

# ИТЕРАЦИОННЫЕ АЛГОРИТМИЧЕСКИЕ СТРУКТУРЫ

Лекция №2

# ЗАДАНИЕ

- 1) Определите количество принадлежащих отрезку  $[2 \cdot 10^{10}; 4 \cdot 10^{10}]$  натуральных чисел, которые делятся на 7 и на 100 000 и при этом не делятся на 13, 29, 43 и 101, а также наименьшее из таких чисел. В ответе запишите два целых числа: сначала количество, затем наименьшее число.
- 2) Напишите программу, которая ищет среди целых чисел, принадлежащих числовому отрезку  $[190201; 190280]$ , числа, имеющие ровно 4 различных **ЧЁТНЫХ** делителя. Выведите эти четыре делителя для каждого найденного числа в порядке убывания.
- 3) Имеется набор данных, состоящий из пар положительных целых чисел. Необходимо выбрать из каждой пары ровно одно число так, чтобы сумма всех выбранных чисел не делилась на 3 и при этом была максимально возможной. Гарантируется, что искомую сумму получить можно. Программа должна напечатать одно число - максимально возможную сумму, соответствующую условиям задачи.

# ЗАДАЧА

Вывести на экран монитора 5 раз слово  
«ПРИВЕТ».

# ОПЕРАТОРЫ ОРГАНИЗАЦИИ ИТЕРАЦИОННЫХ СТРУКТУР

Для реализации циклических структур используются операторы циклов:

- *с заданным числом итераций* (повторов)  
**(FOR)**
- *с неизвестным числом итераций*  
**(While, Repeat).**

# СЧЕТНЫЙ ЦИКЛ

*for*  $i:=m1$  *to*  $m2$  *do*  
    <оператор>;

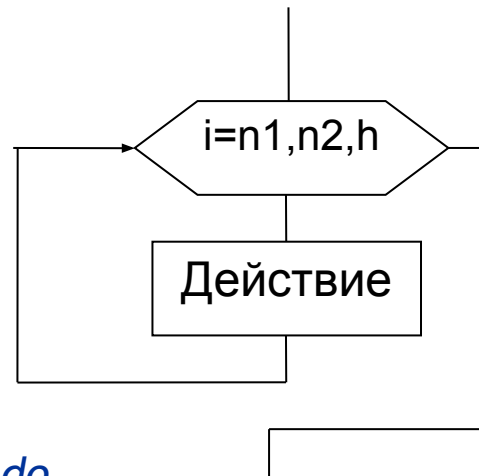
{шаг цикла +1 }

*for*  $i:=m1$  *downto*  $m2$  *do*  
    <оператор>;

{шаг цикла -1 }

*for*  $i:=1$  *to* 10 *do*  
    *begin*  
        ...  
    *end*;

*for*  $j:=10$  *downto* 1 *do*  
    *begin*  
        ...  
    *end*;



# ПРИМЕР

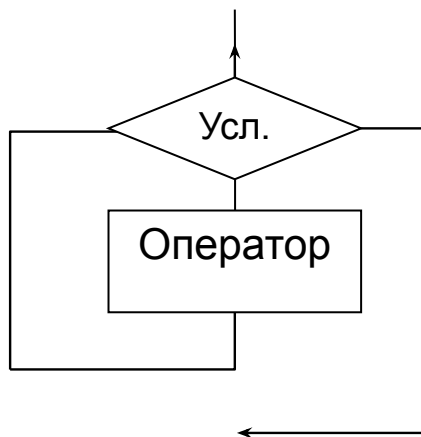
```
Var i:integer;  
Begin  
For i:=1 to 5 do  
    writeln( 'ПРИВЕТ' );  
End.
```

# ПРИМЕР

```
Var i,n:integer;  
Begin  
Write('n?');  
Readln(n);  
For i:=1 to n do  
    writeln('ПРИВЕТ');  
End.
```

# ЦИКЛ С ПРЕДУСЛОВИЕМ (ЦИКЛ-ПОКА)

```
While <усл.> do  
  <оператор>;
```



Цикл работает пока условие истинно (**TRUE**).

```
a:=0;  
While a>=0 do  
  readln(a);
```

Если в цикле требуется выполнить более одного оператора, то используется **составной оператор (операторные скобки)**:

```
While <усл.> do  
  begin  
    <оператор1>;  
    <оператор2>;  
    <оператор3>;  
    ....  
  end;
```



```
i:=1;
```

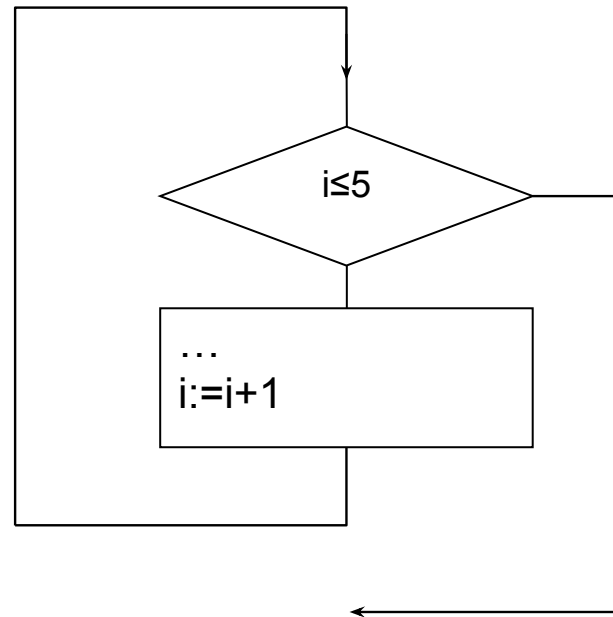
```
While i<=5 do
```

```
begin
```

```
  writeln( 'ПРИВЕТ' );
```

```
  i:=i+1;
```

```
end;
```



# ЦИКЛ С ПОСТУСЛОВИЕМ (ЦИКЛ-ДО)

## *Repeat*

<Оператор 1>;

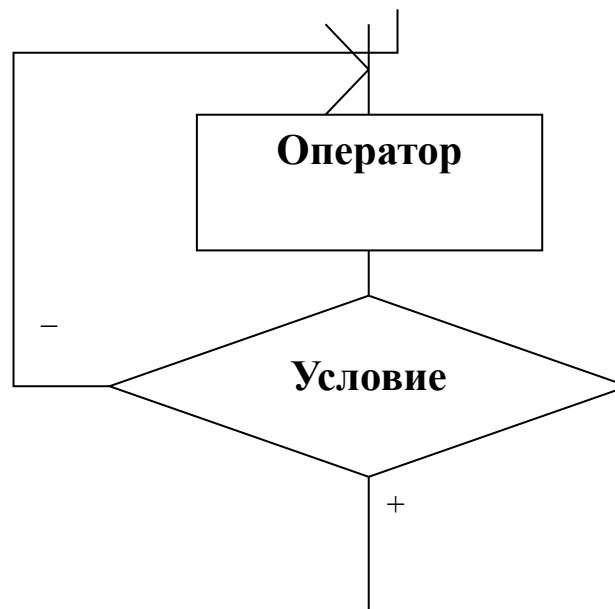
<Оператор 2>;

...

<Оператор n>

*Until* <усл.>;

*Тело цикла повторяется до тех пор, пока условие не станет истинным.*



# ЗАДАНИЕ

Вводится последовательность целых неотрицательных чисел. Вычислить среднее арифметическое последовательности

```
var i: integer;  
Begin  
  i:=0;  
  Repeat  
    i:=i+1;  
    writeln('ПРИВЕТ');  
  Until i>5;  
  
end.
```

Для повтора действий в программе предусмотрены операторы циклов.

- *Циклом называется любая многократно исполняемая последовательность действий.*
- *Последовательность инструкций, предназначенная для многократного исполнения, называется телом цикла.*
- *Единое выполнение тела цикла называется итерацией.*
- *Выражение, определяющее, будет в очередной раз выполняться итерация или цикл завершится, называется условием выхода или условием окончания цикла.*
- *Переменная, хранящая текущий номер итерации, называется **счётчиком (параметром)** цикла.*
- *Исполнение любого цикла включает:*
  - *первоначальную инициализацию переменных цикла,*
  - *проверку условия выхода,*
  - *исполнение тела цикла и*
  - *обновление переменной цикла на каждой итерации.*

# ПРИМЕР

Вывести все делители натурального числа N.

```
Program delits;  
var n, i: integer;  
begin  
  Writeln('вв. натуральное число');  
  Readln(n);  
  For i:=1 to n do  
    If n mod i=0 then write(i:3);  
end.
```

# Вложенные циклы

При программировании вложенных циклов, необходимо помнить, что переход на следующую итерацию внешнего цикла производится только после окончания внутреннего цикла.

```
Program deliteli_chisla;  
var i,j:integer;  
begin  
  For i:=1 to 5 do  
    Begin  
      Write(i);  
      For j:=1 to 3 do  
        Write('*');  
    WriteLn;  
  End;  
end.
```

```
1 ***  
2 ***  
3 ***  
4 ***  
5 ***
```

Параметры внешнего и внутреннего циклов должны быть различны!

# Вложенные циклы

При программировании вложенных циклов, необходимо помнить, что переход на следующую итерацию внешнего цикла производится только после окончания внутреннего цикла.

```
Program deliteli_chisla;  
var N,del,S:integer;  
begin  
For N:=10 to 20 do  
  Begin  
    Write(N, ':');  
    For del:=1 to N do  
      If n mod del=0 then write(del:3);  
    Writeln;  
  End;  
end.
```

```
10: 1 2 5 10  
11: 1 11  
12: 1 2 3 4 6 12  
13: 1 13  
14: 1 2 7 14  
...
```

# ЗАДАНИЕ

Написать программу, которая выводит:

a) 1  
1 1  
1 1 1  
1 1 1 1

б) \* \* \* \*  
\* \* \*  
\* \*  
\*



# ЗАДАНИЕ

- 1) Вводятся два числа. Определить наибольшее из них.
- 2) Вводятся четыре числа. Определить наибольшее из них.
- 3) Вводится последовательность из 20 различных чисел. Определить наибольшее из них.
- 4) Вводится последовательность из  $N$  различных чисел. Определить два наибольших значения.

# АЛГОРИТМ НАХОЖДЕНИЯ МАКСИМУМА

- За первоначальное значение максимума берется первое число последовательности ( $\text{Max} = A_1$ );
- Далее все числа последовательности сравниваются со значением переменной  $\text{Max}$ ;
- Если число оказалось больше максимального значения ( $a > \text{Max}$ ), то запоминаем его как новое значение ( $\text{Max} = a$ ).

```
write('Введите число: ');
readln(a);
max:=a; {первое максимальное значение}
for i:=2 to N do
  begin
    write('Введите число: ');
    readln(a);
    if a>max then max:=a; {выявление очередного
                           максимума}
  end;
writeln('max=', max);
```

# ЗАДАНИЕ

Вводится  $N$  целых чисел. Найти максимальное число, кратное трём. Если таких чисел не было, вывести слово 'NO'.

# ***АЛГОРИТМ НАКОПЛЕНИЯ СУММЫ***

# ЗАДАЧА

Вычислить сумму 5 элементов числовой последовательности.

```
Program Summa_1;  
Var A1, A2, A3, A4, A5, SUMMA:REAL;  
Begin  
    read(a1, a2, a3, a4, a5);  
    SUMMA:=A1+A2+A3+A4+A5;  
    WRITELN('SUMMA= ',SUMMA:5:1)  
End.
```

# АЛГОРИТМ НАКОПЛЕНИЯ СУММЫ

**S:=0;**

**Переменная-накопитель**  
перед началом  
суммирования должна быть  
**инициализирована**

**For i:=1 to N do**

**begin**

**readln(a);**

**S:=S+a;**

**end;**

Последовательно  
**прибавлять** очередное число  
**к переменной-накопителю**

# ЗАДАЧА

Вычислить сумму  $N$  элементов числовой последовательности.

```
Program summa_3;  
Var a,summa: real;  
    i,n:integer;  
Begin  
Write('Введите n=');Readln(n);  
Summa:=0;  
for i:=1 to n do  
    begin  
        write('введите число:');  
        readln(a);  
        summa:=summa+a ;  
    end;  
writeln('summa= ',summa:4:2)  
End.
```

# ЗАДАЧА

Посчитать количество отрицательных элементов последовательности из  $n$  целых чисел.

```
Program Kol_otr_elem;
var A,I,n,kol:INTEGER;
Begin
  write('Введите n=');  readln(n);
  kol:=0;
  FOR I:=1 TO n DO
    BEGIN
      write('ВВЕДИТЕ ЧИСЛО:');
      readln(A)
      If A<0 Then kol:=kol+1;
    END;
  WRITELN('Кол-во отрицательных элементов = ',kol)
end.
```



# ЗАДАНИЕ

- 1) Дано вещественное число  $A$  и целое число  $N$  .  
Вычислить  $A^N = A \cdot A \cdot \dots \cdot A$ .
- 2) Вводится последовательность из  $N$  чисел (Числа могут повторяться). Определить, сколько раз встречается минимальное значение.

# ЗАДАНИЕ

2) Написать программу, которая вычисляет сумму ряда

а)  $S = 2 + 3 + 4 + \dots + 100$

б)  $S = \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{11}$

в)  $S = \frac{x}{2} + \frac{x^2}{3} + \frac{x^3}{4} + \dots + \frac{x^{11}}{12}$

$$S = \frac{x}{1 \cdot 2} - \frac{x^2}{2 \cdot 3} + \frac{x^3}{3 \cdot 4} - \dots + \frac{x^{11}}{11 \cdot 12}$$