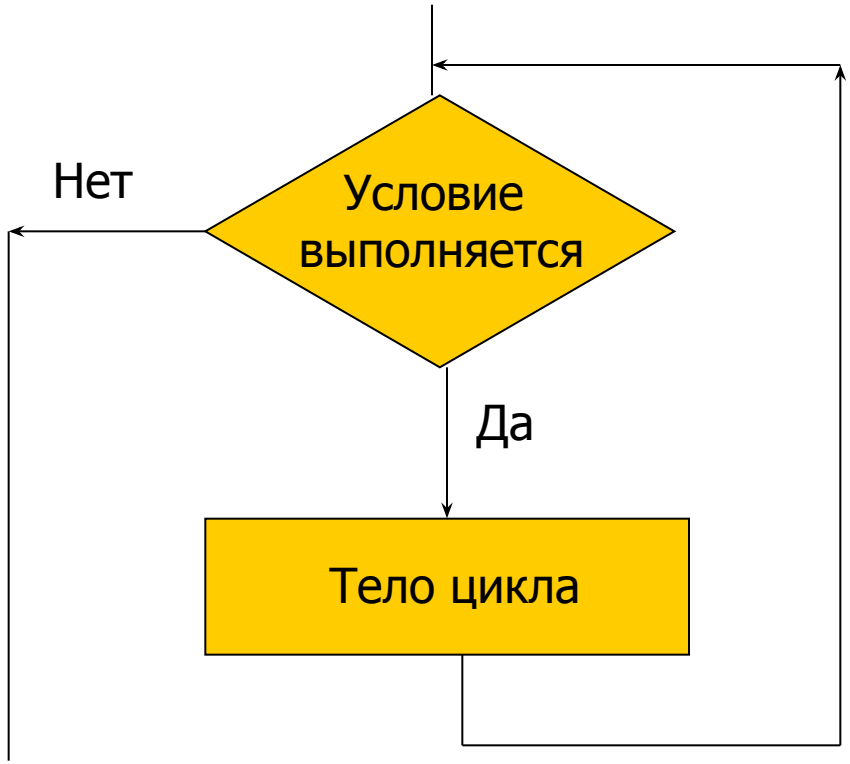
# Циклические конструкции

1. **Цикл с предусловием**
2. **Цикл с** постусловием
3. **Цикл с** параметром
4. Вложенные **циклы**

# предусловием



# Цикл с предусловием

```
WHILE  <логическое условие> DO
            <оператор цикла> ;

WHILE  <логическое условие> DO
            BEGIN
            <оператор 1> ;
            <оператор 2> ;
            <оператор N>
            END;
```

# Цикл с предусловием

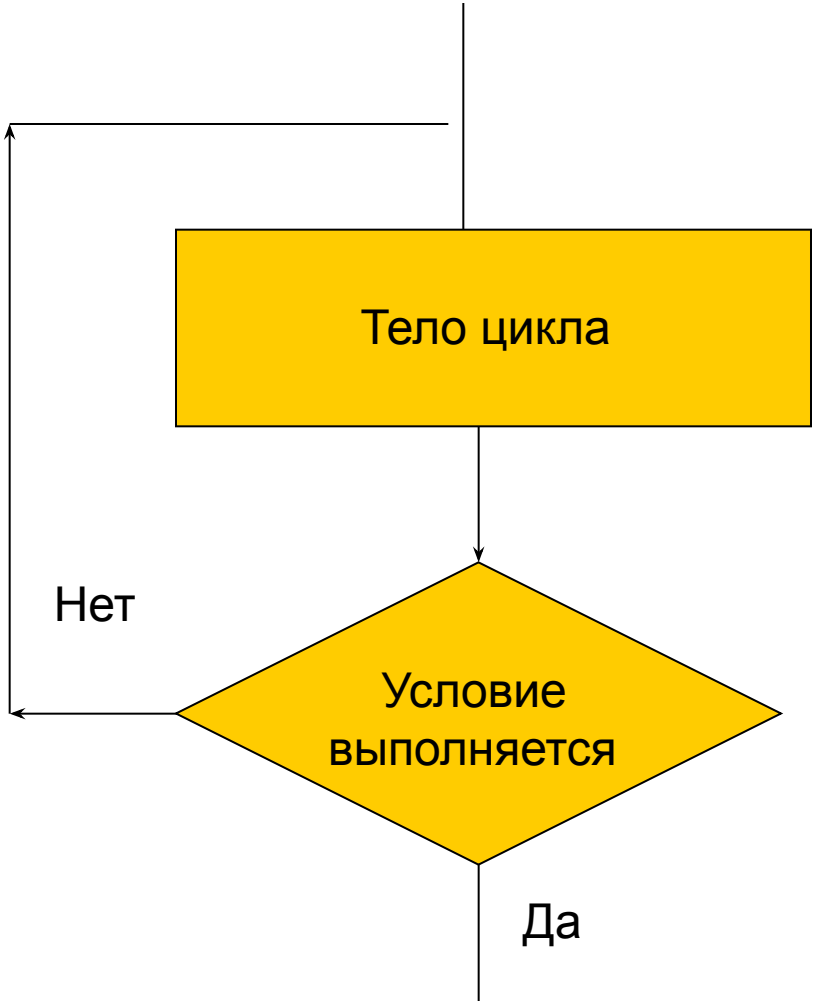
Операторы цикла *выполняются* (DO), *пока* (WHILE) логическое выражение истинно

# Цикл с предусловием

Фрагмент программы вывода на экран квадратов целых чисел от 5 до 10

```
x:=5;
WHILE X<=10 DO
    BEGIN
    Y:=SQR (X) ;
    WRITELN (’квадрат’,X,’=‘ ,Y );
    X:= X+1;
    END;
```

# Цикл с постусловием



# Цикл с постусловием

```
REPEAT
 оператор 1;
 оператор 2;
 оператор n
UNTIL <логическое условие> ;
```

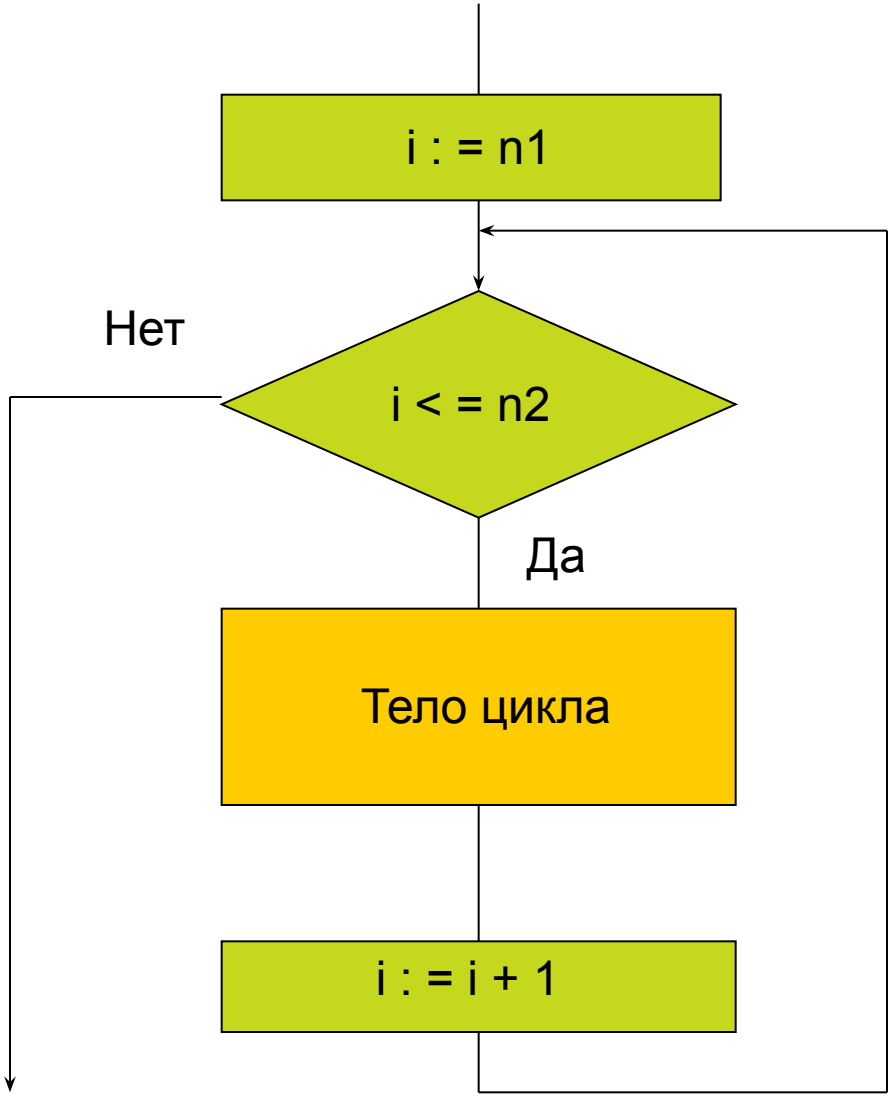
Операторы *повторяются* (REPEAT) *до тех пор* (UNTIL), пока значение логического выражения ложно

# Цикл с постусловием

Фрагмент программы вывода на экран квадратов целых чисел от 5 до 10

```
x:=5;
REPEAT
    Y:=SQR (X) ;
    WRITELN (’квадрат’,X,’=‘ ,Y );
    X:= X+1
UNTIL x >= 10 ;
```

# Цикл с параметром



# Цикл с параметром

FOR i : = n1 TO n2 DO

i - параметр цикла

n1 - начальное значение (выражение)

n2 - конечное значение (выражение)

n1 < n2

шаг приращения + 1

количество повторений n2-n1+1

при n1 = n2 - 1 раз

при n1 > n2 цикл не выполняется ни разу

# Цикл с параметром

FOR i : = n1 DOWNTO n2 DO

n1 > n2

шаг приращения – 1

при n1 = n2 - 1 раз

при n1 < n2 цикл не выполняется ни разу

i , n1, n2 – должны быть описаны (порядковый тип)

i - не переопределяется в цикле

# Цикл с параметром

Фрагмент программы вывода на экран квадратов целых чисел от 5 до 10

```
FOR x : = 5   TO   10  DO
BEGIN
    Y:=SQR (X) ;
    WRITELN (’квадрат’,X,’=‘ ,Y );
END;
```

# Цикл с параметром

Фрагмент программы вывода на экран квадратов целых чисел от 5 до 10

```
FOR x : = 10   DOWNTO   5  DO
BEGIN
    Y:=SQR (X) ;
    WRITELN (’квадрат’,X,’=‘ ,Y );
END;
```

# Цикл с параметром

## Примеры вычисления сумм, количеств и произведений

Алгоритмы подсчета:

- **сумм**

*Первоначальное значение S: =0*

***Подсчет в цикле:***

***S:=S+{текущее значение}***

# Цикл с параметром

- Вычислить: S=I+2+3+...N.

```
var  n,s,i : integer;
begin
writeln (‘Введите натуральное число N=');
readln  (n) ;
s : = 0 ;
for i : = 1 to n do  s : = s + i ;
writeln (‘сумма = ‘ , s)
end.
```

# Цикл с параметром

- **количества**

*Первоначальное значение k : = 0*

***Подсчет в цикле:***

***k : = K + 1***

# Цикл с параметром

- Подсчитать количество нечетных чисел, введенных с клавиатуры

```
var k,a,i: integer;
begin
k:=0;
for i:=1 to 5 do
    begin
        write ( 'Введите целое число ‘ );
   readln ( a );
        if  a mod 2 <> 0  then  k := k +1;     end;
writeln ( 'Нечетных чисел  ', k );
readln
end.
```

# Цикл с параметром

- произведений

*Первоначальное значение p:=1*

***подсчет в цикле:***

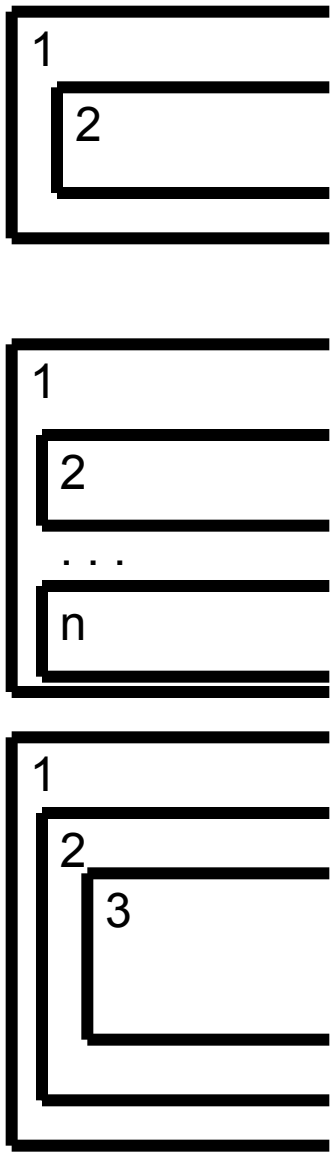
***p:=p * {множитель }***

# Цикл с параметром

- . Вычислять N!

```
var
n, p, i : integer;
begin
   writeln ( ‘введите число n = ‘) ;
   readln   (n) ;
   p:=1;
    for i:=1 to n do
    p:=p*i;        .
   writeln ( ‘факториал числа  ’, n,’ =  ' ,p)
end.
```

# Вложенные циклы




Цикл называется сложным (вложенным), если внутри его содержится один или несколько других циклов

# Вложенные циклы

- ***Среди чисел 1< N < 100 найти все пары чисел, для которых их сумма равнялась бы их произведению***

# Вложенные циклы

```
var  k, a, b: integer;
begin k:=0;
for a:= 1 to 100 do
for b:= 1 to 100 do begin
if a+b=a*b then  begin
            k:=k+1;  writeln('Числа ', a , b)
            end
            end;
if k=0 then writeln('Таких чис.нет')else('k=',k)
end.
```