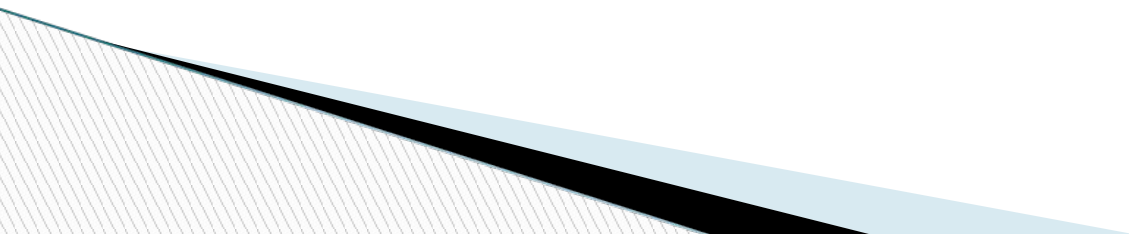


Циклы на языке *Pascal*



Разновидности циклов:

Циклы

```
graph TD; A[Циклы] --- B[Цикл ПОКА (While)]; A --- C[Цикл ДО (Repeat)]; A --- D[Цикл с параметром (For ... to ...)]
```

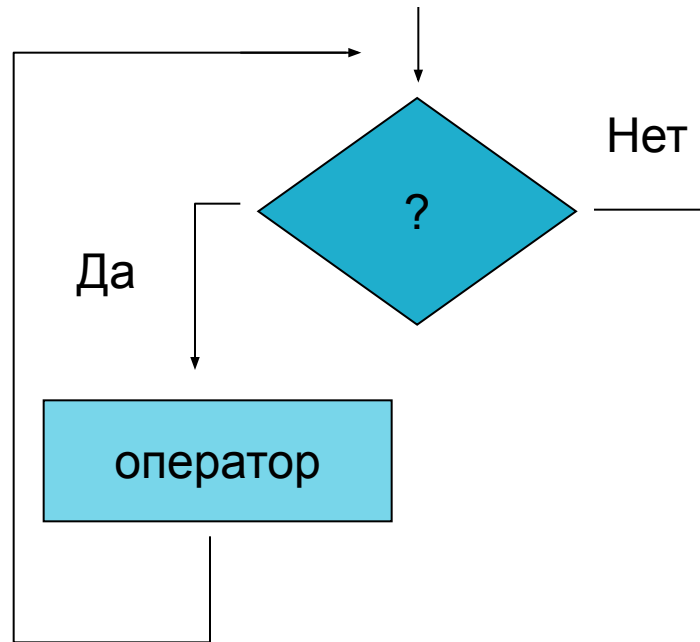
Цикл ПОКА
(While)

Цикл ДО
(Repeat)

Цикл с
параметром
(For ... to ...)

Цикл пока (думающий)

While <условие выхода из цикла> **do**
<оператор>;



Цикл пока (думающий)

- Как работает цикл:

Сначала проверяется условие и если оно верно, то осуществляется переход к телу цикла. После выполнения операторов тела цикла возвращаемся для проверки условия. Так будет повторяться до тех пор пока условие не станет ложным, после чего цикл прекратит выполняться.

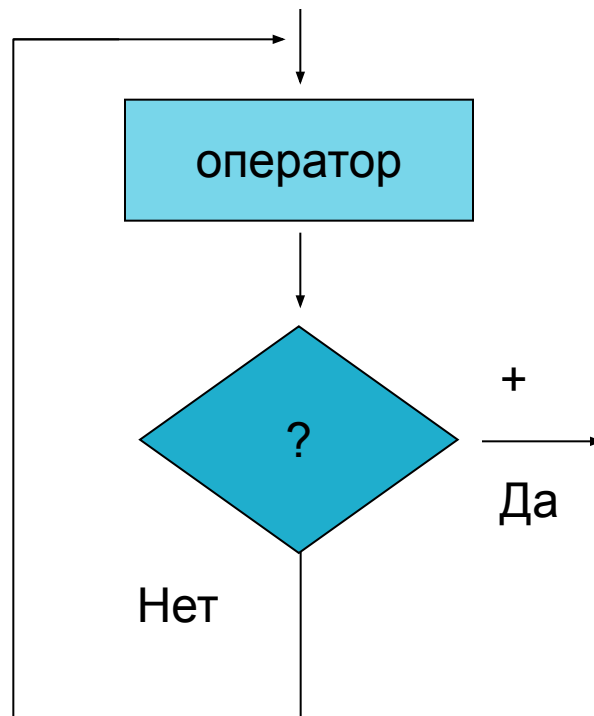
- Почему цикл называют думающий? Потому что прежде чем выполнить тело цикла он проверяет условие...
- Если тело цикла содержит 2 и больше операторов, то оно заключается в операторные скобки **begin ... end**.

Цикл ДО (сначала делаю)

Repeat

<операторы>

Until <условие выхода из цикла>;



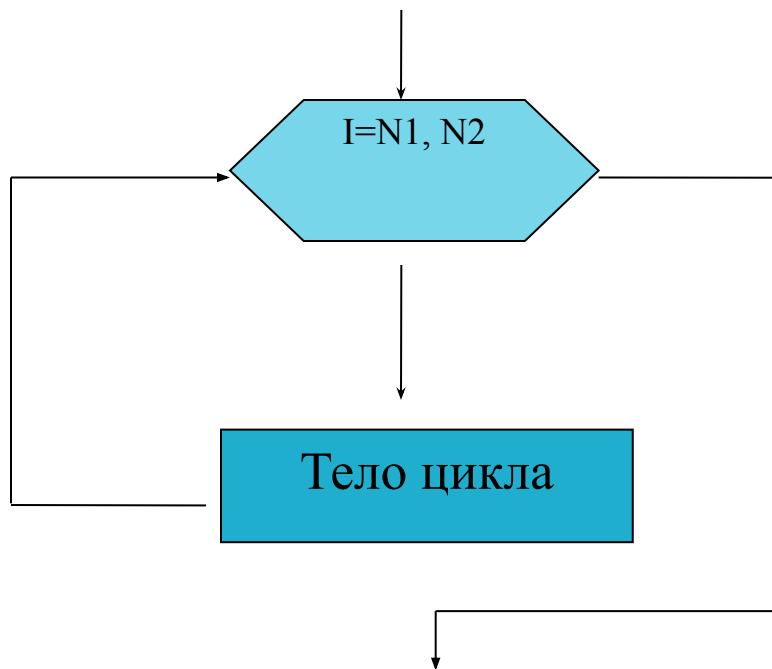
Цикл ДО (сначала делаю)

- Как работает цикл:

Сначала выполняется тело цикла, после чего осуществляется переход к проверке условия. Если условие ложно, то возвращаемся к выполнению тела цикла. Так будет повторяться до тех пор пока условие не станет верным, после чего цикл прекратит выполняться.

- Почему цикл называют сначала думаю? Потому что этот цикл всегда выполнит тело цикла хотя бы один раз.
- Т.к. тело цикла находится между служебными словами **repeat ... until**, то операторные скобки не требуются.

Цикл с параметром



▣ Прямой пересчет

For $i := N1$ **to** $N2$ **do**

<операторы;>

$N2 \geq N1$

▣ Обратный пересчет

For $i := N2$ **down to** $N1$
do

<операторы>;

$N2 \geq N1$

Цикл с параметром

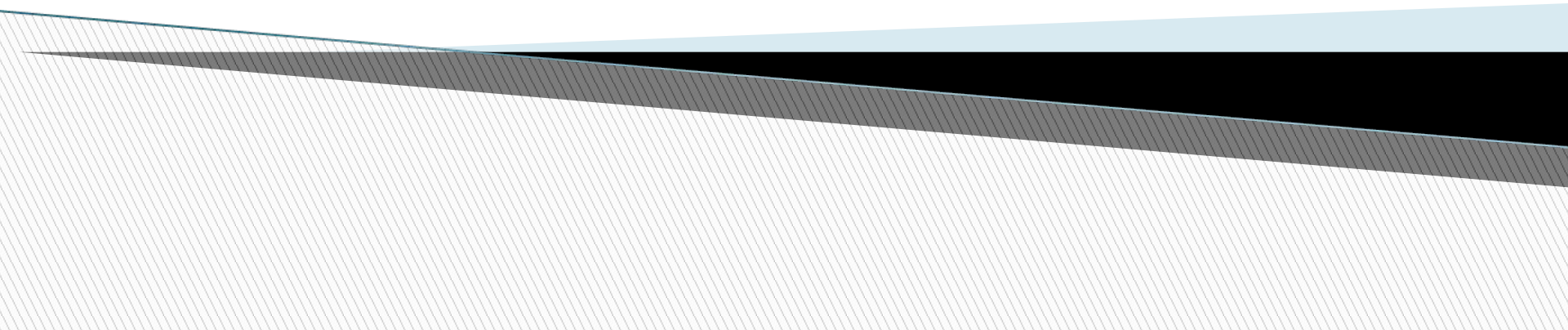
- Как работает цикл:

Сначала переменной i присваивается первоначальное значение $N1$, затем значение i сравнивается с конечным значением $N2$ и если $i \leq N2$, то осуществляется переход к телу цикла. После выполнения тела цикла значение i увеличивается ровно на 1 и опять повторяется сравнение с $N2$. Так продолжается до тех пор пока значение i не станет больше значения $N2$.

Цикл обратного пересчета работает аналогично с разницей в том, что значение i каждый раз уменьшается на 1 и $i \geq N1$. Если значение i станет меньше значения $N1$, то цикл прекратит работу.

- Если тело цикла содержит 2 и больше операторов, то оно заключается в операторные скобки **begin ... end**.

Вывод: В программах можно использовать любой оператор цикла. Есть ограничения в использовании цикла *For* (шаг изменения параметра цикла должен быть *1* или *-1*. Если не известно сколько раз будет выполняться цикл, то используем *While* или *Repeat*)



Примеры программ

- Для приведенных на следующих слайдах программ определить, какие операторы образуют тело цикла, какая переменная является управляющей переменной цикла. Заполнить таблицу:

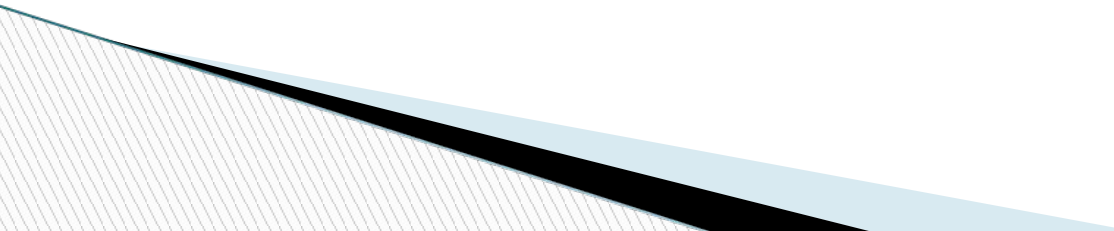
№ задачи	Управляющая переменная цикла	Начальное значение	Конечное значение	Шаг изменения	Количество повторений цикла
1					
2					

1. Определите, что будет напечатано в результате работы следующей программы.

```
Program Prog1;  
var a,b,i: integer;  
begin  
    a:=1;  
    b:=3;  
    for i:=6 downto 1 do  
        b:=b+i;  
        if a+b>25 then  
            b:=b-a  
        else  
            b:=b+a;  
        writeln (b)  
    end.
```

2. Определите, что будет напечатано в результате работы следующей программы, если $x=-5$, а $b=-2$.

```
Program pr2;  
  var  
    x,b,y:real;  
    a: integer;  
begin  
  write('x='); readln (x);  
  write('b='); readln (b);  
  for a:= - 2 to 1 do  
    begin  
      y:=a*x*x + b;  
      writeln ('y=',y)  
    end  
end.
```

- Написать программу, которая подсчитывает сумму чисел от 1 до 100.
 - Написать программу, которая подсчитывает сумму нечетных чисел от 1 до 100.
- 

**Таблицу (10 слайд),
ответы к 11 и 12 слайдам,
программы (слайд 13)
написать в тетради!**

