

Московский государственный технический университет  
имени Н.Э.Баумана

Олимпиада школьников «Шаг в будущее»

Инженерное дело «Профессор Лебедев» ИНФОРМАТИКА 2 тур

2018-2019 учебный год

9 класс

Вариант 3

**Задача 1.**

Андрей обожает рельсовый транспорт. В кармане он всегда носит свой любимый трамвай *Татра Т3*, игрушечный, конечно. В очередной раз он упросил маму не ехать на машине, а ехать на метро. В метро Андрей попросил маму подождать, пока он не научится предсказывать появление поезда. На часах было ровно  $h_0$  часов и  $m_0$  минут, когда показался первый поезд. Андрей наблюдал за прибытием поездов и, когда на часах было  $h_1$  часов и  $m_1$  минут, показался очередной поезд. Какой по счёту это было поезд, если интервал движения поездов составляет  $dm$  минут,  $ds$  секунд.

На вход программы подаются шесть чисел:  $h_0$ ,  $m_0$ ,  $h_1$ ,  $m_1$ ,  $dm$  ( $dm > 0$ ) и  $ds$ , записанные через один пробел. Момент времени  $h_1$ ,  $m_1$  не раньше момента времени  $h_0$ ,  $m_0$ , но в тех же сутках.

Программа должна вывести через одно число – ответ задачи.

**Пример**

Входные данные	Выходные данные
18 00 18 7 2 30	4

*Комментарий к примеру:*

1-ый поезд – 18:00:00, 2-ой – 18:02:30, 3-ий – 18:05:00, 4-ый – **18:07:30**.

**Решение**

```
var h0, m0, h1, m1, dm, ds, t, n: longint;
```

```
Begin
```

```
  read(h0, m0, h1, m1, dm, ds);
```

```
  t := (h1 * 60 + m1) * 60 + 59 - (h0 * 60 + m0) * 60;
```

```
  n := t div (dm * 60 + ds);
```

```
  write(n + 1);
```

```
End.
```

## Тесты

N	Оценка	Входные данные	Выходные данные
1	1	10 50 11 00 3 20	4
2	1	0 0 1 25 1 7	78
3	2	0 0 1 1 1 0	62
4	1	10 00 11 02 01 01	62

### Задача 2.

Петя скоро пойдёт в школу, и он влюблён в числа. Недавно ему рассказали, как считать после девяти. Особенно Пете нравились числа, которые заканчивались на две одинаковые цифры. 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
190 210	2

*Комментарий к примеру:*

от 190 до 210 включительно есть 2 искомым числа: 199 и 200.

### 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;
    write(k)
End.

```

### Тесты

N	Оценка	Входные данные	Выходные данные
1	2	222 999	78
2	2	222 998	77
3	2	223 999	77
4	1	1 10	0
5	3	10 999999000	99999900

### Задача 3.

Будем считать подходящим десятичное число, шестнадцатеричная запись которого содержит не менее  $X$  значащих цифр. Сколько таких чисел существует в диапазоне от  $M$  до  $N$  включительно?

*Формат ввода*

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

$M \leq N \leq 4000000$ ,

$X \leq 6$

*Формат вывода*

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

### Пример

Входные данные	Выходные данные
5 30 2	15

### Решение

```

program my;
const
    O = 16;

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-1);
    for i:=m to n do
        if i>=c then
            s:=s+1;
    writeln(s);
end.
    
```

### Тесты

N	Оценка	Входные данные	Выходные данные
1	3	30 60 2	21
2	3	256 256 1	1
3	3	20 40 1	21
4	3	10 50000 3	49745
5	3	1 4000000 8	2951425

#### Задача 4.

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

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*. Иначе число не изменяется. Например,

$12_{10}$

$00001100_2$

Преобразуется в

$001100\_ \_$  (сдвиг влево)

$00110010_2$  (результат)

$50_{10}$

Напишите программу, которая вычисляет наибольшее число, которое можно получить при преобразовании чисел из интервала  $[A; B]$ .

На вход программы подаётся два целых числа  $A$  и  $B$ , записанные через пробел.

Программа должна вывести одно целое число - вычисленное значение.

#### Пример

Ввод	Вывод
11 13	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:=n
else
f:=2+ (n * 4)mod 256
end;

var a,b,i,x,max,imax:integer;
begin
readln(a,b) ;
for i:=a to b do
begin
```

```

x:=f(i);
if (i=a)or (max<x) then begin
  max:=x;
  imax:=i
  end;
end;
writeln(max);
end.

```

### Тесты

N	Оценка	Входные данные	Выходные данные
1	3	87 88	88
2	3	20 26	98
3	3	31 37	146
4	3	165 166	154
5	3	60 70	254

### Задача 5.

Прототип манипулятора исполняет четыре команды: «вперед», «назад», «опустить захват» и «поднять захват». Поскольку это робот-прототип, он не может двигаться с опущенным захватом; перед движением требуется поднять захват. Команда «опустить захват» при опущенном захвате и «поднять захват» при поднятом захвате не может быть выполнена. Сколько разных выполнимых последовательностей можно составить из заранее известного количества команд, если в начальный момент времени робот стоит с поднятым захватом?

*Формат ввода*

В строке вводится сначала натуральное число  $n$  – количество команд ( $n \leq 15$ ).

*Формат вывода*

Вывести одно целое число – количество способов, которыми можно составить последовательности команд длиной  $n$  для робота.

### Пример

Входные данные	Выходные данные
2	7

## Решение

```
program my;
var n,i,a,b,c,d,pa,pb,pc,pd:longint;
begin
  pa:=0;
  pb:=0;
  pc:=0;
  pd:=1;
  readln(n);
  for i:=1 to n do
  begin
    a:=pa+pb+pd;
    b:=pa+pd+pb;
    c:=pa+pd+pb;
    d:=pc;
    pa:=a;
    pb:=b;
    pc:=c;
    pd:=d;
  end;
  writeln(a+b+c+d);
end.
```

## Тесты

N	Оценка	Входные данные	Выходные данные
1	5	<i>15</i>	665857
2	5	<i>3</i>	17
3	5	<i>1</i>	3
4	5	<i>10</i>	8119
5	5	<i>8</i>	1393

## Задача 6.

Суперагентам Буке и Бяке необходимо получить кодовый ключ к замку. Для того, чтобы составить ключ необходимо записать в одну строку без пробелов все слова, составленные из букв некоторого алфавита, буквы в них идут в алфавитном порядке, но повторяться могут от 1 до N раз. 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 ( $N \leq 10$ ), записанное через пробел после алфавита. В качестве буквы алфавита может быть использован любой символ. Количество букв в алфавите не превышает 10.

### **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>4 12</i>

### **Пояснение:**

Получится составить следующие 6 слов максимальной длины:

***AB ABV AAB AABV***

### **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;
```

```
function doubl(s: string; n: longint): string;
```

```
var res: string;
```

```
    i: longint;
```

```
begin
```

```
    res:= "";
```

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

```
        res := res + s;
```

```
    doubl := res;
```

```
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
```

```
    if alphabet = " then begin
```



```

    amount := amount + 1;
    all_length := all_length + length(wrd);
end
else
for i := 1 to n do
    count_and_build_words(n, copy(alphabet, 2, length(alphabet) - 1), wrd +
dubl(copy(alphabet, 1, 1),i), amount, all_length);
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
    amount2 := round(power(n, length(alphabet)));
    all_length2 := amount2 * (length(alphabet) + trunc((length(alphabet) / 2) * (n-1)));
end;

```

```

var
    alphabet: ansistring;
    amount, all_length, n: longint;
begin
    read(alphabet);
    readln(n);
    amount := 0;
    all_length := 0;
    count_and_build_words(n, alphabet, "", amount, all_length);
    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	Abc 5	125 1125
2	5	Abcde 1	1 5
3	5	Abcd 8	4096 73728

4	7	Abcdefg 10	10000000 385000000
5	10	Abcdefghij 6	60466176 2116316160

**Задача 7 (Ситуационная задача. Оценивается max = 30 баллов).**

Для обогрева салона вездехода «Секач» используется нагревательный элемент, принципиальная схема которого сводится к трем подключенным последовательно блокам сопротивлений. Зная сопротивления каждого из блоков, рассчитать, какой из нагревательных элементов обеспечит наибольшее тепловыделение при фиксированном напряжении 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
4 10 20 10 5 8 3 20 10 20 15 15 15	2

**Решение**

```

program my;

var i,n,opt,r1,r2,r3:integer; optkol:real;
begin
  read(n);
  opt:=0;
  optkol:=maxint;
  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.

```

## Тесты

N	Оценка	Входные данные	Выходные данные
1	10	2 10 10 10 20 30 40	1
2	10	5 10 10 10 20 30 40 5 8 3 50 50 50 90 50 11	3
3	10	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	9