

**Первый (заочный) онлайн-этап научно-образовательного соревнования  
Олимпиады школьников «Шаг в будущее» по профилю «Инженерное дело» специализации  
«Техника и технологии» (общеобразовательный предмет информатика), осень 2019 г.  
8 класс**

**Вариант 1**

**Задача 1.**

Согласно действующих с 2011 года СНиП 21-02-99 параметры стоянки для легкового автомобиля должны быть такими: ширина 2,5 метра; длина 5,3 метра. Какое максимальное количество парковочных мест может быть выделено вдоль дороги известной длины?

На вход программе подаётся два вещественных числа - длина дороги в метрах, не превышающее тысячи, и длина парковочного места, не превышающее десяти. Каждое из чисел с точностью одна цифра в дробной части.

Программа должна вывести одно целое число – максимальное число возможных парковочных мест.

**Пример**

Входные данные	Выходные данные
81.0 5.4	15

**Тесты**

Входные данные	Выходные данные
68.9 5.3	13
162.0 5.4	30
297.0 5.4	55
36.4 5.2	7
317.2 5.2	61
317.1 5.2	60
2 1	2

**Решение**

```
n, s = map(float, input().split())
print(int(n / s + 0.00001))
```

**Задача 2.**

Светофор работает в следующем режиме: красный-жёлтый-зелёный-жёлтый-красный-жёлтый и т.д. Лампочки в светофоре не вечные и иногда перегорают. Известно, что красная лампочка перегорает через **R** включений, жёлтая – через **Y** включений, а зелёная – через **G** включений. Необходимо выяснить сколько раз успеет переключиться светофор до тех пор, пока не перегорит первая лампочка.

Напишите программу, которая по заданным **R, Y, G** определяет количество переключений светофора. Считать, что при первом переключении загорается красный свет.

Формат ввода

В строке вводится три целых неотрицательных числа **R, Y, G** (Сумма трёх данных чисел не превосходит  $10^9$ ). Все числа отделены друг от друга одним или несколькими пробелами.

Формат вывода

Вывести одно целое число – количество переключений светофора до перегорания какой-либо лампочки.

### Пример

Входные данные	Выходные данные
3 3 3	7

Пояснение. Светофор работает по следующей схеме: красный-жёлтый-зелёный- жёлтый-красный-жёлтый-зелёный. При следующем переключении жёлтая лампочка сгорит. Таким образом, до сгорания лампочки светофор успел переключиться 7 раз.

### Тесты

Входные данные	Выходные данные
3 3 3	7
100 100 100	201
2 1 0	2
0 1 2	0
5 0 5	1
142 213 98	394

### Решение

**var**

a, b, c, a1, b1, k: **integer**;

**begin**

readln(a, b, c);

write(a, ' ', b, ' ', c, ' ');

k := min(min(a, b **div** 2), c);

a -= k;

b -= 2 \* k;

c -= k;

**if** a = 0 **then**

writeln(4 \* k)

**else if** b = 0 **then**

writeln(4 \* k + 1)

**else if** c = 0 **then**

writeln(4 \* k + 2)

**else**

writeln(4 \* k + 3);

**end.**

### Задача 3.

Написать программу, которая находит наименьшее целое число, сумма нечетных цифр которого кратна 5, 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 – наименьшее целое число, сумма нечетных цифр которого кратна 5, из диапазона. Если таких чисел нет, вывести 0.

### Пример

Входные данные	Выходные данные
4 30	5

### Тесты

Входные данные	Выходные данные
4 30	5
7 9	0
10 1000	19
5 1000	5
400 401	400

### Решение

```
program z83z92;

function good(num:integer):boolean;
var sum:integer;
begin
  sum:=0;
  while num>0 do
    begin
      if (num mod 2 = 1)then
        sum:=sum+num mod 10;
        num:=num div 10;
      end;
    good:=(sum mod 5 = 0) and (sum > 0);
  end;

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

### Задача 4.

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

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

Например,

$106_{10}$   
 $01101010_2$

Преобразуется в  
 \_\_011010 10 (сдвиг вправо)  
10011010<sub>2</sub> (результат)  
 154<sub>10</sub>

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

*Например,*

105<sub>10</sub>  
 01101001<sub>2</sub>  
 Преобразуется в  
 01001\_\_\_ (сдвиг влево)  
 01001000<sub>2</sub> (результат)  
 72<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	72
106	154

### Тесты

Ввод	Вывод
158	167
114	156
116	29
55	184
157	232
107	88

### Решение

```

program p8_3;
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 должна совпадать сумма цифр. Начинать цепочку можно с любого числа.

Необходимо определить максимальную длину цепочки, которую можно собрать из введенных 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 13 32	2
4 41 23 14 36	3

### Тесты

Входные данные	Выходные данные
3 22 13 32	2
4 41 23 14 36	3
7 33 21 12 23 32 14 41	4
8 14 41 23 14 13 16 61 34	4
8 22 13 31 23 62 26 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.
```

**Первый (заочный) онлайн-этап научно-образовательного соревнования  
Олимпиады школьников «Шаг в будущее» по профилю «Инженерное дело» специализации  
«Техника и технологии» (общеобразовательный предмет информатика), осень 2019 г.  
8 класс**

**Вариант 2**

**Задача 1.**

Ваня хочет подарить бабушке панно из собранных собственноручно пазлов. Но так, чтобы оно поместилось на ломберном столике размера  $W \times D$ , что стоит у бабушки в гостиной, и содержало максимальное количество пазлов. Сколько пазлов можно отобрать для панно, если один пазл в собранном состоянии имеет размер  $w \times d$ ?

На вход программе подаётся четыре натуральных числа  $W$ ,  $D$ ,  $w$  и  $d$ , каждое которых не превышает 1000.

Программа должна вывести одно число – ответ задачи.

**Пример**

Входные данные	Выходные данные
60 60 20 15	12

**Тесты**

Входные данные	Выходные данные
15 10 15 10	1
15 10 10 15	1
1 1 10 10	0
10 10 1 1	100
100 10 30 3	9
100 10 3 30	9
100 200 24 26	28
100 200 26 24	28
200 100 24 26	28
200 100 26 24	28

**Решение**

```
int main()
{
    int W, D, w, d;
    cin >> W >> D >> w >> d;
    int n1 = W / w * (D / d);
    int n2 = D / w * (W / d);
    cout << max(n1, n2);
}
```

## Задача 2.

Ребята, занимающиеся в кружке робототехники, создали робота-сортировщика. Он раскладывает некоторые детали по трём коробочкам. Алгоритм его работы следующий: кладёт первую деталь в коробку с номером 1, следующую – в коробку с номером 2, затем – в коробку с номером 3, затем в коробку с номером 2, затем – в коробку с номером 1, затем – в коробку с номером 2 и т.д. Если коробка, в которую робот должен положить очередную деталь, больше не может вместить в себя ни одной детали, робот останавливается. Коробки могут быть разных размеров.

Зная, сколько деталей вмещает каждая из коробок, необходимо определить сколько деталей успеет разложить робот до своей остановки

Напишите программу, которая по заданным **F, S, T** (вместимость, соответственно коробки с номером 1, коробки с номером 2 и коробки с номером 3) определяет количество деталей, которые успеет разложить робот до своей остановки.

Формат ввода

В строке вводится три целых неотрицательных числа **F, S, T** (Сумма трёх данных чисел не превосходит  $10^9$ ). Все числа отделены друг от друга одним или несколькими пробелами.

Формат вывода

Вывести одно целое число – количество деталей, которые успеет разложить робот до своей остановки.

### Пример

Входные данные	Выходные данные
5 2 7	5

Пояснение. Робот разложит детали в следующие коробки: 1, 2, 3, 2, 1. Далее робот должен положить деталь в коробку с номером 2, но она заполнена. Робот остановится. Таким образом, до своей остановки робот успеет разложить 5 деталей.

### Тесты

Входные данные	Выходные данные
5 2 7	5
100 100 100	201
2 1 0	2
0 1 2	0
5 0 5	1
142 213 98	394

## Решение

**var**

a, b, c, a1, b1, k: **integer**;

**begin**

readln(a, b, c);

write(a, ' ', b, ' ', c, ' ');

k := min(min(a, b div 2), c);

a -= k;

b -= 2 \* k;

c -= k;

**if** a = 0 **then**

writeln(4 \* k)

```

else if b = 0 then
  writeln(4 * k + 1)
else if c = 0 then
  writeln(4 * k + 2)
else
  writeln(4 * k + 3);
end.

```

### Задача 3.

Написать программу, которая находит наименьшее целое число, произведение нечетных цифр которого кратно 9, 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 – наименьшее целое число, произведение нечетных цифр которого кратно 9, из диапазона. Если таких чисел нет, вывести 0.

### Пример

Входные данные	Выходные данные
5 30	9

### Тесты

Входные данные	Выходные данные
5 30	9
10 13	0
10 1000	19
19 1003	19
1900 1901	1900

### Решение

```

program z83z92;

```

```

function good(num:integer):boolean;

```

```

var pr:integer;

```

```

begin

```

```

  pr:=1;

```

```

  while num>0 do

```

```

    begin

```

```

      if (num mod 2 = 1)then

```

```

        pr:=pr*(num mod 10);

```

```

        num:=num div 10;

```

```

      end;

```

```

    good:=(pr mod 9 = 0);

```

```

  end;

```

```

var m,n,i,min:integer;

```

```

begin

```

```

  readln(m,n);

```

```

  min:=0;

```

```

  i:=m;

```

```

while (i<=n) and not good(i) do
begin
i:=i+1;
end;
min:=i;
writeln(min);
end.

```

#### Задача 4.

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

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

*Например,*

**106**<sub>10</sub>

**01101010**<sub>2</sub>

Преобразуется в

**01 101010**\_\_ (сдвиг влево)

**10101001**<sub>2</sub> (результат)

**169**<sub>10</sub>

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

*Например,*

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

**01101001**<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	13
106	169

#### Тесты

Ввод	Вывод
180	210
118	217
92	113
211	26
49	6
107	13

### Решение

```
program p8_4;  
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 последние 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 24	2
4 41 22 13 23	3

### Тесты

Входные данные	Выходные данные
3 22 14 24	2
4 41 22 13 23	3
7 33 21 22 32 16 61 43	4
8 14 41 22 14 24 44 61 32	4
8 42 52 62 32 92 82 72 12	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 mod 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.
```