

Циклични алгоритми

1 част



Цикли с неизвестен
в началото брой
повторения

1. Оператор за цикъл с предусловие



A) Общ вид на оператора

while условие **do** оператор

Където:

- while ... do (докато ... изпълнявай) – служебни думи от езика Паскал
- условие – **логически** израз, определящ края на цикъла
- оператор - произволен оператор, най-често съставен, представляващ тялото на цикъла

while условие do оператор



Б) Действие на оператора:

1. Изчислява се лог. израз
2. Ако стойността му е TRUE, се изпълнява тялото на цикъла и се преминава към 1), а ако е FALSE, се преминава към следващия по ред оператор.

! Тялото на този цикъл се изпълнява, докато лог. израз не получи стойност FALSE.

while условие do оператор

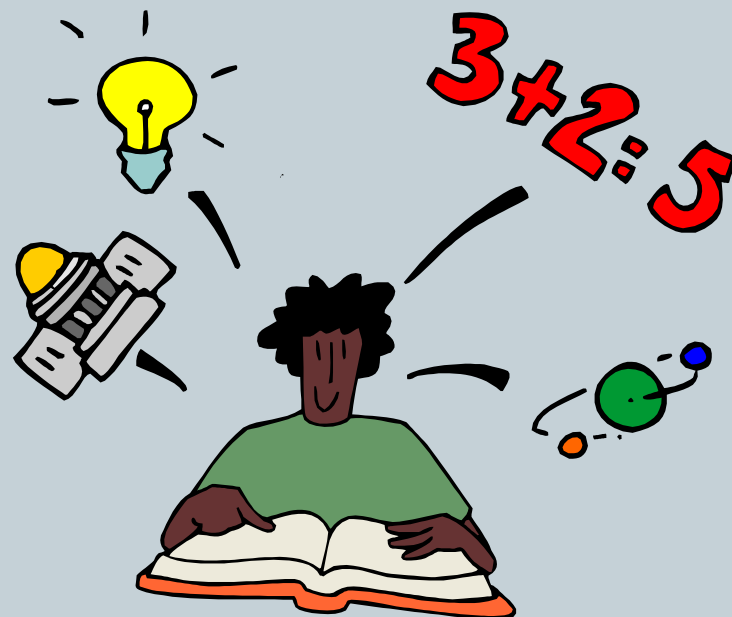


- ! В тялото на цикъла трябва да се променя стойността на поне една променлива, участваща в условието за край. В противен случай, ще се получи безкраен цикъл.



Програма за намиране на НОД по алгоритъма на Евклид

```
program Evklid;  
  var a, b : integer;  
begin  
  write ('a= '); readln (a);  
  write ('b= '); readln (b);  
  while a < > b do  
    begin  
      if a > b then a:= a - b  
        else b:= b - a  
    end;  
  writeln ('NOD e: ',a);  
  readln;  
end.
```



2. Оператор за цикъл с постусловие (следусловие)



А) Общ вид:

repeat

оператор_1;

оператор_2;

... ..

оператор_N

until условие;



Б) Действие

repeat

оператор_1;

оператор_2;

...

оператор_N

until условие;

*Тялото на този цикъл
се изпълнява
задължително поне
веднъж!*

1. Изпълнява се тялото на цикъла.
2. Изчислява се логическия израз (условие) и ако получи стойност FALSE се преминава към т. 1), а ако получи стойност TRUE, се преминава на следващия по ред оператор.

Алгоритъм на Евклид чрез оператор **repeat**



```
program Evklid_2;  
  var a, b : integer;  
begin  
  write ('a='); readln (a);  
  write ('b='); readln (l);  
  repeat  
    if a > b then a:= a - b;  
    if b > a then b:= b - a  
  until a = b;  
  writeln ('NOD e:',a);  
  readln;  
end.
```

Заключение:



- Изборът на подходящия оператор за цикъл (*repeat* или *while*) зависи от задачата и предпочитанията на програмиста.
- Съществена разлика – при ***while*** тялото на цикъла може да не се изпълни, докато при ***repeat*** задължително се изпълнява поне веднъж.
- И при двата оператора в тялото на цикъла трябва да се променя стойността на поне една променлива, участваща в условието за край. В противен случай, ще се получи безкраен цикъл.

Организиране защита на входа:



- За предотвратяване въвеждането на неправилни стойности.
- ...

repeat

write ('Въведете броя на учениците:');

readln (n)

until $n \geq 0$;