

**Первый (заочный) онлайн-этап научно-образовательного соревнования  
Олимпиады школьников «Шаг в будущее» по программированию «Профессор Лебедев»  
(общеобразовательный предмет информатика), осень 2019 г.**

**8 класс**

**Вариант 1**

**Задача 1.**

Вася любит заниматься физкультурой, но не в спортзале, а каждый раз, когда предоставляется естественная возможность напрячься. В том числе он никогда не пользуется лифтом. Поднимаясь до нужного этажа, Вася, по возможности, замечает, какой этаж находится ровно на середине подъёма. Напишите программу, которая по этажу  $k$ , находящемуся ровно на середине подъёма, определит этаж, на который поднимается Вася.

На вход программе подаётся одно натуральное число  $k$  ( $k < 100$ ).

Выведите целое число, равное этажу, на который поднимается Вася .

**Пример**

Входные данные	Выходные данные
2	3

**Тесты**

Входные данные	Выходные данные
1	1
10	19
11	21
98	195
99	197

**Решение**

```
var k: longint;  
Begin  
  read(k);  
  write(1 + 2 * (k - 1))  
End.
```

**Задача 2.**

Андрей, разглядывая семейный фотоальбом, заметил, что бабушка пользовалась то треугольной, то овальной заколкой, даже если платье было одно и то же. На фотографиях были даты, и Андрей подумал, а не связано ли это с днём недели. Календаря под рукой не оказалось, но Андрей легко умел определять день недели, на который приходилось 1 января любого года. Помогите Андрею проверить его гипотезу.

На вход программе подаётся дата, когда была сделана фотография, и номер дня недели 1 января того же года (1 – понедельник, 2 – вторник, ..., 7 – воскресенье). Дата указывается как три натуральных числа, записанные через пробел (число, месяц год).

Программа должны вывести номер дня недели, когда была сделана фотография.

### Пример

Входные данные	Выходные данные
29 02 2016 5	1

Комментарий к примеру: 01.01.2016 выпадало на пятницу, а 29.01.2016 – на понедельник (год был високосным).

### Тесты

Входные данные	Выходные данные
01 01 2010 5	5
31 12 2010 5	5
30 12 2010 5	4
29 12 2010 5	3
28 12 2010 5	2
27 12 2010 5	1
26 12 2010 5	7
26 12 2016 5	1
26 12 2000 6	2
26 12 1900 1	3

### Решение

```
#include <iostream>
using namespace std;
int main()
{
    int monthes[13] = {0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
    int d, m, y, n;
    scanf("%d.%d.%d %d", &d, &m, &y, &n);
    if(y % 4 == 0 && y % 100 != 0 || y % 400 == 0)
        monthes[2] = 29;
    for(d--, m--; m > 0; m--)
        d += monthes[m];
    // d - номер дня После 01.01
    cout << (n - 1 + d) % 7 + 1;
    return 0;
}
```

### Задача 3.

Написать программу, которая находит наибольшее целое число, сумма четных положительных цифр которого кратна 3, в диапазоне от  $M$  до  $N$  включительно.

*Формат ввода*

В строке вводятся два целых числа  $M, N, 0 \leq M \leq N \leq 30000$ .

*Формат вывода*

Вывести одно целое число – наибольшее число, сумма цифр которого кратна 4, из диапазона. Если таких чисел нет, вывести 0.

### Пример

Входные данные	Выходные данные
5 30	24

### Тесты

Входные данные	Выходные данные
5 30	24
7 8	0
10 1000	996
10 1003	996
400 403	402

### Решение

```
program z83z92;

function good(num:integer):boolean;
var sum:integer;
begin
  sum:=0;
  while num>0 do
    begin
      if (num mod 10 > 0) and (num mod 2 = 0)then
        sum:=sum+num mod 10;
        num:=num div 10;
      end;
    good:=(sum mod 3 = 0) and (sum > 0);
  end;

var m,n,i,max:integer;
begin
  readln(m,n);
  max:=0;
  i:=n;
  while (i>=m) and not good(i) do
    begin
      i:=i-1;
    end;
  max:=i;
  writeln(max);
end.
```

#### Задача 4.

Есть исполнитель *Битик*. Он преобразует целое неотрицательное число, которое хранится в одном байте, по следующему правилу:

1) Если число нечётное, то циклически смещаем все двоичные цифры в байте на 2 позиции вправо.

*Например,*

$105_{10}$

$01101001_2$

Преобразуется в

$011010$   $01$  (сдвиг вправо)

$010011010_2$  (результат)

$90_{10}$

2) Если число чётное, то смещаем все двоичные цифры в байте на 3 позиции влево нециклически (записываем на образовавшиеся места 0).

*Например,*

$106_{10}$

$01101010_2$

Преобразуется в

$01010$   $000$  (сдвиг влево)

$01010000_2$  (результат)

$80_{10}$

Напишите программу, которая осуществляет данное преобразование над введённым неотрицательным числом.

На вход программы подаётся одно целое число.

Программа должна вывести одно целое число - вычисленное значение.

#### Примеры

Ввод	Вывод
105	90
106	80

#### Тесты

Ввод	Вывод
47	203
57	78
167	233
158	240
114	144
110	112

#### Решение

```
program p8_1;  
var n,m:integer;  
begin  
  read(n);
```

```

if n mod 2=0 then
  m:=(n*8)mod 256
else
  m:=(n div 4+(n mod 4)*64)mod 256;
  write(m)
end.

```

### Задача 5.

С клавиатуры вводится несколько целых двузначных десятичных чисел. Точное количество этих чисел неизвестно, но может быть очень велико.

Числа можно собирать в цепочку по следующему правилу: у двух соседних чисел должно совпадать произведение цифр. Начинать цепочку можно с любого числа.

Необходимо определить максимальную длину цепочки, которую можно собрать из введённых чисел по этим правилам

#### *Формат ввода*

В строке вводится сначала целое число  $n$  – количество двузначных чисел в цепочке ( $n \leq 1000$ ), затем  $n$  двузначных целых чисел, все числа отделены друг от друга одним или несколькими пробелами.

#### *Формат вывода*

Вывести одно целое число – количество чисел в цепочке.

### Примеры

Входные данные	Выходные данные
3 22 14 32	2
4 41 22 14 36	3

### Тесты

Входные данные	Выходные данные
3 22 14 32	2
4 41 22 14 36	3
7 33 21 12 23 32 16 61	4
8 14 41 22 14 13 16 61 32	4
8 22 14 32 23 61 16 17 71	4

### Решение

```

program z85z94_1;

var n,i,x,len,maxlen,ld,ldn:integer;
begin
  read(n);
  len:=0;
  ld:=-1;
  maxlen:=0;
  for i:=1 to n do

```

```

begin
read(x);
ldn:=(x mod 10)*(x div 10);
if ld<>ldn then
begin
if len>maxlen then
maxlen:=len;
ld:=ldn;
len:=1;
end
else
len:=len+1;
end;
if len>maxlen then
maxlen:=len;
writeln(maxlen);
end.

```

### Ситуационная задача.

Для обогрева салона вездехода «Секач» используется нагревательный элемент, принципиальная схема которого сводится к трём подключённым последовательно блокам сопротивлений. Зная сопротивления каждого из блоков, рассчитать, какой из нагревательных элементов обеспечит наибольшее тепловыделение при фиксированной силе тока в цепи. Все элементы отличаются общим сопротивлением.

#### Формат ввода

В строке вводится  $n$  – количество сборок, подходящих для обогревателя ( $n \leq 10$ ), затем  $n$  троек целых чисел – сопротивление каждого из трёх блоков сборки. Все числа отделены друг от друга одним или несколькими пробелами.

#### Формат вывода

Вывести одно целое число – номер сборки, обеспечивающей наибольшее тепловыделение.

### Пример

Входные данные	Выходные данные
2 10 10 10 20 5 1	1

### Тесты

Входные данные	Выходные данные
2 10 10 10 20 30 40	2
5 10 10 10 20 30 40 5 8 3 50 50 50 90 50 11	5
10 10 10 10 20 30 40 50 10 10 5 8 3 7 6 6 50 60 80 70 70 70 35 35 35 1 1 2 10 8 6	7

## Решение

```
program my;

var i,n,opt,r1,r2,r3:integer; optkol:real;
begin
  read(n);
  opt:=0;
  optkol:=0;
  for i:=1 to n do
    begin
      read(r1);
      read(r2);
      read(r3);
      if r1+r2+r3>optkol then
        begin
          optkol:=r1+r2+r3;
          opt:=i;
        end;
      end;
    writeln(opt);
  end.
```

**Первый (заочный) онлайн-этап научно-образовательного соревнования  
Олимпиады школьников «Шаг в будущее» по программированию «Профессор Лебедев»  
(общеобразовательный предмет информатика), осень 2019 г.**

**8 класс**

**Вариант 2**

**Задача 1.**

Вася Васечкин очень хочет пятёрку в четверти по информатике, но его средний балл не дотягивает до 4,7 (минимальная средняя оценка для выставления пятёрки в четверти в Васиной школе). Какое минимальное количество пятёрок должно быть у Васи к окончанию четверти?

На вход программе подаётся два целых неотрицательных числа  $a$ ,  $b$ , - количества троек и четвёрок в журнале соответственно. Двоек у Васи, к счастью, нет. Общее количество текущих оценок не превосходит 100. Все оценки имеют одинаковый вес.

Программа должна вывести одно целое число – необходимое количество пятёрок.

**Пример**

Входные данные	Выходные данные
0 1	3

**Тесты**

Входные данные	Выходные данные
1 1	8
0 4	10
3 5	29
10 10	80
10 0	57

**Решение**

```
var a, b: longint;
Begin
  read(a, b);
  write((47 * (a + b) - 10 * (3*a + 4*b) + 2) div 3);
End.
```

**Задача 2.**

Андрей, разглядывая семейный фотоальбом, заметил, что бабушка пользовалась то треугольной, то овальной заколкой, даже если платье было одно и тоже. На фотографиях были даты, и Андрей подумал, а не связано ли это с днём недели. Календаря под рукой не оказалось. Помогите Андрею проверить его гипотезу.

На вход программе подаётся дата, когда была сделана фотография. Дата реальная – три натуральных числа, записанные через точку.

Выведите номер дня недели, когда была сделана фотография: 1 – понедельник, 2 – вторник, ..., 7 – воскресенье.

**Пример**

Входные данные	Выходные данные
29.02.2016	1



Комментарий к примеру: 29.01.2016 – на понедельник (год был високосным).

### Тесты

Входные данные	Выходные данные	Баллы
01.01.2010	5	10
31.12.2010	5	10
30.12.2010	4	10
29.12.2010	3	10
28.12.2010	2	10
27.12.2010	1	10
26.12.2010	7	10
26.12.2016	1	10
26.12.2000	2	10
26.12.1900	3	10

### Решение

```
#include <iostream>
using namespace std;
int main()
{
    int monthes[13] = {0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
    int d, m, y;
    scanf("%d.%d.%d", &d, &m, &y);
    // будем идти к 01.01.2019 года, это был вторник
    d--;
    if(y % 4 == 0 && y % 100 != 0 || y % 400 == 0)
        monthes[2] = 29;
    for(m--; m > 0; m--)
        d += monthes[m];
    for(; y > 2019;){
        y--;
        d += y % 4 == 0 && y % 100 != 0 || y % 400 == 0 ? 366 : 365;
    }
    for(; y < 2019; y++)
        d -= y % 4 == 0 && y % 100 != 0 || y % 400 == 0 ? 366 : 365;
    // d - номер дня После 01.01.2019
    cout << ((2 /* 1.1.2019 было вторником */ - 1 + d) % 7 + 7) % 7 + 1;
    return 0;
}
```

### Задача 3.

Написать программу, которая находит наибольшее целое число, произведение положительных четных цифр которого кратно 4, в диапазоне от  $M$  до  $N$  включительно.

*Формат ввода*

В строке вводятся два целых числа  $M, N, 0 \leq M \leq N \leq 30000$ .

*Формат вывода*

Вывести одно целое число – наибольшее целое число, произведение положительных четных цифр которого кратно 4. Если таких чисел нет, вывести 0.

### Пример

Входные данные	Выходные данные
5 30	28

### Тесты

Входные данные	Выходные данные
5 30	28
6 7	0
10 2200	2200
10 1003	998
400 401	401

### Решение

```
program z83z92;
```

```
function good(num:integer):boolean;  
var pr:integer;  
begin  
  pr:=1;  
  while num>0 do  
    begin  
      if (num mod 10 > 0) and (num mod 2 = 0)then  
        pr:=pr*(num mod 10);  
        num:=num div 10;  
      end;  
    good:=(pr mod 4 = 0);  
  end;
```

```
var m,n,i,max:integer;  
begin  
  readln(m,n);  
  max:=0;  
  i:=n;  
  while (i>=m) and not good(i) do  
    begin  
      i:=i-1;  
    end;  
  max:=i;  
  writeln(max);  
end.
```

### Задача 4.

Есть исполнитель *Битик*. Он преобразует целое неотрицательное число, которое хранится в одном байте, по следующему правилу:

1) Если число нечётное, то циклически смещаем все двоичные цифры в байте на 2 позиции влево.

*Например,*

**105<sub>10</sub>**

**01101001<sub>2</sub>**

**Преобразуется в**

01 101001\_\_ (сдвиг влево)  
**10100101**<sub>2</sub> (результат)  
**165**<sub>10</sub>

2) Если число чётное, то смещаем все двоичные цифры в байте на 3 позиции вправо нециклически (записываем на образовавшиеся места 0).

*Например,*

**106**<sub>10</sub>  
**01101010**<sub>2</sub>  
 Преобразуется в  
 \_\_ \_\_ **01101** (сдвиг вправо)  
00001101<sub>2</sub> (результат)  
**13**<sub>10</sub>

Напишите программу, которая осуществляет данное преобразование над введённым неотрицательным числом.

На вход программы подаётся одно целое число.

Программа должна вывести одно целое число - вычисленное значение.

**Примеры**

Ввод	Вывод
105	165
106	13

**Тесты**

Ввод	Вывод
103	157
163	142
57	228
156	19
246	30
110	13

**Решение**

```

program p8_2;
var n,m:integer;
begin
  read(n);
  if n mod 2=0 then
    m:=n div 8
  else
    m:=(n div 64+(n mod 64)*4)mod 256;
  write(m)
end.
```

### Задача 5.

С клавиатуры вводится несколько целых двузначных десятичных чисел. Точное количество этих чисел неизвестно, но может быть очень велико.

Числа можно собирать в цепочку по следующему правилу: у двух соседних чисел должны отличаться первые цифры. Начинать цепочку можно с любого числа.

Необходимо определить максимальную длину цепочки, которую можно собрать из введённых чисел по этим правилам

#### Формат ввода

В строке вводится сначала целое число  $n$  – количество двузначных чисел в цепочке ( $n \leq 1000$ ), затем  $n$  двузначных целых чисел, все числа отделены друг от друга одним или несколькими пробелами.

#### Формат вывода

Вывести одно целое число – количество чисел в цепочке.

#### Примеры

Входные данные	Выходные данные
3 22 14 12	2
4 41 22 14 16	3

#### Тесты

Входные данные	Выходные данные
3 22 14 12	2
4 41 22 14 16	3
7 33 21 22 23 32 16 61	4
8 14 41 22 14 13 16 61 32	4
8 22 24 22 23 21 26 27 21	1

#### Решение

```
program z85z94_1;
```

```
var n,i,x,len,maxlen,ld,ldn:integer;
```

```
begin
```

```
  read(n);
```

```
  len:=0;
```

```
  ld:=-1;
```

```
  maxlen:=0;
```

```
  for i:=1 to n do
```

```
  begin
```

```
    read(x);
```

```
    ldn:=(x div 10);
```

```
    if (ld=ldn) or (ld = -1) then
```

```
    begin
```

```
      if len>maxlen then
```

```
        maxlen:=len;
```

```
      ld:=ldn;
```

```
      len:=1;
```

```
    end
```

```
  else
```

```
  begin
```

```

len:=len+1;
ld:=ldn;
end;
end;
if len>maxlen then
maxlen:=len;
writeln(maxlen);
end.

```

### Ситуационная задача.

Для обогрева салона вездехода «Секач» используется нагревательный элемент, принципиальная схема которого сводится к двум соединённым последовательно блокам сопротивлений и третьему, который подключается параллельно к первым двум. Зная сопротивления каждого из блоков, рассчитать, какой из нагревательных элементов обеспечит наибольшее тепловыделение при фиксированной силе тока в цепи. Все элементы отличаются общим сопротивлением.

#### Формат ввода

В строке вводится  $n$  – количество сборок, подходящих для обогревателя ( $n \leq 10$ ), затем  $n$  троек целых чисел – сопротивление каждого из трех блоков сборки. Все числа отделены друг от друга одним или несколькими пробелами.

#### Формат вывода

Вывести одно целое число – номер сборки, обеспечивающей наибольшее тепловыделение.

### Пример

Входные данные	Выходные данные
2 10 10 10 20 5 1	1

### Тесты

Входные данные	Выходные данные
2 10 10 10 20 5 1	1
5 10 10 10 20 30 40 5 8 3 50 50 50 90 50 11	4
10 10 10 10 20 30 40 50 10 10 5 8 3 7 6 6 50 60 80 70 70 35 35 35 1 1 2 10 8 6	7

### Решение

```

program my;

var i,n,opt,r1,r2,r3:integer; optkol:real;
begin
read(n);
opt:=0;
optkol:=0;
for i:=1 to n do
begin
read(r1);
read(r2);
read(r3);
if (r3*(r1+r2))/(r1+r2+r3)>optkol then

```

```
begin  
  optkol:=(r3*(r1+r2))/(r1+r2+r3);  
  opt:=i;  
end;  
end;  
writeln(opt);  
end.
```