

# ЗНАЙ И ЛЮБИ ОПЕРАТОРЫ ЦИКЛА



СО СЧЁТЧИКОМ



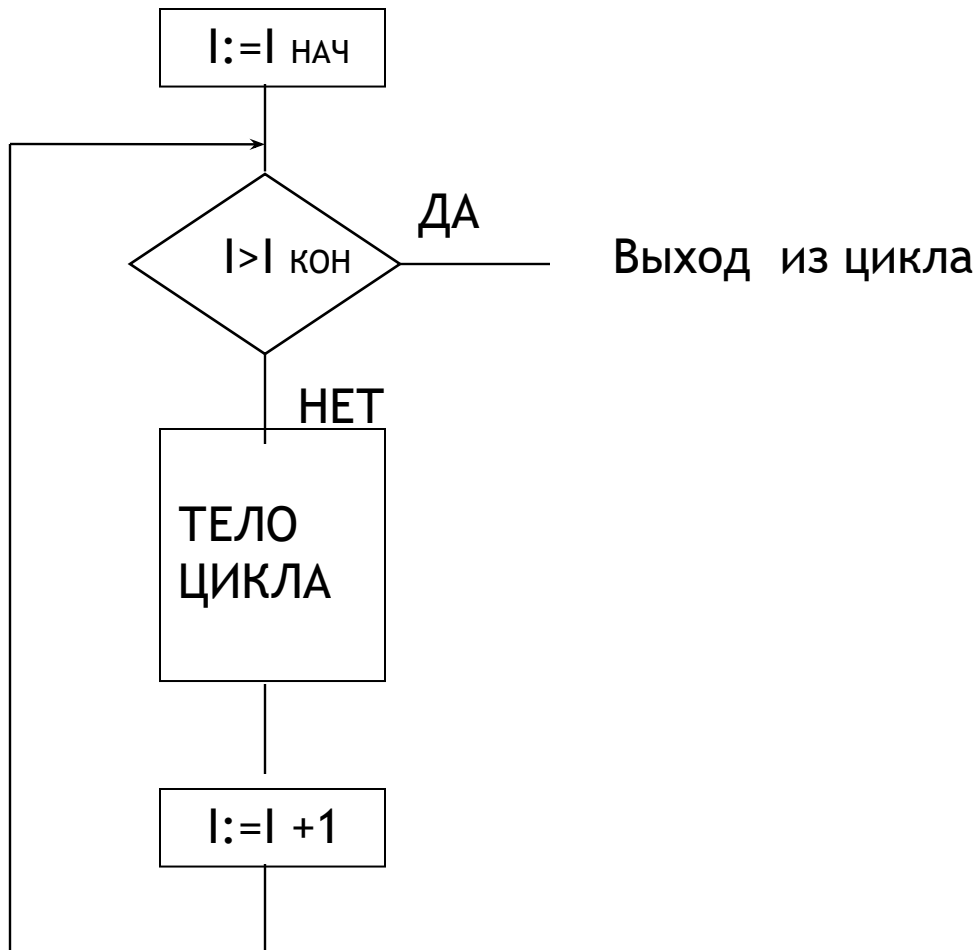
С ПРЕДУСЛОВИЕМ



С ПОСЛЕУСЛОВИЕМ

# ДЕЙСТВИЕ ОПЕРАТОРА FOR:

Организует многократное исполнение тела цикла



## ЗАПОМНИ ТЕРМИНЫ:

- СЧЁТЧИК( или параметр) цикла - искусственно вводимый объект для организации работы цикла; настройка начального и конечного значения счётчика задаёт количество раз работы цикла;
- ТЕЛО ЦИКЛА - оператор(может быть составной) для многократного исполнения;
- ПРОХОД ПО ЦИКЛУ - однократное исполнение тела цикла;
- ЗАЦИКЛИВАНИЕ - бесконечное исполнение цикла (это очень плохо!);

# СИНТАКСИС ОПЕРАТОРА ЦИКЛА СО СЧЁТЧИКОМ



## ПРИМЕРЫ:

1. FOR J:=1 TO 20 DO X:=X-1;
2. FOR K:=-5 TO 0 DO BEGIN S:=S\*P; WRITE(S) END;
3. FOR AVATAR:=A+B TO C-D DO IF (AVATAR MOD 2)=0 THEN WRITE(AVATAR:6);

# УСЛОВИЯ КОРРЕКТНОЙ РАБОТЫ ОПЕРАТОРА ЦИКЛА:

1. Счётчик - объект перечисляемого типа!
2. Шаг изменения счётчика задается автоматически +1(с to) или -1(с downto)!
3. Пусть А-нач. Знач. Счётчика, В-кон. Знач. Счётчика;  
тогда количество проходов по телу цикла
 

|     | с <u>to</u> | с <u>downto</u> |
|-----|-------------|-----------------|
| А=В | 0           | 0               |
| А<В | В-А+1       | 0               |
| А>В | 0           | А-В+1           |
4. В теле цикла изменять значение счётчика **НЕЛЬЗЯ!**
5. Работает правило `begin.....end` для тела цикла

# ПРИМЕРЫ ИЗ ЖИЗНИ:

1. For i:=1 to 99 do write(i\*i:6);
2. For j:=50 downto 1 do writeln(2\*j);
3. A:=2;b:=102; For k:=a to b do if (k mod 3)=0 then p:=p+1;
4. For l:='a' to 'я' do write(ord(l):5);
5. For m:=1 to length(str) do writeln(copy(str),m,1);
6. For n:='z' to 'a' do str:=str+n;
7. SUM:=0; For p:=10 to 101 do begin read(S);SUM:=SUM+S end; write(SUM);
8. For r=32 to 255 do writeln(chr(r),'-',r:4);
9. Progress:=1; Q:=1.21; For t:=33 downto 1 do write(progress\*Q);
10. For s:=1 to 128 do If (s mod 10)=0 then write('\*') else write(' ');

# НАЙТИ ПОЗИЦИИ БУКВЫ В ФРАЗЕ

```
Program letter;  
Var fr:string; buk:char; i,j,L:byte;  
Label Mm;  
Begin write('введи фразу'); readln(fr);  
      write('введи букву');readln(buk);  
L:=length(fr); j:=0;  
Mm:  i:=pos(buk,fr);  
If i<>0 then begin j:=j+i; write(j:4);  
fr:=copy(fr,i+1,L-i); goto Mm end;  
Write('end');  
End.
```

P

A

S

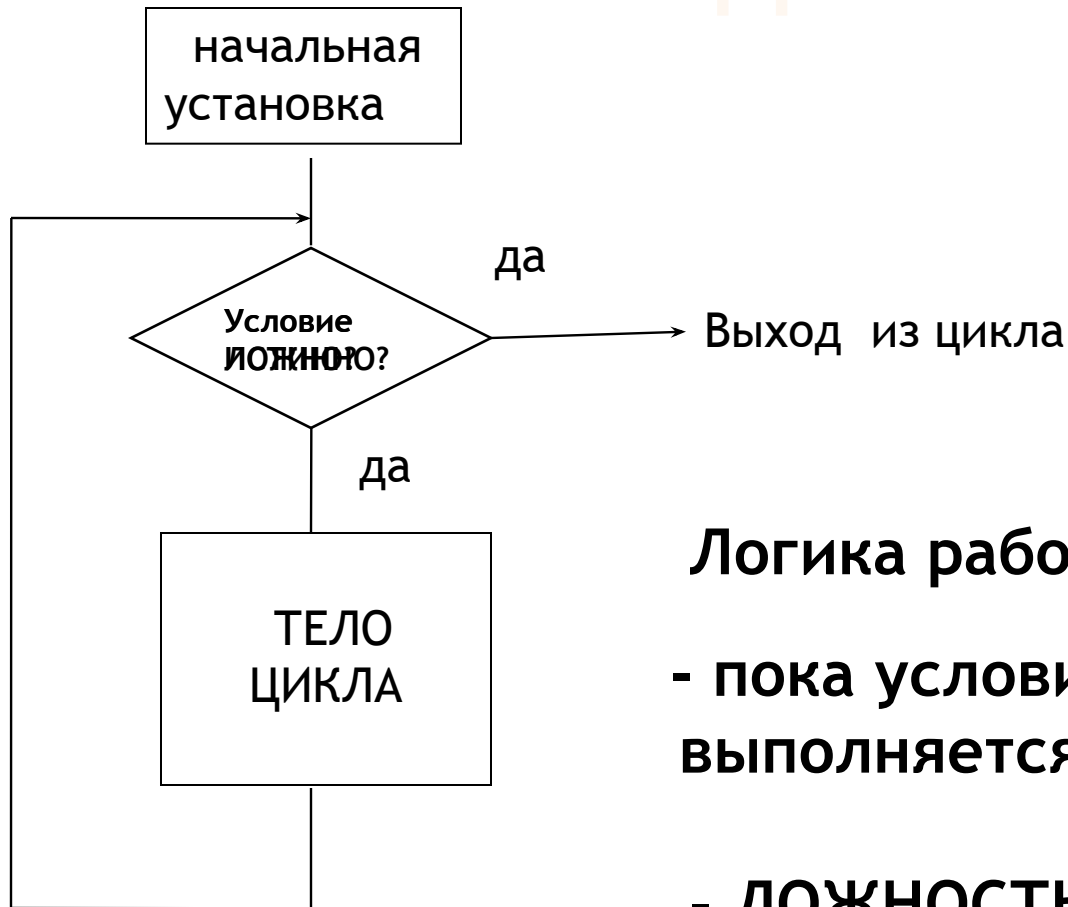
C

A

L



# ДЕЙСТВИЕ ОПЕРАТОРА С ПРЕДУСЛОВИЕМ :



**Логика работы цикла:**

- пока условие **ИСТИННО** выполняется тело цикла;
- **ЛОЖНОСТЬ** условия завершает цикл;

# ПРИМЕНЕНИЕ ОПЕРАТОРА В ЗАДАЧАХ:

## ***Задача 1***

Автомат для оплаты услуг принимает монеты любого достоинства и выдает сдачу. Надо прекратить прием монет, как только денежная сумма станет не меньше суммы оплаты услуг, и выдать сдачу.

## ***Задача 2***

Построить расписание движения автобуса, если известно время начала и конца работы маршрута, длительность рейса в один конец, время отдыха на конечных остановках.

## ***Задача 3***

Для уравнения с двумя неизвестными подобрать корень с заданной точностью( например , 0.001).

# СИНТАКСИС ОПЕРАТОРА:



Например:

```
While x >= y do begin x:=x-1;y:=y+1; end;
```

# ЛИСТИНГ РАБОТЫ ЦИКЛА:

```
X :=10; y :=2; while X>=Y do begin X :=X-1;Y :=Y+1 end;
```

| действия       | X>=Y   | X  | Y |
|----------------|--------|----|---|
|                |        | 10 | 2 |
| Запуск цикла   | истина | 9  | 3 |
|                | истина | 8  | 4 |
|                | истина | 7  | 5 |
|                | истина | 6  | 6 |
|                | истина | 5  | 7 |
| Выход из цикла | ложь   |    |   |

Тело цикла исполнится 5 раз !

# ЛИСТИНГ РАБОТЫ ЦИКЛА:

```
X :=2; Y:=2; while X>=Y do begin X :=X-1;Y :=Y+1 end;
```

| действия       | X>=Y   | X | Y |
|----------------|--------|---|---|
|                |        | 2 | 2 |
| Запуск цикла   | истина | 1 | 3 |
|                | ложь   |   |   |
| Выход из цикла |        |   |   |

Тело цикла исполнится 1 раз !

# ЛИСТИНГ РАБОТЫ ЦИКЛА:

```
X:=2; y:=2; while X>=Y do begin X:=X+1;Y:=Y+1 end;
```

| действия     | $X \geq Y$ | X | Y |
|--------------|------------|---|---|
|              |            | 2 | 2 |
| Запуск цикла | истина     | 3 | 3 |
|              | истина     | 4 | 4 |
|              | истина     | 5 | 5 |
|              | истина     | 6 | 6 |
|              | истина     | 7 | 7 |

Цикл не завершается

Произошло зацикливание !

# ЛИСТИНГ РАБОТЫ ЦИКЛА:

```
X :=2; Y :=3; while X>=Y do begin X :=X+1;Y :=Y+1 end;
```

| действия | $X \geq Y$ | X | Y |
|----------|------------|---|---|
|----------|------------|---|---|

|              |      |   |   |
|--------------|------|---|---|
| Запуск цикла | ложь | 2 | 3 |
|--------------|------|---|---|

Выход из цикла

Тело цикла не исполнится ни разу !

# УСЛОВИЯ КОРРЕКТНОЙ РАБОТЫ ОПЕРАТОРА ЦИКЛА:

1. **Необходимо обеспечить запуск цикла:**  
тело цикла исполнится хотя бы один раз, если при запуске цикла условие ИСТИННО!
2. **Необходимо обеспечить завершение цикла:**  
в ходе работы цикла условие должно изменить свое значение на ЛОЖЬ!
3. **Количество проходов по телу цикла задается :** -  
начальной установкой переменных, участвующих в цикле;  
-изменением значений переменных в теле цикла;



## ПРИМЕРЫ ИЗ ЖИЗНИ:

1.  $S := 0$ ; While  $S < 100$  do begin read(money);  
 $S := S + \text{money}$ ; END;
2.  $\text{score} := 0$ ; While  $\text{score} < 33$  do begin read(in);  
if  $\text{in} > 0$  then  $\text{score} := \text{score} + 1$ ; end;
3.  $x := 0$ ;  $\text{eps} := 0.01$ ; While  $F1 - F2 > \text{eps}$  do begin  
 $F2 := F1$ ;  $F1 := x^3 + x^2 + 5x - 12$ ;  $x := x + \text{eps}$ ; end;
4.  $i := 1$ ;  $a := a_0$ ; While  $i < 50$  do begin  $a := a + 1$ ;  
if  $a \bmod 2 = 0$  then  $i := i + 1$ ; write( a:5);

# ПОСТРОИТЬ ПРОГРАММУ ДЛЯ РЕШЕНИЯ ЗАДАЧИ:

Разменный автомат имеет набор монет достоинством  $k_1$ ,  $k_2$  и  $k_3$ , причем  $k_1 > k_2 > k_3$ ,  $k_3 = 1$  рубль.

Предложить вариант размена заданной денежной суммы, которая поступает от клиента в виде набора купюр, с минимальным количеством выдаваемых монет.

Признак окончания ввода купюр - введена сумма в 0 рублей.

# ЗАПОМНИ:

- 1- В цикле **WHILE** условие **ПРОДОЛЖАЕТ** цикл.
- 2- Количество проходов по телу цикла **заранее неизвестно.**
- 3- Логика работы цикла:
  - пока условие **ИСТИННО**  
выполняется тело цикла;
  - **ЛОЖНОСТЬ** условия  
завершает цикл;
- 4 - Для тела цикла работает правило операторных скобок