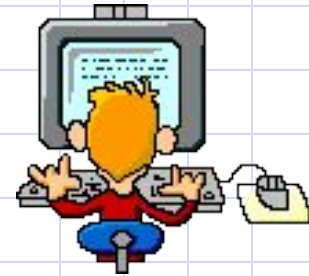


Программирование циклов



ЭТАПЫ РЕШЕНИЯ ЗАДАЧ НА ЭВМ



- **1. Постановка задачи.**
- **2. Математическая формализация.**
- **3. Построение алгоритма.**
- **4. Составление программы.**
- **5. Отладка и тестирование программы.**
- **6. Проведение расчетов и анализ полученных результатов.**

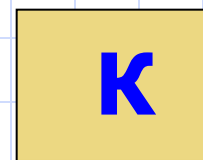
Задача



- Дано N кубиков, на которых написаны разные буквы. Сколько различных N -буквенных сочетаний можно составить из этих кубиков (слова не обязательно должны иметь смысл)?

- Дано: N

- Найти: F



И К

К И

Комбинаторика



Количество различных комбинаций из N предметов, получаемых изменением их порядка, называется **числом перестановок**. Это число выражается функцией от N , которая называется **факториалом** и записывается так:

$N!$ – N факториал

$$1! = 1$$

$$2! = 1 \times 2 = 2$$

$$3! = 1 \times 2 \times 3 = 6$$

$$4! = 1 \times 2 \times 3 \times 4 = 24$$

$$5! = 1 \times 2 \times 3 \times 4 \times 5 = 120 \text{ и т.д.}$$

$$F = N! = 1 \times 2 \times \dots \times N$$

алг СЛОВА

цел F, N, R

нач

ввод N

$F:=1$

$R:=1$

пока $R \leq N$ повторять

нц

$F:=F \cdot R$

$R:=R+1$

кц

вывод F

кон

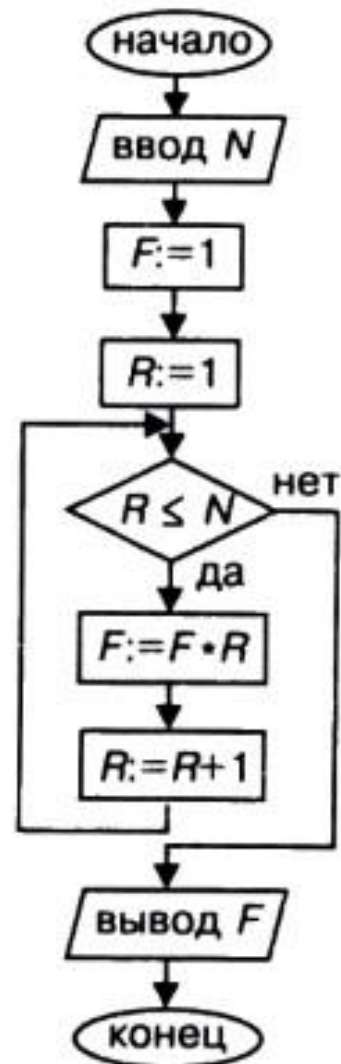


Рис. 2.7. Блок-схема алгоритма вычисления факториала

Шаг	Операция	N	F	R	Условие
1	ввод N	3	-	-	
2	$F:=1$		1	-	
3	$R:=1$			1	
4	$R \leq N$				$1 \leq 3$, да
5	$F:=F * R$		1		
6	$R:=R+1$			2	
7	$R \leq N$				$2 \leq 3$, да
8	$F:=F * R$		2		
9	$R:=R+1$			3	
10	$R \leq N$				$3 \leq 3$, да
11	$F:=F * R$		6		
12	$R:=R+1$			4	
13	$R \leq N$				$4 \leq 3$, нет
14	вывод F		6		
15	конец				

Цикл с предусловием, «пока».

WHILE ... DO

Цикл – это команда исполнителю многократно повторить указанную последовательность команд.

- **WHILE** условие **DO** оператор (тело цикла) ;
- Цикл выполняется до тех пор, пока истинно условие
- Если несколько операторов в теле цикла, то **BEGIN END**

```
Program Words;  
var F, N, R: integer;  
begin  
    write('Введите число букв');  
    readln(N);  
    F:=1;  
    R:=1;  
    while R<=N do  
        begin  
            F:=F*R;  
            R:=R+1  
        end;  
    write('Из ',N,' букв можно составить ',  
          F,' слов')  
end.
```


Практическая работа:

- 1) Протестировать программу при входных данных: $N=6$.
- 2) Построить таблицу квадратов чисел от 1 до 10 с помощью цикла `while`.
- 3) Дано целое число c и натуральное a . Написать программу, вычисляющую c^a .

Коротко о главном:

Последовательность этапов работы программиста при решении задачи на компьютере называется технологией решения задачи на компьютере.

Таких этапов шесть:

- 1) постановка задачи;
- 2) математическая формализация;
- 3) построение алгоритма;
- 4) составление программы на языке программирования;
- 5) отладка и тестирование программы;
- 6) проведение расчетов и анализ полученных результатов.

Количество различных комбинаций из N предметов, получаемых изменением их порядка, называется числом перестановок. Число перестановок равно $N!$

(N факториал):

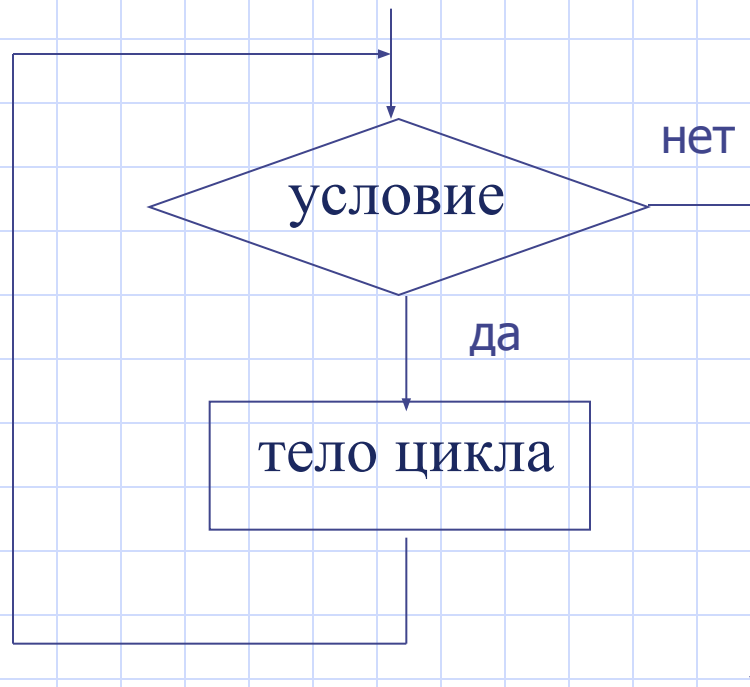
$$N! = 1 * 2 * \dots * N.$$

Любой циклический алгоритм может быть построен с помощью команды «цикл-пока» (цикл с предусловием).

Формат оператора цикла с предусловием в Паскале: `while <выражение> do <оператор>`

Оператор, составляющий тело цикла, может быть простым или составным.

Блок-схема цикла с предусловием



```
While (x<0) do  
x:=x+1;
```

```
While (i<100) do  
begin  
  writeln('i=',i);  
  i:=i+1;  
end;
```

Практическая работа

- Ввести и отладить программу вычисления факториала (стр.375).
- Протестировать на значении **N=6** (ответ – 720 слов)

Домашнее задание

Алгоритм Евклида

Наибольший общий делитель двух натуральных чисел (НОД) – это самое большое натуральное число, на которое они делятся нацело.

$$\text{НОД}(12, 18) = 6$$

Постановка задачи

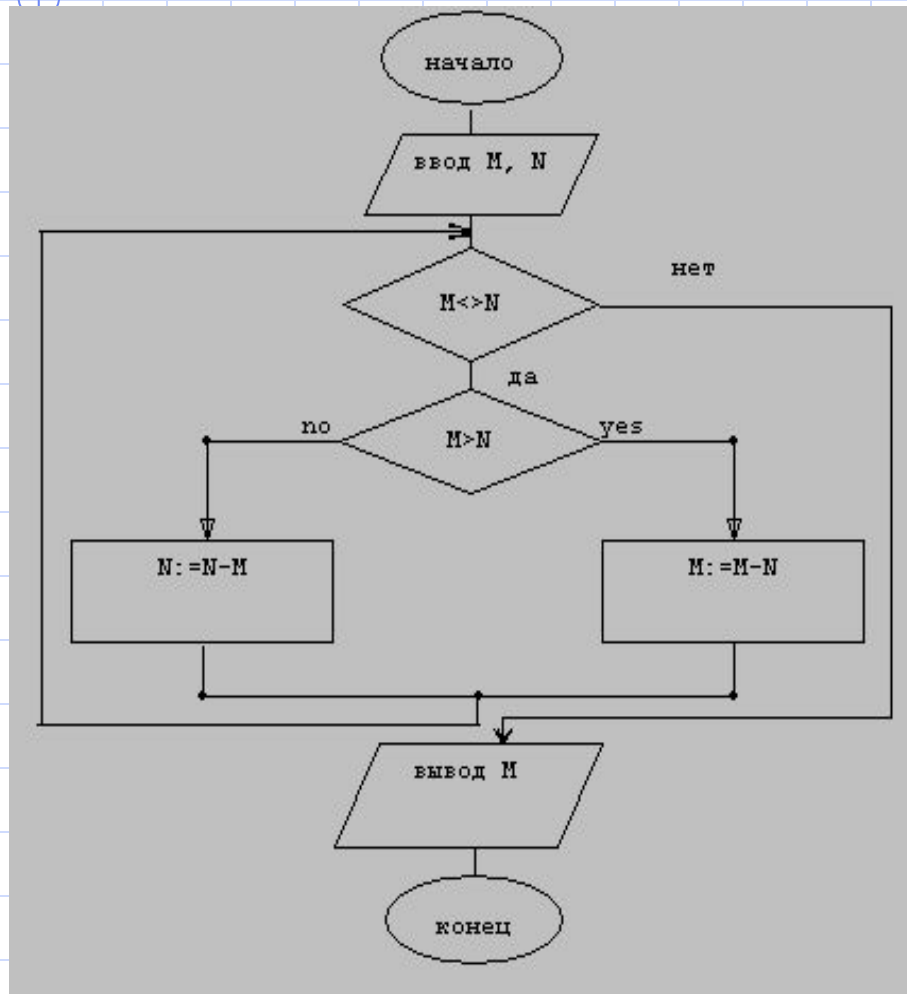
Дано: M, N

Найти: НОД(N, M)

Решение задачи

Из большего числа вычесть меньшее до тех пор пока числа не будут равны.

Блок-схема



```
Program Evklid;  
Var M, N: integer;  
Begin
```

```
  Writeln('Введите M и N');  
  Readln(M, N);
```

```
  While M > N do  
    begin
```

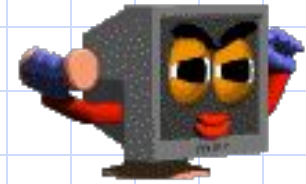
```
      if M > N then M := M - N  
      else N := N - M
```

```
    end;
```

```
  Write('НОД=', M);  
  Readln
```

```
End.
```

Практическая работа



- №7 Цикл с предусловием

```
Program Summa1;  
Var S, N, a: integer;  
Begin  
  Writeln('Введите N');  
  Readln(N);  
  a:=1; S:=0;  
  While a<=N do  
    begin  
      S:=S+a;  
      a:=a+1;  
    end;  
  Write('сумма =', S);  
  Readln  
End.
```

- №9

Выполнить на компьютере программу **Evklid**. Протестировать ее на значениях:

M = 32, N = 24


Ответ: **8**

M = 696, N = 234

Ответ: **6**

Домашнее задание

- § 12.6, 12.7 (читать, ответить на вопросы: №1-6 устно, № 8 письменно)



**§12.7, §12.6
№8**

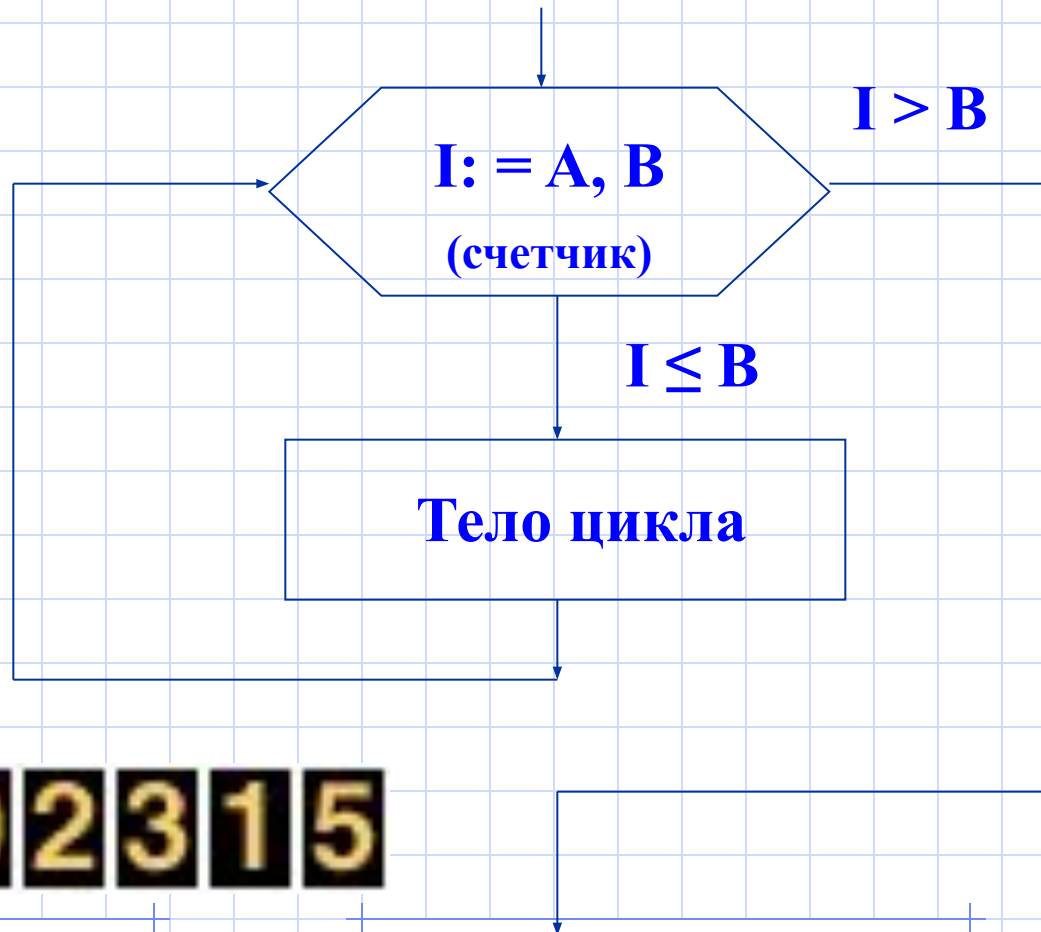
Программирование циклических алгоритмов

Существуют 3 разновидности циклов:

- Цикл "Пока",
- Цикл "До",
- Цикл "Для".

Цикл со счетчиком (для)

в котором тело цикла выполняется определенное количество раз



Цикл с увеличением параметра

FOR I:= A TO B DO

<Тело цикла>;



I – параметр, увеличивается на 1

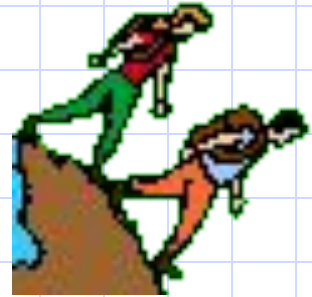
A – начальное значение

B – конечное значение

Тело цикла – один оператор!!!

Или – скобки **BEGIN ... END;**

Цикл с уменьшением параметра



- **FOR I:=A DOWNTO B DO**
<ТЕЛО ЦИКЛА> ;
- выполняется аналогичным образом, но значение параметра уменьшается на 1.

Цикл с постусловием, цикл «до» Операторы REPEAT ... UNTIL

Формат оператора:

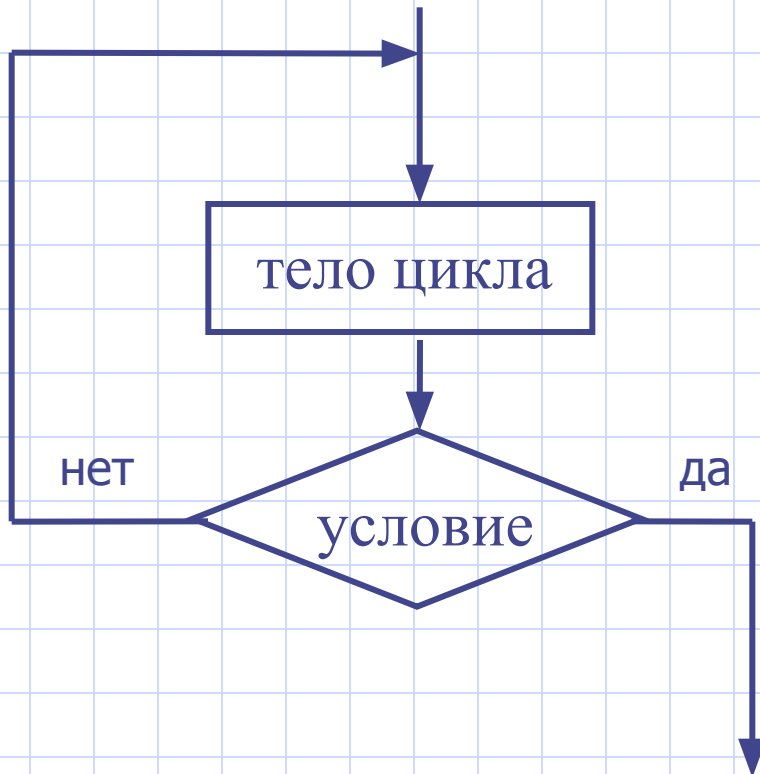
REPEAT {повторять}

<Тело цикла>

UNTIL условие ; {до тех пор, пока не}

Цикл выполняется пока условие не
станет истинным!

Блок-схема



- Тело цикла может содержать любое количество операторов без использования операторных скобок.
- Проверка условия происходит не до, а после очередного выполнения цикла.
- Цикл REPEAT выполняется хотя бы один раз, цикл WHILE может не выполниться ни разу.
- Условие в цикле «до» является условием выхода из цикла, а в цикле пока – условием продолжения цикла.

№7. Найти сумму всех положительных целых чисел, не превышающих данного натурального числа N.

Цикл с постусловием

```
Program Summa2;  
Var S, N, a: integer;  
Begin  
  Writeln('Введите N');  
  Readln(N);  
  a:=1; S:=0;  
  Repeat  
    S:=S+a;  
    a:=a+1;  
  Until a>N  
  Write('сумма =', S);  
  Readln  
End.
```

Цикл с параметром

```
Program Summa3;  
Var S, N, a: integer;  
Begin  
  Writeln('Введите N');  
  Readln(N);  
  S:=0;  
  For a:= 1 to N do  
    S:=S+a;  
  Write('сумма =', S);  
  Readln  
End.
```

Практическая работа

- Программа будет состоять из трех частей, в которых повторяется решение одной и той же задачи с использованием операторов цикла *while*, *repeat* и *for*.
- Наберите и отладьте программу, убедитесь в том, что все три результата **одинаковые**.
- Тест: при $N=10$ $S=55$

Домашнее задание

§12.7, №8

§4.4.3

том 1

- § 4.4.3. (том 1) – учить циклы
- №8 к §12.7 – решить 3-мя способами

Примечание: Вычислить X^N

$P = X^N$; $P = X * X * X * \dots * X$ – N раз

В цикле вычисляется **$P = P * X$** .

Начальное значение **$P = 1$** .