Петя.

- Задание 3.** При  $S = 9$  возможно 2 хода: ход «+1» приводит к позиции  $S = 10$ , она выигрышная (см. предыдущий пункт); ход «\*2» приводит к позиции  $S = 18$ , она тоже выигрышная (см. первый пункт). Таким образом, все возможные ходы ведут в выигрышные для соперника позиции, и позиция  $S = 9$  – проигрышная (для Пети). Выигрышную стратегию имеет Ваня. При построении дерева для проигрывающего (Пети) указываем все возможные ходы, а для выигрывающего (Вани) – только один выигрышный ход. Дерево можно нарисовать так:



27

На вход программе подаются сведения о номерах школ учащихся, участвовавших в олимпиаде. 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-х 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 (укажите используемую версию языка программирования, например, Borland Pascal 7.0), 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>=1000.

**Содержание верного ответа**

(допускаются иные формулировки ответа, не искажающие его смысла)

по условию, единственная информация, которая нам нужна в итоге для вывода результата – это количество участников по каждой школе так как номер школы состоит (по условию!) не более, чем из двух цифр, всего может быть не более 99 школ (с номерами от 1 до 99) поэтому можно ввести массив C из 99 элементов; для всех k от 1 до 99 элемент C[k] будет ячейкой-счетчиком, в которой накапливается число участников от школы с номером k; сначала во все элементы этого массива записываются нуль (обнуление счетчиков):  
 for k:=1 to 99 do C[k]:=0;  
 во многих системах программирования на Паскале все глобальные переменные автоматически обнуляются, и таким образом, этот цикл ничего не дает; однако на всякий случай нужно продемонстрировать эксперту, который будет проверять часть C вашей работы, что вы понимаете суть дела («счетчик необходимо сначала обнулить»)  
 основной цикл обработки вводимых строк можно записать на псевдокоде так:  
 for i:=1 to N do begin  
   { читаем очередную строку }  
   { определяем номер школы k }  
   C[k] := C[k] + 1; { увеличиваем счетчик k-ой школы }  
 end;  
 поскольку данные вводятся в виде символьной строки, нужно выделить в памяти переменную s типа string  
 для чтения очередной строки будем использовать оператор readln

остается понять, как выделить из строки номер школы; по условию он закодирован в последней части строки, после второго пробела; значит, нужно найти этот второй пробел, вырезать из строки весь «хвост» после этого пробела, и преобразовать его из символьного формата в числовой чтобы найти первый пробел и «отрезать» первую часть строки с этим пробелом, можно использовать команды

```
p := Pos(' ', s);
s := Copy(s, p+1, Length(s)-p);
```

первая команда определяет номер первого пробела и записывает его в целую переменную p, в вторая – записывает в строку s весь «хвост», стоящий за этим пробелом, начиная с символа с номером p+1; длина хвоста равна Length(s)-p, где Length(s) – длина строки;

поскольку нас интересует часть после второго пробела, эти две строчки нужно повторить два раза, в результате в переменной s окажется символьная запись номера школы;

заметим, что можно избежать дублирования двух строк, «свернув» их во внутренний цикл, но это вряд ли сильно упростит запись:

```
for k:=1 to 2 do begin
  p := Pos(' ', s);
  s := Copy(s, p+1, Length(s)-p);
end;
```

в пп. 8-10 описан достаточно общий метод, при котором инициалы могут быть любой длины, (но без пробела); в данном случае в условии четко сказано, что инициалы представляют собой именно 4 символа (буква, точка, буква, точка), поэтому можно найти первый пробел, а затем взять «хвост», который идет через 6 символов от него:

```
p := Pos(' ', s);
s := Copy(s,p+6,Length(s));
```

или так

```
p := Pos(' ', s);
Delete(s, 1, p-5);
```

для преобразования номера школы из символьного вида в числовой можно использовать функцию Val:

```
Val(s, k, r);
```

эта процедура (Turbo Pascal, Borland Pascal, PascalABC, среда АЛГО) преобразует символьную строку s в числовое значение k; с помощью переменной r обнаруживается ошибка: если раскодировать число не удалось (в строке не число), в r будет записан нуль (здесь мы не будем обрабатывать эту ошибку, полагая, что все данные правильные); если вы работаете на ПаскалеABC (никто не может вам запретить написать, что этот так), вместо Val можно использовать более удобную и понятную функцию StrToInt:  
 k := StrToInt(s);  
 таким образом, основной цикл выглядит так:  
 for i:=1 to N do begin



ТРЕНИРОВОЧНЫЙ КИМ №051602



```

readln(s); { читаем очередную строку }
  { выделяем часть после второго пробела }
p := Pos(' ', s);
Delete(s, 1, p+5);
  { определяем номер школы k }
Val(s, k, r);
C[k] := C[k] + 1; { увеличиваем счетчик k-ой школы }
end;
дальше стандартным алгоритмом определяем в массиве C минимальный
элемент Min, не учитывая нули (школы, из которых не было участников):
Min := N;
for k:=1 to 99 do
  if (C[k] <> 0) and (C[k]<Min) then Min := C[k];
здесь интересна первая строчка, Min:=N: по условию всего было N участников,
поэтому минимальное значение не может быть больше N; обратите внимание,
что привычный вариант (который начинается с Min:=C[1]) работает неверно,
если из первой школы не было ни одного участника
и выводим на экран номера всех школ (обратите внимание – номера!), для
которых C[k]=Min:
for k:=1 to 99 do
  if C[k] = Min then writeln(k);
остается «собрать» программу, чтобы получилось полное решение;
максимальное количество школ мы задали в виде константы LIM:
const LIM = 99;
var C:array[1..LIM] of integer;
  i, p, N, k, r, Min: integer;
  s:string;
begin
for k:=1 to 99 do C[k]:=0;
readln(N);
for i:=1 to N do begin
  readln(s); { читаем очередную строку }
  { выделяем часть после второго пробела }
  p := Pos(' ', s);
  Delete(s, 1, p+5);
  { определяем номер школы k }
  Val(s, k, r);
  C[k] := C[k] + 1; { увеличиваем счетчик k-ой школы }
end;
Min := N;
for k:=1 to LIM do
  if (C[k] <> 0) and (C[k]<Min) then Min := C[k];
for k:=1 to LIM do

```

```

if C[k] = Min then writeln(k);
end.

```

