

**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,

106₁₀

01101010₂

Преобразуется в

011010 10 (сдвиг вправо)

10011010₂ (результат)

154₁₀

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

Например,

105_{10}

01101001_2

Преобразуется в

$01001_ _ _$ (сдвиг влево)

01001000_2 (результат)

72_{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	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 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 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.
```