

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 1

**Задача 1**

Вася знал номер квартиры  $n$ , где живёт его одноклассник, но совершенно не помнил этаж. На табличке над входом в подъезд было написано, что первая квартира имеет номер  $n_1$ . Войдя в подъезд, мальчик увидел, что на этаже расположено  $k$  квартир. Напишите программу, которая поможет Васе определить этаж, на котором живёт его одноклассник, исходя из имеющейся информации.

На вход программы подаются три числа  $n$ ,  $n_1$  и  $k$ , записанные через один пробел, каждое из которых не превышает тысячи.

Программа должна вывести одно число – номер этажа.

**Пример**

Входные данные	Выходные данные
140 131 4	3

**Решение**

```
var n, n1, k: longint;  
Begin  
  readln(n, n1, k);  
  writeln((n - n1) div k + 1)
```

**End.**

N	Оценка	Входные данные	Выходные данные
1	1	123 12 8	14
2	1	124 12 8	15
3	2	1000 1000 1000	1
4	1	5 1 3	2

## Задача 2.

Петя скоро пойдёт в школу, и он влюблён в числа. Недавно ему рассказали, как считать после девяноста девяти. Особенно Пете нравились числа, которые начинались и заканчивались на одну и ту же цифру. 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 Пете особенно понравятся два числа: 191 и 202.

### Решение

```
function senior(n: longint): longint;
```

```
begin
```

```
  while n > 9 do
```

```
    n := n div 10;
```

```
  senior := n;
```

```
end;
```

```
var a, b, i, k: longint;
```

```
Begin
```

```
  read(a, b);
```

```
  k := 0;
```

```
  if a < 10 then begin
```

```
    inc(k, 10 - a);
```

```
    a := 10;
```

```
  end;
```

```
  if a mod 10 > senior(a) then
```

```
    dec(k);
```

```
  a := a div 10 * 10;
```

```
  if b mod 10 < senior(b) then
```

```
    dec(k);
```

```
  b := b div 10 * 10 + 10;
```

```
  inc(k, (b - a) div 10);
```

```
  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, восьмеричная 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  $M$ ,  $N$ ,  $X$ ,

$M \leq N \leq 4000000$ ,

$X \leq 9$

*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	23

### Решение

```
program my;
const
  O = 8;

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 3	0
2	3	64 64 3	1
3	3	20 40 1	21
4	3	10 50000 6	17233
5	3	1 4000000 8	1902849

### 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 нечётное, то смещаем все двоичные цифры в байте на 2 позиции вправо и записываем на образовавшиеся место два бита со значением *01*. Иначе число не изменяется. Например,

$19_{10}$

$00010011_2$

Преобразуется в

$000100$  (сдвиг вправо)

$01000100_2$  (результат)

$68_{10}$

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
18 20	68

```
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:=64+ n div 4
  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	1 10	66
2	3	1 63	79
3	3	64 72	81
4	3	145 146	100
5	3	240 254	252

### Задача 5.

Прототип робота-уборщика исполняет четыре команды: «вперед», «стоп», «поворот налево» и «поворот направо». Поскольку это робот-прототип, он не может поворачивать на ходу; перед поворотом надо отдать команду «стоп». Сколько разных выполнимых последовательностей можно составить из заранее известного количества команд, если в начальный момент времени робот стоит?

*Формат ввода*

В строке вводится сначала натуральное число  $n$  – количество команд ( $n \leq 30$ ).

*Формат вывода*

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

### Пример

Входные данные	Выходные данные
2	14

### Решение

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

## Тесты

N	Оценка	Входные данные	Выходные данные
1	5	30	1818198016
2	5	3	48
3	5	1	4
4	5	10	259808
5	5	8	22288

### Задача 6.

Суперагентам Маше и Вите 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 ( $N$  – порядковый номер этой буквы в алфавите). 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 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
<b><i>AB</i></b>	<b><i>3 9</i></b>

#### *Пояснение:*

В приведённом алфавите буква ***A*** стоит на 1 месте, а буква ***B*** - на втором. Соответственно буква ***A*** может присутствовать в слове не более 1 раза, а буква ***B*** – не более 2 раз. Тогда получится составить следующие 3 слова максимальной длины: ***ABB VAB VBA***

## 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;  
end;
```

```
procedure count_and_build_words(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) < i then begin  
            is_end := false;  
            count_and_build_words(alphabet, wrd + copy(alphabet, i, 1), amount, all_length);  
        end;  
        if is_end then begin  
            //writeln(wrd);  
            amount := amount + 1;  
            all_length := all_length + length(wrd);  
        end  
    end;  
end;
```

```
function fact(f: longint): longint;  
var i, p: longint;  
begin  
    p := 1;  
    for i := f downto 1 do  
        p := p * i;  
    fact := p;  
end;
```

```
procedure count_and_build_words_fast(const alphabet: ansistring; const wrd: ansistring; var  
amount, all_length: longint);  
var wrd_len, i: longint;  
    is_end: boolean;  
begin  
    wrd_len := 0;  
    for i := 1 to length(alphabet) do  
        wrd_len := wrd_len + i;  
  
        amount := 1;  
        all_length := wrd_len;
```



```

for i := 1 to length(alphabet) do begin
    amount := amount * fact(all_length) div fact(all_length - i) div fact(i);
    all_length := all_length - i;
end;

all_length := amount * wrd_len;
end;

var
    alphabet: ansistring;
    amount, all_length: longint;
begin
    readln(alphabet);

    { amount := 0;
    all_length :=0;
    count_and_build_words(alphabet, ", amount, all_length);
    writeln(amount, ' ', all_length);}

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

```

### Тесты

N	Оценка	Входные данные	Выходные данные
1	3	BA	3 9
2	5	ABC	60 360
3	5	CBA	60 360
4	7	ABDC	259808
5	10	ABCDE	22288

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

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

### 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.
```

### Тесты

N	Оценка	Входные данные	Выходные данные
1	10	2 10 10 10 20 30 40	2
2	10	5 10 10 10 20 30 40 5 8 3 50 50 50 90 50 11	5
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	7