



# ЦИКЛЫ

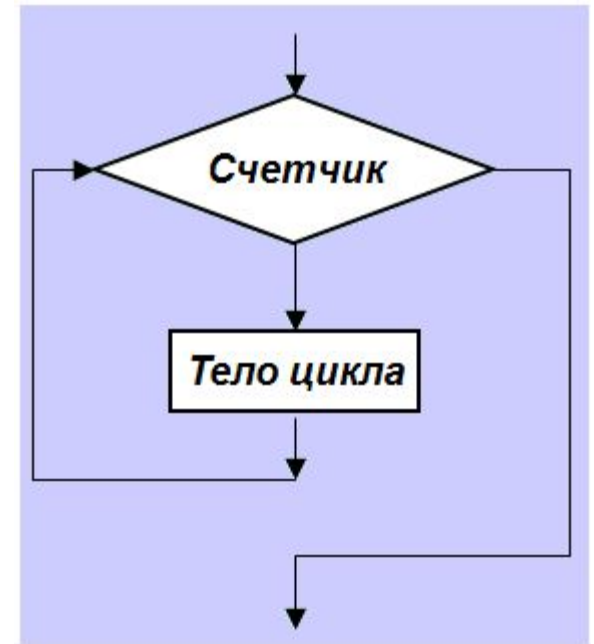


# Цикл for

- **Цикл** – это повторяющаяся последовательность операторов.
- Циклы бывают **арифметические** и **итерационные**
- **Арифметический цикл** – это такой цикл, число повторений которого известно заранее. В Pascal такой цикл обычно реализуется с помощью оператора *for*.
- **Итерационный цикл** – это такой цикл, число повторений которого заранее неизвестно и выход из цикла производится в случае выполнения или невыполнения какого-то условия. В Pascal такие циклы обычно реализуются с помощью операторов *while* и *repeat*

# Цикл for

```
for счетчик := начальное to конечное do
    значение           значение
begin
    тело цикла
end
```



**Принцип работы:** Сначала счетчику цикла присваивается начальное значение. Если это значение не больше конечного значения, то выполняется тело цикла. Затем значение счетчика увеличивается на 1 и опять сравнивается с конечным значением. Если оно по-прежнему не больше конечного значения, то оператор выполняется еще раз и так далее.

**Замечание:** если тело цикла состоит из одного оператора, то begin и end можно опустить

```
for счетчик := начальное downto конечное do  
                               значение           значение  
begin  
    тело цикла  
end
```

**Принцип работы:** как и в первом случае, пока начальное значение не меньше конечного значения, то выполняется тело цикла. Затем значение счетчика уменьшается на 1

# Решение задач ОГЭ

```
var s, n: integer;  
begin  
s:=2;  
for n:=2 to 5 do  
s:=s+n*2;  
write (s);  
end.
```

Дан фрагмент программы. Чему будет равна переменная  $s$  в результате работы данной программы?

## Решение:

В этой задаче есть цикл for. Тело цикла будет выполняться 4 раза (при  $n=2,3,4,5$ ).

$$s=2$$

$$n=2 \quad s=2+2*2=6$$

$$n=3 \quad s=6+3*2=12$$

$$n=4 \quad s=12+4*2=20$$

$$n=5 \quad s=20+5*2=30$$

Программа выводит значение переменной  $s$ , значит на экране будет число 30

**Ответ: 30**



# **Цикл While**

## **(цикл с предусловием)**

**Задача:** написать программу, которая выводит на экран все натуральные числа от 1 до n

**Решение:** умеем составлять алгоритм с циклом for

### Программа с циклом for

```
Program m1;  
var i,n:integer;  
begin  
writeln('Введите n');  
readln(n);  
for i:=1 to n do  
  write(i, ' ');  
end.
```

### Программа с циклом while

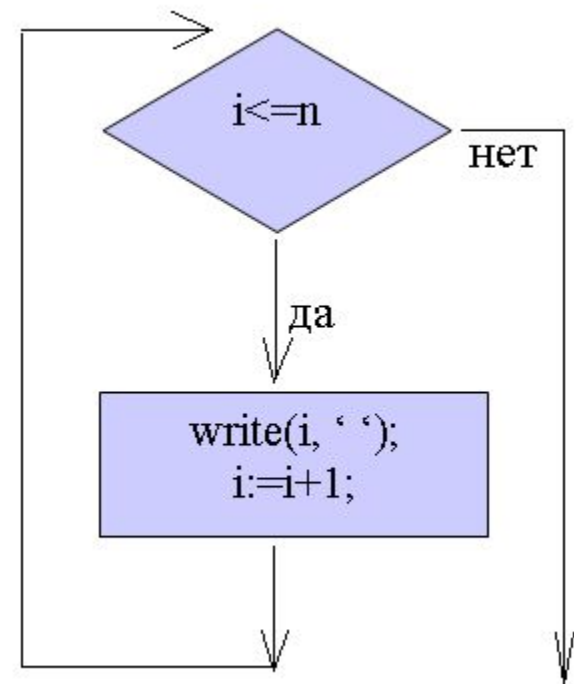
```
Program m2;  
var i,n:integer;  
begin  
writeln('Введите n'); readln(n);  
i:=1;  
while i<=n do  
  begin  
    write(i, ' '); i:=i+1;  
  end;  
end.
```



**Задача:** написать программу, которая выводит на экран все натуральные числа от 1 до  $n$

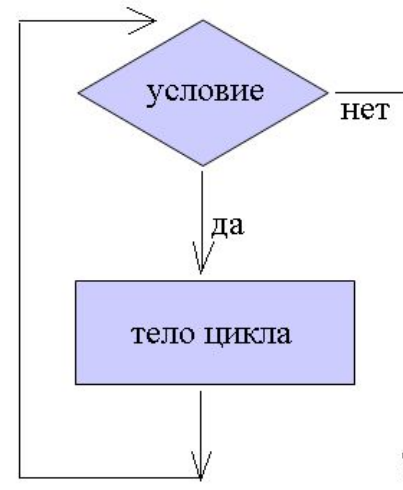
Пока истинно условие  $i \leq n$ , будет выполняться тело цикла:

- 1) вывод числа  $i$
- 2) увеличиваем на единицу значение  $i$



# Общий вид цикла while

```
while <условие> do  
begin  
    тело цикла;  
end;
```



Цикл while организует выполнение тела цикла неизвестное заранее число раз. Выход осуществляется, если условие окажется ложным.

**Принцип работы:** перед каждым выполнением цикла анализируется условие: если оно истинно, выполняется тело цикла и управление передается на повторную проверку условия. Если условие ложно – цикл завершается и управление передается на оператор, следующий за циклом while.

## Замечания:

- 1) Если условие во время цикла не будет изменяться, то возможна ситуация заикливания, т.е. цикл не закончится никогда. Поэтому внутри цикла должны находиться операторы, приводящие к изменению условия, чтобы цикл мог корректно завершиться
- 2) Если тело цикла состоит из одного оператора, то `begin` и `end` можно опустить

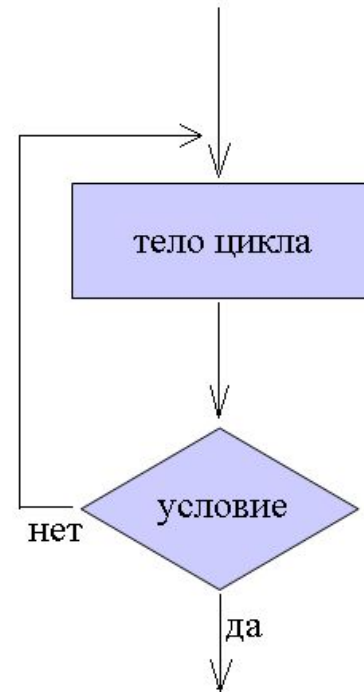
```
while <условие> do  
  тело цикла;
```



# **Цикл repeat** **(цикл с постусловием)**

# Общий вид цикла

```
repeat  
<тело цикла>  
until  
<условие>
```



**Принцип работы:** выполняется тело цикла. Если <условие> истинно, то выполнение цикла завершится. Если <условие> ложно, то снова выполняется тело цикла

## Замечания:

- 1) т. к. условие завершения цикла проверяется в конце цикла, то операторы тела цикла выполнятся хотя бы один раз
- 2) В цикле `repeat`, так же как и в операторе `while` возможна ситуация зацикливания в случае, если `<условие>` всегда будет оставаться ложным

**Задача:** написать программу, которая выводит на экран все натуральные числа от 1 до n

### **Программа с циклом while**

```
Program m2;  
var i,n:integer;  
begin  
  writeln('Введите n'); readln(n);  
  i:=1;  
  while i<=n do  
  begin  
    write(i, ' '); i:=i+1;  
  end;  
end.
```

### **Программа с циклом repeat**

```
Program m3;  
var i,n:integer;  
begin  
  writeln('Введите n'); readln(n);  
  i:=1;  
  repeat  
    write(i, ' ');  
    i:=i+1;  
  until i>n  
end.
```