

## 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.

- Обозначения для логических связок (операций):
  - отрицание (инверсия, логическое НЕ) обозначается  $\neg$  (например,  $\neg A$ );
  - конъюнкция (логическое умножение, логическое И) обозначается  $\wedge$  (например,  $A \wedge B$ ) либо  $\&$  (например,  $A \& B$ );
  - дизъюнкция (логическое сложение, логическое ИЛИ) обозначается  $\vee$  (например,  $A \vee B$ ) либо  $|$  (например,  $A | B$ );
  - следование (импликация) обозначается  $\rightarrow$  (например,  $A \rightarrow B$ );
  - 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);
  - 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 единицей «байт» выражается степенью двойки.



**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** Даны 4 целых числа, записанных в различных системах счисления:  $32_{10}$ ,  $FA_{16}$ ,  $234_8$ ,  $1027_{10}$ . Сколько среди них чисел, двоичная запись которых содержит ровно 6 единиц?

Ответ: \_\_\_\_\_.

**2** Логическая функция F задаётся выражением  $x \vee y \& (\neg z)$ . Определите, какому столбцу таблицы истинности функции F соответствует каждая из переменных x, y, z.

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

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: ux.

Ответ: \_\_\_\_\_.

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

	A	B	C	D	E	F	Z
A		4	9				25
B	4		3				
C	9	3		2		13	20
D			2		4		
E				4			4
F			13				1
Z	25		20		4	1	

Определите длину кратчайшего пути между пунктами A и Z (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) Решко Л.П. 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 0, для буквы К — кодовое слово 10. Какова наименьшая возможная суммарная длина всех четырёх кодовых слов?

Ответ: \_\_\_\_\_.

6

Автомат получает на вход трёхзначное число. По этому числу строится новое число по следующим правилам.

1. Складываются первая и вторая, а также вторая и третья цифры исходного числа.
2. Полученные два числа записываются друг за другом в порядке убывания (без разделителей).

*Пример.* Исходное число: 843. Суммы:  $8 + 4 = 12$ ;  $4 + 3 = 7$ . Результат: 127.

Укажите наименьшее число, в результате обработки которого автомат выдаст число 157.

Ответ: \_\_\_\_\_.

7

Дан фрагмент электронной таблицы. Из ячейки D2 в одну из ячеек диапазона E1:E4 была скопирована формула. При копировании адреса ячеек в формуле автоматически изменились, и значение формулы стало равным 8. В какую ячейку была скопирована формула? В ответе укажите только одно число – номер строки, в которой расположена ячейка.

	A	B	C	D	E
1	1	2	3	4	
2	2	3	4	=B\$3+\$C2	
3	3	4	5	6	
4	4	5	6	7	

*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 <= 100 do begin
    s := s + 30;
    n := n * 2;
  end;
  write(s)
end.
```

Ответ: \_\_\_\_\_.

9

Производилась двухканальная (стерео) звукозапись с частотой дискретизации 64 кГц и 24-битным разрешением. В результате был получен файл размером 48 Мбайт, сжатие данных не производилось. Определите приблизительно, сколько времени (в минутах) проводилась запись? В качестве ответа укажите ближайшее к времени записи целое число.

Ответ: \_\_\_\_\_.

10

Сколько слов длины 4, начинающихся с согласной буквы и заканчивающихся гласной буквой, можно составить из букв Л, Е, Т, О? 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**

```
function F(n: integer): integer;
begin
  if n > 2 then
    F := F(n-1)+F(n-2)
  else
    F := 1;
end;
```

Чему будет равно значение, вычисленное алгоритмом при выполнении вызова F(5)?

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 224.128.112.142 адрес сети равен 224.128.96.0. Чему равен третий слева байт маски? Ответ запишите в виде десятичного числа.

Ответ: \_\_\_\_\_.



**13** При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 11 символов и содержащий только символы А, Б, В, Г, Д, Е. 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 20 паролей.

Ответ: \_\_\_\_\_.

**14** Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду **сместиться на (a, b)** (где a, b – целые числа), перемещающую Чертёжника из точки с координатами (x, y) в точку с координатами (x + a, y + b). Чертёжнику был дан для исполнения следующий алгоритм:

Например, если Чертёжник находится в точке с координатами (4, 2), то команда сместиться на (2, -3) переместит Чертёжника в точку (6, -1).

```

Цикл
  ПОВТОРИ число РАЗ
    последовательность команд
  КОНЕЦ ПОВТОРИ
    
```

означает, что последовательность команд будет выполнена указанное число раз (число должно быть натуральным).

Чертёжнику был дан для исполнения следующий алгоритм (буквами n, a, b обозначены неизвестные числа, при этом n > 1):

```

НАЧАЛО
  сместиться на (-1, -2)
  ПОВТОРИ n РАЗ
    сместиться на (a, b)
    сместиться на (-1, -2)
  КОНЕЦ ПОВТОРИ
  сместиться на (-20, -12)
КОНЕЦ
    
```

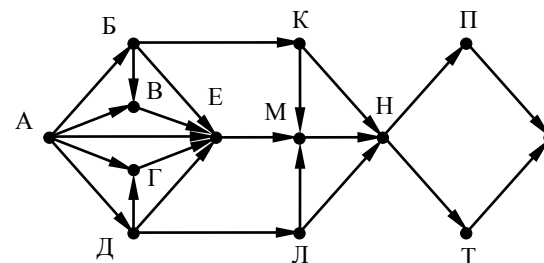
Какое наибольшее число повторений могло быть указано в конструкции «ПОВТОРИ n РАЗ»?

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?



Ответ: \_\_\_\_\_.

**16** Решите уравнение:  $121_x + 1_{10} = 101_8$

Ответ запишите в троичной системе (основание системы счисления в ответе писать не нужно).

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>Ростов &amp; (Орёл &amp; Курск   Белгород)</i>	370
<i>Ростов &amp; Белгород</i>	204
<i>Ростов &amp; Орёл &amp; Курск &amp; Белгород</i>	68

Какое количество страниц (в тыс.) будет найдено по запросу  
*Ростов & Орёл & Курск?*

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

Ответ: \_\_\_\_\_.

**18** Элементами множества *A* являются натуральные числа. Известно, что выражение

$$(x \in \{2, 4, 6, 8, 10, 12\}) \rightarrow (((x \in \{4, 8, 12, 16\}) \wedge \neg(x \in A)) \rightarrow \neg(x \in \{2, 4, 6, 8, 10, 12\}))$$

истинно (т. е. принимает значение 1) при любом значении переменной *x*.

Определите наименьшее возможное значение суммы элементов множества *A*.

Ответ: \_\_\_\_\_.

**19** В программе описан одномерный целочисленный массив с индексами от 0 до 10. Ниже представлен записанный фрагмент программы, обрабатывающей данный массив:

```

Паскаль
s := 15;
n := 10;
for i:=0 to n-3 do begin
    s:=s+A[i]-A[i+2]
end;
    
```

В начале выполнения этого фрагмента в массиве находились трёхзначные натуральные числа.

Какое наибольшее значение может иметь переменная *s* после выполнения данной программы?

Ответ: \_\_\_\_\_.

**20** Ниже записан алгоритм. Сколько существует таких чисел *x*, при вводе которых алгоритм печатает сначала 2, а потом 22?

```

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 100);
        x := x div 100
    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,R,H :integer;
Function F(H, x: integer):integer;
begin
  F := (x-50)*(x+50)+7;
end;
BEGIN
  readln(H);
  a := 0; b := 50;
  M := a; R := F(a);
  for t := a to b do begin
    if (F(t) > R) then begin
      M := t;
      R := F(t)
    end
  end;
  write(R)
END.

```

Ответ: \_\_\_\_\_.

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

1. Прибавь 1
2. Прибавь 2
3. Прибавь 5

Первая команда увеличивает число на экране на 1, вторая увеличивает это число на 2, третья – на 5. Сколько существует программ, которые число 21 преобразуют в число 30.

Ответ: \_\_\_\_\_.

- 23 Сколько различных решений имеет система уравнений?

$$\begin{aligned}
 (x_1 \rightarrow x_2) \wedge (x_2 \rightarrow x_3) \wedge (x_3 \rightarrow x_4) \wedge (x_4 \rightarrow x_5) &= 1 \\
 (y_1 \rightarrow y_2) \wedge (y_2 \rightarrow y_3) \wedge (y_3 \rightarrow y_4) \wedge (y_4 \rightarrow y_5) &= 1 \\
 (z_1 \rightarrow z_2) \wedge (z_2 \rightarrow z_3) \wedge (z_3 \rightarrow z_4) \wedge (z_4 \rightarrow z_5) &= 1 \\
 x_1 \vee y_1 \vee z_1 &= 1
 \end{aligned}$$

где  $x_1, x_2, \dots, x_5, y_1, y_2, \dots, y_5, z_1, z_2, \dots, z_5$  – 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

Требовалось написать программу, при выполнении которой с клавиатуры считывается положительное целое число  $N$ , не превосходящее  $10^9$ , и определяется сумма цифр этого числа. Программист торопился и написал программу неправильно:

**Паскаль**

```
var N: longint;
sum, d: integer;
begin
  readln(N);
  sum := 1;
  while N > 0 do begin
    d := N mod 10;
    N := N div 10;
    sum := d;
  end;
  writeln(sum);
end.
```

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе числа 256.
2. Приведите пример такого трёхзначного числа, при вводе которого программа выведет правильный ответ.
3. Найдите в программе все ошибки (их может быть одна или несколько).

Для каждой ошибки:

- 1) выпишите строку, в которой она допущена;
- 2) укажите, как исправить ошибку, т. е. приведите правильный вариант строки.

25

Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от  $-10\,000$  до  $10\,000$  включительно. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, в которых сумма элементов делится на 3, но не делится на 9. В данной задаче под парой подразумеваются два соседних элемента массива.

**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 добавить в кучу один камень или увеличить количество камней в куче в шесть раз.

Например, имея кучу из 10 камней, за один ход можно получить кучу из 11 или 60 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится не менее 361. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 361 или больше камней.

В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 360$ .

Говорят, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника.

#### Задание 1.

- При каких значениях числа  $S$  Петя может выиграть первым ходом? Укажите все такие значения и выигрывающий ход Пети;
- Укажите такое значение  $S$ , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом. Опишите выигрышную стратегию Вани.

**Задание 2.** Укажите два значения  $S$ , при которых у Пети есть выигрышная стратегия, причём (а) Петя не может выиграть первым ходом, но (б) Петя может выиграть своим вторым ходом, независимо от того, как будет ходить Ваня. Для указанных значений  $S$  опишите выигрышную стратегию Пети.

**Задание 3.** Укажите такое значение  $S$ , при котором у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети, и при этом у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом. Для указанного значения  $S$  опишите выигрышную стратегию Вани. Постройте дерево всех партий, возможных при этой выигрышной стратегии Вани (в виде рисунка или таблицы). На рёбрах дерева указывайте, кто делает ход, в узлах — количество камней в позиции.

27

По каналу связи передаются положительные целые числа, не превышающие 1000, — результаты измерений, полученных в ходе эксперимента (количество измерений известно заранее). После окончания эксперимента передаётся контрольное значение — наименьшее число  $R$ , удовлетворяющее следующим условиям:

1)  $R$  — сумма двух различных переданных элементов последовательности («различные» означает, что нельзя просто удваивать переданные числа, суммы различных, но равных по величине элементов допускаются);

2)  $R$  — нечётное число.

Если чисел, соответствующих приведённым условиям, нет, считается, что  $R = -1$ . В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Free Pascal 2.6.4), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

Вычисленное контрольное значение: ...

Контроль пройден (или Контроль не пройден)

Если удовлетворяющее условию контрольное значение определить невозможно (то есть  $R = -1$ ), то выводится только фраза «Контроль не пройден».

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел  $N$ . В каждой из последующих  $N$  строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

Пример входных данных:

```
6
10
8
33
45
19
200
27
```

Пример выходных данных для приведённого выше примера входных данных:

Вычисленное контрольное значение: 27

Контроль пройден



## Система оценивания экзаменационной работы по информатике и ИКТ

## Часть 2

## Часть 1

## Критерии оценивания заданий с развернутым ответом

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

№ задания	Ответ
1	1
2	zxy
3	17
4	7
5	9
6	169
7	1
8	210
9	2
10	128
11	5
12	224
13	100
14	7
15	22
16	21
17	234
18	24
19	1798
20	121
21	10007
22	75
23	91

24

Требовалось написать программу, при выполнении которой с клавиатуры считывается положительное целое число  $N$ , не превосходящее  $10^9$ , и определяется сумма цифр этого числа. Программист торопился и написал программу неправильно:

**Паскаль**

```
var N: longint;
sum, d: integer;
begin
  readln(N);
  sum := 1;
  while N > 0 do begin
    d := N mod 10;
    N := N div 10;
    sum := d;
  end;
  writeln(sum);
end.
```

Последовательно выполните следующее.

4. Напишите, что выведет эта программа при вводе числа 256.
5. Приведите пример такого трёхзначного числа, при вводе которого программа выведет правильный ответ.
6. Найдите в программе все ошибки (их может быть одна или несколько).  
Для каждой ошибки:
  - 3) выпишите строку, в которой она допущена;
  - 4) укажите, как исправить ошибку, т. е. приведите правильный вариант строки.



<b>Содержание верного ответа</b> (допускаются иные формулировки ответа, не искажающие его смысла)
<p>Решение использует запись программы на Паскале. Допускается использование программы на других языках.</p> <p>1. Программа выдаст 2.</p> <p>2. Пример числа, для которого программа выдаст верный результат: 100. Заметим, что программа выдаёт верный результат для любого трёхзначного числа, у которого сумма цифр равна цифре в старшем разряде.</p> <p>3. Возможные варианты исправления для языка Паскаль:</p> <p>1) исправление инициализации суммы: Было: <code>sum := 1</code> Нужно: <code>sum := 0</code></p> <p>2) исправление приращения суммы Было: <code>sum := d</code> Нужно: <code>sum := sum + d</code></p>

**25** Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от  $-10\,000$  до  $10\,000$  включительно. Опишите на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, в которых сумма элементов делится на 3, но не делится на 9. В данной задаче под парой подразумеваются два соседних элемента массива.

<b>Паскаль</b>
<pre> 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.</pre>

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

<b>Содержание верного ответа</b> (допускаются иные формулировки ответа, не искажающие его смысла)
<p><b>Паскаль</b></p> <pre> k := 0; for i := 1 to N-1 do   if ((a[i]+a[i+1]) mod 3=0) and ((a[i]+a[i+1]) mod 9 &lt;&gt; 0)   then inc(k); writeln(k);</pre> <p><b>Алгоритмический язык</b></p> <pre> k := 0; нц для i от 1 до N-1 если mod(a[i]+a[i+1],3)=0 и mod(a[i]+a[i+1],9) &lt;&gt; 0 то k := k+1 все кц вывод k</pre> <p><b>Бейсик</b></p> <pre> K = 0 K = 0 FOR I = 1 TO N-1 IF (A(I)+A(I+1)) MOD 3 = 0 AND (A(I)+A(I+1)) MOD 9 &lt;&gt; 0 THEN K = K+1 END IF NEXT I PRINT K</pre> <p><b>Python</b></p> <pre> k = 0 for i in range(0, n - 1):   if ((a[i]+a[i+1])%3 == 0 and (a[i]+a[i+1])%9 &lt;&gt; 0):     k += 1 print(k)</pre>



```

Си
k = 0;
k = 0;
for (i = 0; iif ((a[i]+a[i+1]) %3 == 0 && (a[i]+a[i+1])%9 != 0)
k++;
printf("%d", 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 добавить в кучу один камень или увеличить количество камней в куче в шесть раз.

Например, имея кучу из 10 камней, за один ход можно получить кучу из 11 или 60 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится не менее 361. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 361 или больше камней.

В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 360$ .

Говорят, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника.

### Задание 1.

- а) При каких значениях числа  $S$  Петя может выиграть первым ходом? Укажите все такие значения и выигрывающий ход Пети;
- б) Укажите такое значение  $S$ , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом. Опишите выигрышную стратегию Вани.

**Задание 2.** Укажите два значения  $S$ , при которых у Пети есть выигрышная стратегия, причём (а) Петя не может выиграть первым ходом, но (б) Петя может выиграть своим вторым ходом, независимо от того, как будет ходить Ваня. Для указанных значений  $S$  опишите выигрышную стратегию Пети.

**Задание 3.** Укажите такое значение  $S$ , при котором – у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым

ходом при любой игре Пети, и при этом – у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом. Для указанного значения  $S$  опишите выигрышную стратегию Вани. Постройте дерево всех партий, возможных при этой выигрышной стратегии Вани (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)

- а) Петя может выиграть, если  $S = 61, \dots, 360$ . При меньших значениях  $S$  за один ход нельзя получить кучу, в которой больше 360 камней. Пете достаточно увеличить количество камней в 6 раз. При  $S < 61$  получить за один ход больше 360 камней невозможно.
  - б) Ваня может выиграть первым ходом (как бы ни играл Петя), если исходно в куче будет  $S = 60$  камней. Тогда после первого хода Пети в куче будет 61 камень или 360 камней. В обоих случаях Ваня увеличивает количество камней в 6 раз и выигрывает в один ход.
  2. Возможные значения  $S$ : 10, 59. В этих случаях Петя, очевидно, не может выиграть первым ходом. Однако он может получить кучу из 60 камней (при  $S = 10$  он увеличивает количество камней в 6 раз; при  $S = 59$  — добавляет 1 камень). Эта позиция разобрана в п. 1б. В ней игрок, который будет ходить (в данном случае это Ваня), выиграть не может, а его противник (то есть Петя) следующим ходом выигрывает.
  3. Возможное значение  $S$ : 58. После первого хода Пети в куче будет 59 или 348 камней. Если в куче станет 348 камней, Ваня увеличит количество камней в 6 раз и выиграет своим первым ходом. Ситуация, когда в куче 59 камней, разобрана в п. 2. В этой ситуации игрок, который будет ходить (теперь это Ваня), выигрывает своим вторым ходом.
- В таблице изображено дерево возможных партий при описанной стратегии Вани. Заключительные позиции (в них выигрывает Ваня) подчёркнуты. На рисунке это же дерево изображено в графическом виде (оба способа изображения дерева допустимы).



И.п.	Положения после очередных ходов			
	1-й ход Пети (разобраны все ходы)	1-й ход Вани (только ход по стратегии)	2-й ход Пети (разобраны все ходы)	2-й ход Вани (только ход по стратегии)
58	$58 + 1 = 59$ $58 * 6 = 348$	$59 + 1 = 60$ $348 * 6 = 2088$	$60 + 1 = 61$ $60 * 6 = 360$	$61 * 6 = 366$ $360 * 6 = 2160$

Дерево всех партий, возможных при Ваниной стратегии.  
Знаком >> обозначены позиции, в которых партия заканчивается

27

По каналу связи передаются положительные целые числа, не превышающие 1000, – результаты измерений, полученных в ходе эксперимента (количество измерений известно заранее). После окончания эксперимента передаётся контрольное значение – наименьшее число  $R$ , удовлетворяющее следующим условиям:

1)  $R$  — сумма двух различных переданных элементов последовательности («различные» означает, что нельзя просто удваивать переданные числа, суммы различных, но равных по величине элементов допускаются);

2)  $R$  — нечётное число.

Если чисел, соответствующих приведённым условиям, нет, считается, что  $R = -1$ . В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Free Pascal 2.6.4), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

Вычисленное контрольное значение: ...

Контроль пройден (или Контроль не пройден)

Если удовлетворяющее условию контрольное значение определить невозможно (то есть  $R = -1$ ), то выводится только фраза «Контроль не пройден».

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел  $N$ . В

каждой из последующих  $N$  строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

Пример входных данных:

6  
10  
8  
33  
45  
19  
200  
27

Пример выходных данных для приведённого выше примера входных данных:

Вычисленное контрольное значение: 27

Контроль пройден

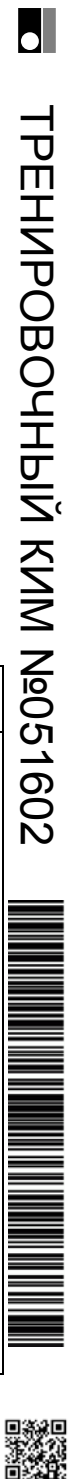
**Содержание верного ответа**  
(допускаются иные формулировки ответа, не искажающие его смысла)

Сумма двух чисел нечётна, если одно из них — чётное, а другое — нечётное. Программа, вычисляющая контрольное значение, читает все входные данные один раз, не запоминая их в массиве. Для прочитанного фрагмента входной последовательности программа хранит значения двух величин:  $M_0$  — наименьшее чётное число;  $M_1$  — наименьшее нечётное число. После того как все данные прочитаны, искомое контрольное значение вычисляется как сумма  $M_0 + M_1$ .

Ниже приведены реализующие этот алгоритм программы на языке Паскаль, а также на алгоритмическом языке. Допускаются решения, записанные на других языках программирования.

**Пример правильной и эффективной программы на языке Паскаль.**

```
var M0,M1,res,i,N,d: longint;
begin
  M0 := 1002;
  M1 := 1001;
  readln(N);
  for i := 1 to N do
    begin
```



```

readln(dat);
if (dat mod 2 = 0) and (dat < M0) then
  M0 := dat;
if (dat mod 2 <> 0) and (dat < M1) then
  M1 := dat;
end;
if (M0 <= 1000) and (M1 <= 1000) then
  res := M0+M1
else res := -1;
readln(R);
if res > 0 then
  writeln('Вычисленное контрольное значение: ',res);
if (R > 0) and (R = res) then
  writeln('Контроль пройден')
else writeln('Контроль не пройден');
end.

```

**Пример правильной и эффективной программы на Алгоритмическом языке.**

```

алг
нач
цел N | количество чисел на входе
цел x | исходные данные
цел m0=1002 | миним. четное число
цел m1=1001 | миним. нечетное число,
цел R | введенное контрольное значение
цел res | вычисленное контрольное значение
ввод N
нц N раз
ввод x
если mod(x,2) = 0 и x < m0
  то m0 := x
все
если mod(x,2) <> 0 и x < m1
  то m1 := x
все
кц
если m0 <= 1000 и m1 <= 1000

```

```

то res := m0+m1
иначе
  res := -1
все
ввод R
если res>0
  вывод нс, 'Вычисленное контрольное значение: ',res
все
если R>0 и R=res
  то вывод нс, "Контроль пройден"
иначе
  вывод нс, "Контроль не пройден"
все
кон

```

