



**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 единиц в двоичной записи числа 345<sub>8</sub>?

Ответ: \_\_\_\_\_.

**2** Логическая функция F задаётся выражением

$$((\neg x_1 \vee x_2) \wedge \neg x_3 \vee x_4) \wedge \neg x_5 \vee \neg x_6.$$

Определите, какие числа (0 или 1) пропущены в таблице истинности функции. В ответе запишите пропущенные числа в том порядке, в котором идут соответствующие им столбцы таблицы.

x1	x2	x3	x4	x5	x6	F
0	1	0	1	1	?	1
1	0	1	1	?	1	0
0	1	1	?	0	1	1

Числа в ответе пишите подряд, никаких разделителей между ними ставить не нужно.

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	6	8			25
B	4		4				
C	6	4		5			16
D	8		5		4	8	10
E				4		1	8
F				8	1		2
Z	25		16	10	8	2	

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

Ответ: \_\_\_\_\_.

**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.

В каталоге находится 7 файлов:

- acsacal.db
- cassandra.db2
- crow.db2
- curasao.dbr
- crasao.dbase
- cherry.db
- cargo.db2



Сколько файлов из данного каталога соответствует маске «\*с\*г\*.db\*»?

Ответ: \_\_\_\_\_.

- 5 Для передачи помехоустойчивых сообщений в алфавите, который содержит 16 различных символов, используется равномерный двоичный код. Этот код удовлетворяет следующему свойству: в любом кодовом слове содержится четное количество единиц (возможно, ни одной). Какую наименьшую длину может иметь кодовое слово?

Ответ: \_\_\_\_\_.

- 6 Учитель предложил детям потренироваться в действиях с шестнадцатеричными цифрами, поиграв в такую игру.

Учитель предлагает три шестнадцатеричные цифры. Ученики должны сначала найти сумму первой и второй цифр, потом — сумму второй и третьей цифр. Обе суммы должны быть записаны, как шестнадцатеричные числа. Затем эти числа записываются друг за другом в порядке убывания.

*Пример.* Исходные цифры: А, А, 3. Суммы: А+А=14; А+3=D. Результат: 14D.

Среди полученных учениками результатов были такие числа: 1E, 94, F4, G4. Сколько из этих четырех чисел действительно можно было получить в результате подобной игры?

Ответ: \_\_\_\_\_.

- 7 Дан фрагмент электронной таблицы:

	A	B	C	D	E
1	1	2	3	4	5
2	2	3	4	5	6
3	3	4	5	6	=D1+E\$2
4	4	5	6	7	8
5				?	

Из ячейки E3 в одну из ячеек диапазона A5:E5 скопировали формулу. При копировании адреса ячеек в формуле автоматически изменились. Каким в результате будет значение в ячейке D5?

*Примечание:* знак \$ используется для обозначения абсолютной адресации.

Ответ: \_\_\_\_\_.

- 8 Определите значение переменной с после выполнения следующего фрагмента программы. Ответ округлите до *целого* числа.

**Паскаль**

```
a := 30;
b := 6;
a := a/5 + b/3;
if 2*a > b then
  c:=a-5*b+27
else
  c:=a+5*b+27
```

Ответ: \_\_\_\_\_.

- 9 Какой минимальный объём памяти (в *Кбайт*) понадобится для сохранения любого растрового изображения размером 128×128 пикселей при условии, что в изображении может использоваться 16 различных цветов? В ответе запишите только *целое число* без единиц измерения.

Ответ: \_\_\_\_\_.



- 10** Друзья решили составить таблицу кодовых слов для передачи секретных сообщений. Каждому сообщению соответствует определённое кодовое слово из шести букв.

Сколько различных кодовых слов друзья могут использовать, если в словах есть только буквы **А**, **Б** и **В**, причем буква **А** используется в каждом слове, но не больше одного раза?

Ответ: \_\_\_\_\_.

- 11** Ниже записаны рекурсивные функции (процедуры) А и В:

Паскаль
<pre> <b>procedure</b> A(n: integer); <b>begin</b>     <b>if</b> n &gt; 0 <b>then</b>         B(n-1);     <b>end</b>; <b>procedure</b> B(n: integer); <b>begin</b>     <b>writeln</b>('@')     <b>if</b> n &gt; 1 <b>then</b>         A(n-3);     <b>end</b>; </pre>

Сколько *символов* будет напечатано на экране при выполнении вызова A(15)?

Ответ: \_\_\_\_\_.

- 12** В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, 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. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.

*Например*, если IP-адрес узла равен 178.65.255.19, а маска равна 255.255.240.0, то адрес сети равен 178.65.240.0.

По заданным IP-адресу узла и адресу сети определите наибольшее

возможное значение второго слева байта маски. Ответ запишите в виде десятичного числа.

IP-адрес узла: 217.119.130.15

Адрес сети: 192.103.0.13

Ответ: \_\_\_\_\_.

- 13** Автомобильный номер состоит из нескольких букв (количество букв одинаковое во всех номерах), за которыми следуют три цифры. При этом используются 10 цифр и только 5 букв: **Н**, **О**, **М**, **Е** и **Р**. Нужно иметь не менее 100 тысяч различных номеров.

Какое наименьшее количество букв должно быть в автомобильном номере?

Ответ: \_\_\_\_\_.

- 14** Исполнитель Робот умеет перемещаться по прямоугольному лабиринту, начерченному на плоскости, разбитой на клетки. Между соседними по сторонам клетками может стоять стена. Система команд исполнителя Робот содержит восемь команд. Четыре команды - это команды-приказы: **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*

*последовательность команд*

КОНЕЦ ПОКА

выполняется, пока условие истинно.

В конструкции



ЕСЛИ *условие*  
 ТО команда 1  
 ИНАЧЕ команда 2  
 КОНЕЦ ЕСЛИ

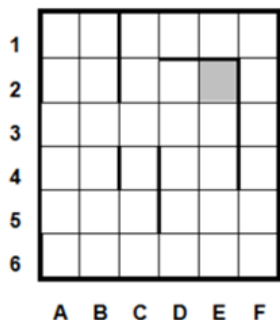
выполняется *команда 1* (если условие истинно) или *команда 2* (если условие ложно).

В конструкциях ПОКА и ЕСЛИ условие может содержать команды проверки, а также слова И, ИЛИ, НЕ, обозначающие логические операции.

Если РОБОТ начнёт движение в сторону находящейся рядом с ним стены, то он разрушится и программа прервётся.

После выполнения этого алгоритма Чертёжник возвращается в исходную точку.

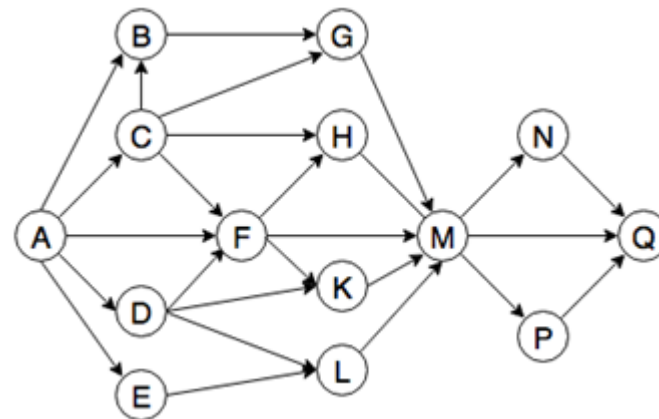
НАЧАЛО  
 ПОКА *сверху свободно ИЛИ справа свободно*  
     ЕСЛИ *справа свободно*  
         ТО *вправо*  
         ИНАЧЕ *вверх*  
     КОНЕЦ ЕСЛИ  
 КОНЕЦ ПОКА  
 КОНЕЦ



Сколько клеток лабиринта соответствуют требованию, что, начав движение в ней и выполнив предложенную программу, Робот уцелеет и остановится в закрашенной клетке (клетка E2)?

Ответ: \_\_\_\_\_.

15 На рисунке — схема дорог, связывающих города A, B, C, D, E, F, G, H, K, L, M, N, P, Q. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой.



Сколько существует различных путей из города A в город Q?

Ответ: \_\_\_\_\_.

16 В системе счисления с основанием N запись числа  $77_{10}$  заканчивается на 0, а запись числа  $29_{10}$  — на 1. Чему равно число N?

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>Амур &amp; Волга</i>	200
<i>Волга &amp; (Амур   Днепр)</i>	350
<i>Амур &amp; Волга &amp; Днепр</i>	80

Какое количество страниц (в тыс.) будет найдено по запросу *Волга & Днепр* ?

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

Ответ: \_\_\_\_\_.

**18** На числовой прямой даны два отрезка: P = [5, 15] и Q = [11, 21].

Каким должен быть отрезок A, чтобы формула

$$((x \in A) \rightarrow \neg(x \in Q)) \vee (x \in P)$$

была тождественно истинной, то есть принимала значение 1 при любом значении переменной x. В ответе запишите наименьшее целое число, которое не может принадлежать отрезку A.

Ответ: \_\_\_\_\_.

**19** В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 6, 3, 5, 4, 2, 8, 9, 1, 5, 7 соответственно, т.е. A[0]=6, A[1]=3 и т.д.

Определите значение переменной s после выполнения следующего фрагмента программы, записанного ниже:

```

Паскаль
s := 0;
n := 9;
z := A[n];
for i := 0 to n-1 do begin
    if A[i]<z then
        s := s + A[i];
end;
    
```

Ответ: \_\_\_\_\_.

**20** Ниже на пяти языках записан алгоритм, который получает на вход число x и печатает два числа a и b. Укажите наибольшее из таких чисел x, при вводе которых алгоритм сначала печатает 3, а потом 5.

```

Паскаль
var x, a, b: integer;
begin
    readln(x);
    a := 0; b := 1;
    while x>0 do
        begin
            a := a + 1;
            b := b*(x mod 10);
            x := x div 10;
        end;
    writeln(a); write(b);
end.
    
```

Ответ: \_\_\_\_\_.

- 21 Ниже записан алгоритм. Определите, какое число будет напечатано в результате выполнения этого алгоритма.

```

Паскаль
var a, b, t, M, R: integer;
function F(x: integer) : integer;
begin
  F := 19*(x-19)*(x-17)+17;
end;
begin
  a := 20; b := 20;
  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(M);
end.

```

Ответ: \_\_\_\_\_.

- 22 Исполнитель Накопитель преобразует число на экране. У исполнителя есть две команды, которым присвоены номера:

1. Прибавь 5
2. Прибавь 10

Первая команда увеличивает число на экране на 5, вторая — увеличивает его на 10. Программа для исполнителя Накопитель — это последовательность команд.

Сколько различных чисел можно получить при исходном числе 1 с помощью программы, которая состоит ровно из 7 команд?

Ответ: \_\_\_\_\_.

- 23 Сколько существует различных наборов значений логических переменных  $x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8$ , которые удовлетворяют всем перечисленным ниже условиям?

$$(x_1 \vee x_2) \rightarrow (x_3 \vee x_4) = 1$$

$$(x_3 \vee x_4) \rightarrow (x_5 \vee x_6) = 1$$

$$(x_5 \vee x_6) \rightarrow (x_7 \vee x_8) = 1$$

В ответе **не нужно** перечислять все различные наборы значений переменных  $x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8$ , при которых выполняется данная система равенств. В качестве ответа укажите 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

На обработку поступает натуральное число, не превышающее  $10^9$ . Нужно написать программу, которая выводит на экран максимальную цифру числа, кратную 3. Если в числе нет цифр, кратных 3, требуется на экран вывести «NO». Программист написал программу неправильно. Ниже приведена эта программа.

**Напоминание:** 0 делится на любое натуральное число.

## Паскаль

```
var N, digit, maxDigit: longint;
begin
  readln(N);
  maxDigit := N mod 10;
  while N > 0 do
  begin
    digit := N mod 10;
    if digit mod 3 = 0 then
      if digit > maxDigit then
        maxDigit := digit;
    N := N div 10;
  end;
  if maxDigit = 0 then
    writeln('NO')
  else
    writeln(maxDigit)
end.
```

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе числа 134.
2. Приведите пример такого трехзначного числа, при вводе которого программа выдает правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько).

Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:

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

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в имеющейся программе, а не написать свою, возможно, использующую другой алгоритм решения.

Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

25

Дан массив, содержащий неотрицательные целые числа. Если сумма всех элементов массива чётная, то нужно вывести количество чётных (по значению) элементов массива, если нечётная – количество нечётных.

Например, для массива из 6 элементов, равных соответственно 2, 6, 12, 17, 3, 8, ответом будет 4 – количество чётных элементов, так как общая сумма всех элементов чётна.

Напишите на одном из языков программирования программу для решения этой задачи. Исходные данные объявлены так, как показано ниже.

Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных.

## Паскаль

```
const
  N=2000;
var
  a: array [1..N] of integer;
  i, 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 **три** раза. Например, имея кучу из 15 камней, за один ход можно получить кучу из 16 или 45 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится не менее 66. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 66 или больше камней. В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 65$ .

Будем говорить, 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.

Выполните следующие задания. Во всех случаях обосновывайте свой ответ.

1. а) Укажите все такие значения числа  $S$ , при которых Петя может выиграть в один ход. Обоснуйте, что найдены все нужные значения  $S$ , и укажите выигрышающий ход для каждого указанного значения  $S$ .

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.

2. Укажите два таких значения  $S$ , при которых у Пети есть выигрышная стратегия, причём (а) Петя не может выиграть за один ход и (б) Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня. Для каждого указанного значения  $S$  опишите выигрышную стратегию Пети.

3. Укажите значение  $S$ , при котором:

— у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети, и

— 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 куче.

27

По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число  $R$ , удовлетворяющее следующим условиям:

1)  $R$  — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных элементов последовательности, равных по величине, допускаются);

2)  $R$  делится на 10.

Если такого числа  $R$  нет, то контрольное значение полагается равным 0. 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, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

*Вычисленное контрольное значение: ...*

*Контроль пройден (или — 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
95
17
10
102
957
95
9690
```

*Пример выходных данных для приведённого выше примера входных данных:*

*Вычисленное контрольное значение: 9690*

*Контроль пройден*



## 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	5
2	110
3	15
4	5
5	5
6	2
7	11
8	1
9	8
10	192
11	4
12	239
13	3
14	16
15	48
16	7
17	230
18	16
19	26
20	511
21	18
22	8
23	121

## Часть 2

## Критерии оценивания заданий с развернутым ответом

24

На обработку поступает натуральное число, не превышающее  $10^9$ . Нужно написать программу, которая выводит на экран максимальную цифру числа, кратную 3. Если в числе нет цифр, кратных 3, требуется на экран вывести «NO». Программист написал программу неправильно.

**Напоминание:** 0 делится на любое натуральное число.

```

Паскаль
var N,digit,maxDigit: longint;
begin
  readln(N);
  maxDigit := N mod 10;
  while N > 0 do
  begin
    digit := N mod 10;
    if digit mod 3 = 0 then
      if digit > maxDigit then
        maxDigit := digit;
    N := N div 10;
  end;
  if maxDigit = 0 then
    writeln('NO')
  else
    writeln(maxDigit)
end.

```

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе числа 134.
2. Приведите пример такого трехзначного числа, при вводе которого программа выдает правильный ответ.
3. Найдите все ошибки в этой программе (их может быть одна или несколько).

Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:

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

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в имеющейся программе,



а не написать свою, возможно, использующую другой алгоритм решения.

Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

1. В *maxDigit* изначально запишется 4. При работе программы будет найдена одна цифра, которая делится на 3 — это 3. Но так как она меньше 4, то ответ обновлён не будет и результатом работы программы будет 4.

2. 333. Изначально в *maxDigit* запишется 3. Чисел, делящихся на 3 и больших, чем 3, нет, поэтому ответ не обновится ни разу за время работы программы и результатом будет 3.

3. Для языка **Pascal**:

#### Первая ошибка.

Неверная инициализация переменной *maxDigit*:

```
maxDigit := N mod 10;
```

Следует проинициализировать её числом, меньшим нуля, например, -1, что будет означать, что пока в числе не найдена цифра, которая делится на 3:

```
maxDigit := -1;
```

#### Вторая ошибка.

0 может быть ответом, поэтому ошибка в строчках:

```
if maxDigit = 0 then
  writeln('NO')
```

И так как мы условились изначально проинициализировать *maxDigit* как -1, то именно это значение и стоит использовать для обнаружения отсутствия ответа:

```
if maxDigit = -1 then
  writeln('NO')
```

25

Дан массив, содержащий неотрицательные целые числа. Если сумма всех элементов массива чётная, нужно вывести количество чётных (по значению) элементов массива, если нечётная — количество нечётных.

Например, для массива из 6 элементов, равных соответственно 2, 6, 12, 17, 3, 8, ответом будет 4 — количество чётных элементов, так как общая сумма всех элементов чётна.

Напишите на одном из языков программирования программу для решения этой задачи. Исходные данные объявлены так, как показано ниже.

Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных.

#### Паскаль

```
const
  N=2000;
var
  a: array [1..N] of integer;
  i, 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.

Решение:

На первый взгляд для решения задачи нужны 4 переменных: счётчик для перебора элементов, счётчики чётных и нечётных элементов, сумма всех элементов, но в условии разрешено использовать только 2 переменные.

Чтобы обойти это ограничение, заметим, что на самом деле достаточно подсчитывать только количество нечётных элементов. Значение суммы можно не вычислять, так как требуется только чётность суммы, а она однозначно определяется количеством нечётных элементов. Количество чётных элементов, если оно потребуется, можно вычислить, зная общее число элементов и количество нечётных.

Эта идея реализована в следующем фрагменте на языке Паскаль.

```
k:=0;
for i:=1 to N do begin
  if a[i] mod 2 = 1 then k:=k+1;
end;
if k mod 2 = 1
then writeln(k)
else writeln(N-k);
```

Возможно также двухпроходное решение: на первом проходе



определяется общая сумма, на втором количество элементов нужной чётности.

Эта идея реализована в следующем фрагменте на алгоритмическом языке.

```

k:=0;
нц для i от 1 до N
k:=k+a[i]
кц
если mod(k,2)=0
то
k:=0;
нц для i от 1 до N
если mod(a[i],2) = 0
то k:=k+1
все
кц
иначе
k:=0;
нц для i от 1 до N
если mod(a[i],2) = 1
то k:=k+1
все
кц
все
Вывод k
begin
j:=0;
for i:=1 to n do
begin
readln(a[i]);
if (a[i]<0) and (j=0) then j:=i;
end;
if j=0 then writeln('no') else writeln(j);
end.

```

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 **три** раза. Например, имея кучу из 15 камней, за один ход можно

получить кучу из 16 или 45 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится не менее 66. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 66 или больше камней. В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 65$ .

Будем говорить, 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.

Выполните следующие задания. Во всех случаях обосновывайте свой ответ.

1. а) Укажите все такие значения числа  $S$ , при которых Петя может выиграть в один ход. Обоснуйте, что найдены все нужные значения  $S$ , и укажите выигрышающий ход для каждого указанного значения  $S$ .

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.

2. Укажите два таких значения  $S$ , при которых у Пети есть выигрышная стратегия, причём (а) Петя не может выиграть за один ход и (б) Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня. Для каждого указанного значения  $S$  опишите выигрышную стратегию Пети.

3. Укажите значение  $S$ , при котором:

— у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети, и

— 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 куче.

1. а) Петя может выиграть, если  $S = 22, \dots, 65$ . Во всех этих случаях достаточно утроить количество камней. При меньших значениях  $S$  за один ход нельзя получить кучу, в которой больше 65 камней.

б) Ваня может выиграть первым ходом (как бы ни играл Петя), если исходно в куче будет  $S = 21$  камней. Тогда после первого хода Петя в куче будет 22 или 63 камня. В обоих случаях Ваня утраивает количество камней и выигрывает в один ход.

2. Возможные значения  $S$ : 7 и 20. В этих случаях Петя, очевидно, не может выиграть первым ходом. Однако он может получить кучу из 21

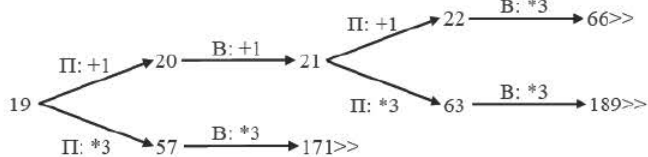


камня: в первом случае утроением, во втором добавлением одного камня. Эта позиция разобрана в п. 16. В ней игрок, который будет ходить (теперь это Ваня), выиграть не может, а его противник (то есть Петя) следующим ходом выигрывает.

3. Возможное значение  $S$ : 19. После первого хода Пети в куче будет 20 или 57 камней. Если в куче станет 57 камней, Ваня утроит количество камней и выигрывает первым ходом. Ситуация, когда в куче 20 камней, уже 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 Ванн. Заключительные позиции (в них выигрывает Ваня) подчеркнуты. На рисунке это же дерево изображено в графическом виде (оба способа изображения дерева допустимы).

Положения после очередных ходов				
И.п.	1-й ход Пети (разобраны все холы)	1-й ход Ванн (только ход по стратегии)	2-й ход Пети (разобраны все холы)	2-й ход Ванн (только ход по стратегии)
19	$19+1=20$	$20+1=21$	$21+1=22$	$22*3=66$
	$19*3=57$	$57*3=171$	$21*3=63$	$63*3=189$



значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

*Вычисленное контрольное значение: ...*

*Контроль пройден (или — 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  
95  
17  
10  
102  
957  
95  
9690

*Пример выходных данных для приведённого выше примера входных данных:*

*Вычисленное контрольное значение: 9690  
Контроль пройден*

Произведение двух чисел делится на 10 если:  
- один из сомножителей делится на 10 (второй может быть любым) либо  
- ни один из сомножителей не делится на 10. причём один из сомножителей делится на 2. а другой — на 5.  
Поэтому программа, вычисляющая кодовое число, может работать

27 По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число  $R$ , удовлетворяющее следующим условиям:

1)  $R$  — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных элементов последовательности, равных по величине, допускаются);

2)  $R$  делится на 10.

Если такого числа  $R$  нет, то контрольное значение полагается равным 0. В результате помех при передаче как сами числа, так и контрольное



так.

Программа читает все входные данные один раз, не запоминая все данные в массиве. Программа для прочитанного фрагмента входной последовательности хранит значения четырёх величин:

M2 — самое большое чётное число, не кратное 5;

M7 — самое большое число, кратное 5, но не кратное 2;

M10 — самое большое число, кратное 10;

MAX — самое большое число среди всех элементов последовательности, отличное от M10 (если число M10 встретилось более одного раза и оно же является максимальным, то MAX = M10).

После того как все данные прочитаны, искомое кодовое слово вычисляется как максимум из произведений M10\*MAX и M2\*M5.

Ниже приведён пример программы на языке Паскаль, которая реализует описанный алгоритм.

Кроме того, приведён пример программы на языке Бейсик, которая правильно решает задачу, но использует алгоритм, немного отличающийся от описанного выше. Возможны и другие правильные алгоритмы. Допускаются решения, записанные на других языках программирования

#### Пример правильной и эффективной программы на языке Паскаль

```
var M2,M5,M10,R,MAX,dat,res,i,N: longint;
begin
  M2 := 0;
  M5 := 0;
  M10 := 0;
  MAX := 0;
  readln(N);
  for i := 1 to N do
  begin
    readln(dat);
    if ((dat mod 2) = 0) and ((dat mod 5) > 0) and (dat > M2) then M2 := dat;
    if ((dat mod 5) = 0) and ((dat mod 2) > 0) and (dat > M5) then M5 := dat;
    if (dat mod 10 = 0) and (dat > M10) then begin
      if M10 > MAX then MAX := M10; M10 := dat
    end
  else
    if dat > MAX then MAX := dat;
  end;
  readln(R);
  if (M2*M5 < M10 *MAX) then
```

```
res := M10*MAX else
res := M2*M5; writeln('Вычисленное контрольное значение: ',res);
if R = res then writeln('Контроль пройден')
else writeln('Контроль не пройден');
end.
```

#### На языке Бейсик

```
M10 = 0
M2 = 0
M5 = 0
MAX = 0
INPUT N
FOR I = 1 TO N
INPUT DAT
IF DAT MOD 2=0 AND DAT > M2 THEN M2 = DAT ELSE
IF DAT MOD 5=0 AND DAT > M5 THEN M5 = DAT
END IF
END IF
IF DAT MOD 10 = 0 AND DAT > M10 THEN IF M10 > MAX THEN
MAX = M10 END IF M10 = DAT ELSE
IF DAT > MAX THEN
MAX = DAT
END IF
END IF
NEXT I
INPUT R
IF M5 * M2 < M10 * MAX THEN
RES = M10 * MAX
ELSE
RES = M5 * M2
END IF
PRINT "Вычисленное контрольное значение:";
RES
IF RES = R
THEN
PRINT "Контроль пройден"
ELSE
PRINT "Контроль не пройден"
END IF
END
```

