

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

Вариант 2

Задача 1.

Петя скоро пойдёт в школу, и он влюблён в числа. Недавно ему рассказали, как считать после девяти. Особенно Пете нравились числа, у которых нечётная сумма цифр. Он принялся их считать.

Напишите программу, которая проверит Петины подсчёты.

На вход программы подаётся два числа A и B , записанные через пробел ($1 \leq A \leq B \leq 10^9$).

Программа должна вывести количество чисел от A до B (включая сами числа A и B), которые так понравились Пете.

Пример

Входные данные	Выходные данные
20 30	6

Комментарий к примеру:

от 20 до 30 есть 6 искомым чисел: 21, 23, 25, 27, 29 и 30.

Решение

```
function nice(n: longint): boolean;
var s: longint;
begin
  s := 0;
  while n > 0 do begin
    inc(s, n mod 10);
    n := n div 10;
  end;
  nice := s mod 2 = 0;
end;

function count(a, b: longint): longint;
var res, i: longint;
begin
  res := 0;
  for i := a to b do
    if nice(i) then
      inc(res);
  count := res;
end;
```

```

var a, b, k, a1, b1: longint;
Begin
  read(a, b);
  k := 0;
  if b >= a then begin
    a1 := min(b, (a div 10 + 1) * 10 - 1); // конец текущего десятка A - aaa9
    inc(k, count(a, a1));
    a := a1 + 1;
    if b >= a then begin
      b1 := max(a, b div 10 * 10); // последний десяток bbb0 - xxxb
      inc(k, count(b1, b));
      b := b1 - 1;
      if b >= a then
        inc(k, 5 * (b + 1 - a) div 10);
    end;
  end;
end;

write(k)
End.

```

Тесты

N	Оценка	Входные данные	Выходные данные
1	1	99 99	0
2	2	99 300	101
3	2	999 3000	1002
4	2	999 1001	2
5	3	15 999999985	70503255

Задача 2.

Будем считать подходящим десятичное число, двоичная запись которого заканчивается не менее, чем X нулями. Сколько таких чисел существует в диапазоне от M до N включительно?

Формат ввода

В строке вводятся три натуральных числа M , N , X ,

$M \leq N \leq 4000000$,

$X \leq 20$

Формат вывода

Вывести одно целое число – количество целых чисел, удовлетворяющих условию. Если таких чисел нет, вывести 0.

Пример

Входные данные	Выходные данные
2 30 3	3

Решение

```
program my;
const
  O = 2;

function intpower(x:longint):longint;
var i,c:integer;
begin
  c:=1;
  for i:=1 to x do
    c:=c*O;
  intpower:=c;
end;

var i,s,c,m,n,x:longint;
begin
  read(m,n,x);
  s:=0;
  c:=intpower(x);
  for i:=m to n do
    if i mod c = 0 then
      s:=s+1;
  writeln(s);
end.
```

Тесты

N	Оценка	Входные данные	Выходные данные
1	3	60 60 2	1
2	3	30 63 6	0
3	3	2 28 1	14
4	3	100 500 6	6
5	3	1 4000000 8	15625

Задача 3.

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

1) Если количество единиц в двоичной записи этого числа нечётное, то смещаем все двоичные цифры в байте на 2 позиции вправо и записываем на образовавшиеся место два бита со значением *10*.

Например,

19_{10}

00010011_2

Преобразуется в

$__000100$ (сдвиг вправо)

10000100_2 (результат)

132_{10}

2) Если количество единиц в двоичной записи этого числа чётное, то смещаем все двоичные цифры в байте на 2 позиции влево и записываем на образовавшиеся место два бита со значением 10.

Например,

12_{10}

00001100_2

Преобразуется в

$001100_ _$ (сдвиг влево)

00110010_2 (результат)

50_{10}

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

На вход программы подаётся число N .

Программа должна вывести одно целое число - вычисленное значение.

Пример

Ввод	Вывод
19	132
12	50

Решение

```
function f(n:integer):integer;
var s,x:integer;
begin
x:=n;
s:=0;
while x>0 do
begin
s:=s+ x mod 2;
x:= x div 2;
end;
if s mod 2 =0 then
f:=2+(n*4) mod 256
else
f:=128+(n div 4)
end;

var n:integer;
begin
readln(n) ;
writeln(f(n))
end.
```

N	Оценка	Входные данные	Выходные данные
1	1	0	2
2	1	1	128
3	1	2	128
4	1	22	133
5	1	39	158
6	2	79	147
7	2	102	154
8	2	128	160
9	2	192	2
10	2	255	254

Задача 4.

Суперагентам Матильде и Виттеру необходимо получить кодовый ключ к замку. Для того, чтобы составить ключ необходимо записать в одну строку без пробелов все слова максимально возможной длины, составленные из букв некоторого алфавита, в которых каждая буква повторяется не более N раз ($N \leq 5$). Слова можно записать в любом порядке.

В качестве кода требуется набрать два числа: длину строки и количество слов в строке.

Входные данные:

На вход программе подаётся строка, содержащая алфавит, из которого надо составлять слова, в виде последовательности символов, записанных без пробела и число N , записанное через пробел после алфавита. В качестве буквы алфавита может быть использован любой символ. Количество букв в алфавите не превышает 5.

Выходные данные:

Два числа, записанные через один пробел. Первое число - количество слов, второе - длина строки.

Пример:

Входные данные	Выходные данные
<i>AB 2</i>	<i>6 24</i>

Пояснение:

Получится составить следующие 6 слов максимальной длины: ***AABB ABAB ABBA BAAB BABA BBA A***

Решение

```
type ansistring = string;
function count_letter(const symbol, wrd: ansistring): longint;
var i, amount: longint;
```

```

begin
  amount := 0;
  for i := 1 to length(wrd) do
    if copy(wrd, i, 1) = symbol then
      amount := amount + 1;
    count_letter := amount;
  end;
procedure count_and_build_words(N: longint; const alphabet: ansistring; const wrd: ansistring; var
amount, all_length: longint);
var i: longint;
    is_end: boolean;
begin
  is_end := true;
  for i := 1 to length(alphabet) do
    if count_letter(copy(alphabet, i, 1), wrd) < N then begin
      is_end := false;
      count_and_build_words(n, alphabet, wrd + copy(alphabet, i, 1), amount, all_length);
    end;
  if is_end then begin
    amount := amount + 1;
    all_length := all_length + length(wrd);
  end
end;
function fact(f: int64): int64;
var i, p: int64;
begin
  p := 1;
  for i := f downto 1 do
    p := p * i;
  fact := p;
end;
procedure count_and_build_words_fast(n: longint; const alphabet: ansistring; const wrd: ansistring; var
amount2, all_length2: longint);
var wrd_len, i: longint;
    is_end: boolean;
    amount, all_length: int64;
begin
  wrd_len := length(alphabet) * n;
  amount := 1;
  all_length := wrd_len;
  for i := 1 to length(alphabet) do begin
    amount := amount * fact(all_length) div fact(all_length - n) div fact(n);
    all_length := all_length - n;
  end;
  all_length := amount * wrd_len;
  amount2 := amount;
  all_length2 := all_length;
end;
var
  alphabet: ansistring;
  amount, all_length, n: longint;
begin

```

```

read(alphabet);
readln(n);
amount := 0;
all_length := 0;
count_and_build_words_fast(n, alphabet, "", amount, all_length);
writeln(amount, ' ', all_length);
end.

```

Тесты

N	Оценка	Входные данные	Выходные данные
1	3	AB 5	252 2520
2	5	ABC 2	90 540
3	5	ABCD 2	2520 20160
4	7	ABC 5	756756 11351340
5	10	ABCD 4	63063000 1009008000