

Увод у објектно програмирање:

Пограмске петље / цикличне алгоритамске структуре

Професор: **др Светлана Штрбац-Савић**

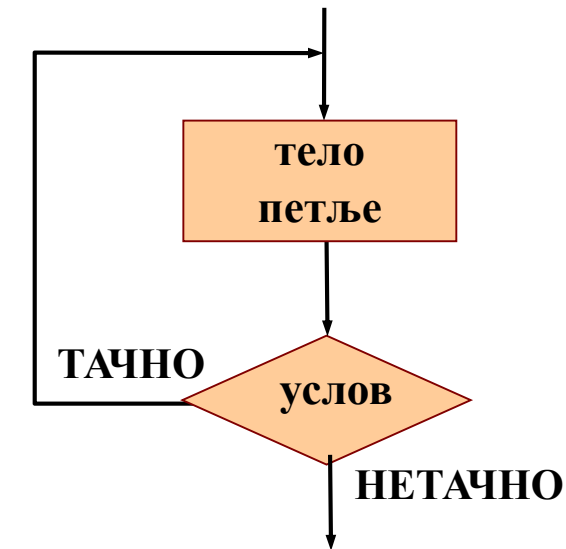
Маил / Кабинет: **svetlanas@viser.edu.rs / 501**

Наредбе циклуса се користе за понављање неке радње или неких радњи (блока наредби) одређени број пута или док услов за излазак из петље није задовољен. На пример лако је наћи максимум два или три броја, а како решити проблем када има више бројева или када се не зна унапред колико ће бројева бити?

Основне врсте циклуса

- петље са условним излазом
(на врху - While-Do,
на дну - Repeat-Until и
у средини - Loop-Exit),
- петље са експлицитним бројачем (For Next),
- петље са имплицитним бројачем (Loop Times),
- бесконачне петље са (Exit Do) и без механизма условног излаза (Do-Loop),
- петље по елементима скупа.

Петље са излазом на дну do...while



Општи облик:

```
do  
    iskaz  
Loop while (izraz)
```

Омогућава реализацију петљи са излазом на дну

1. Извршавају се искази иза do (тело петље)
2. Израчунава се вредност услова
3. Ако је израз ИСТИНИТ иди на корак 1
4. Ако је израз ЛАЖ прекида се извршавање петље и прелази се на следећи исказ иза while

Петља са излазом на дну

Do ... Loop While

и

Do ... Loop Until

Do

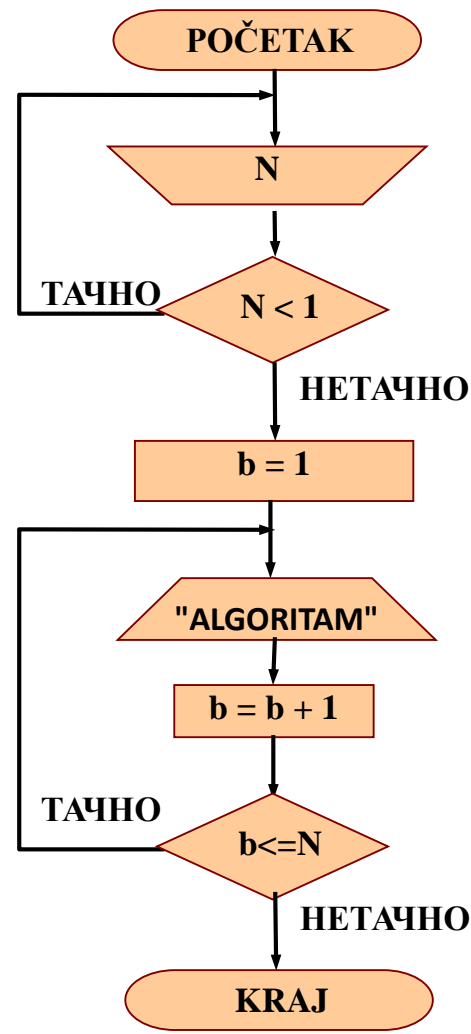
[Blok naredbi]

[Continue Do]

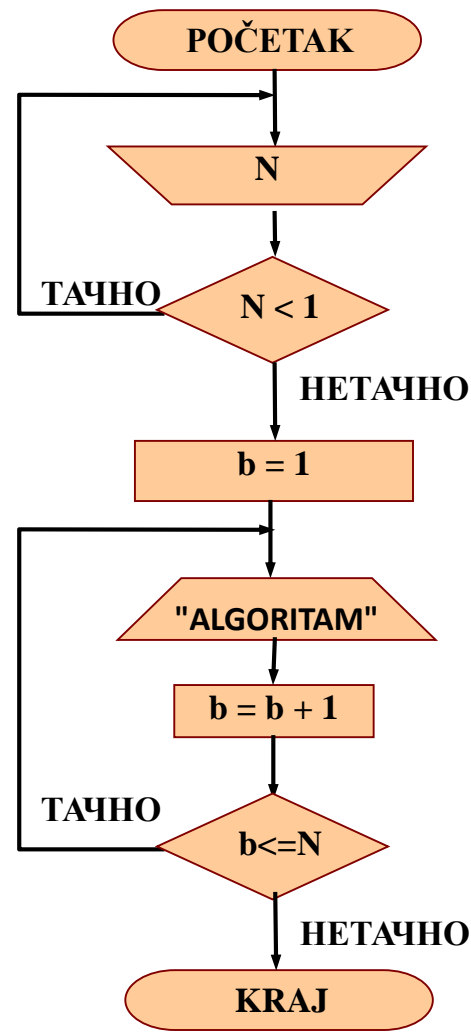
[Exit Do]

Loop {While | Until} Uslov

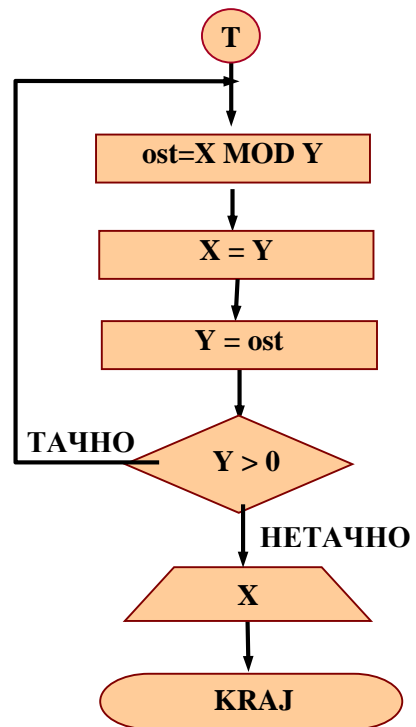
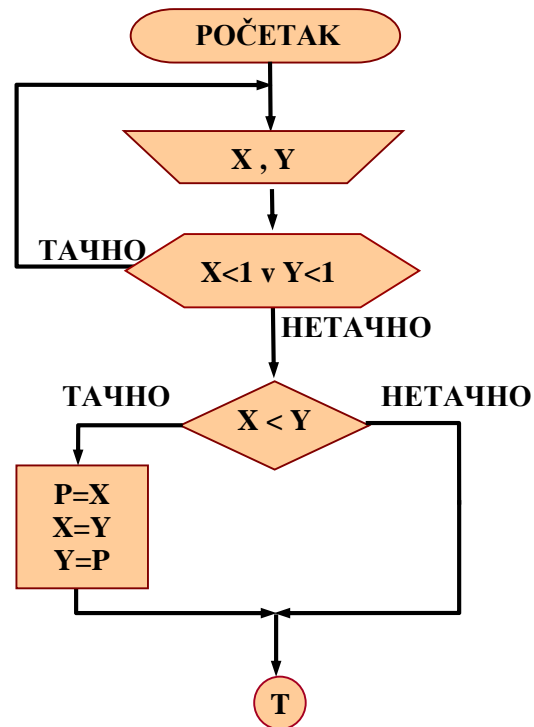
Пример do...while петље - Учитати природан број N и толико пута исписати реч "ALGORITAM".

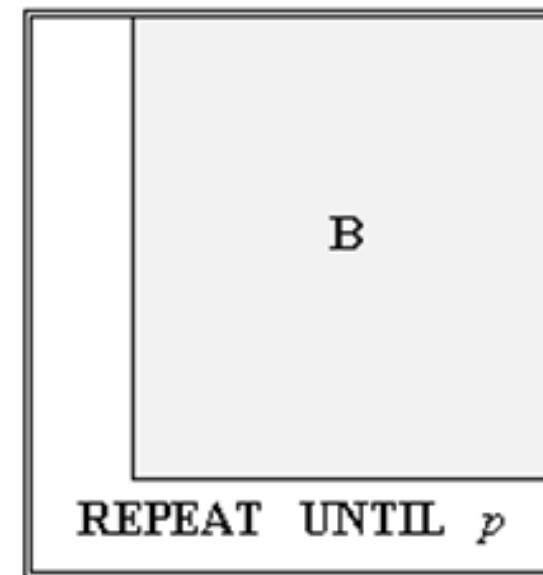
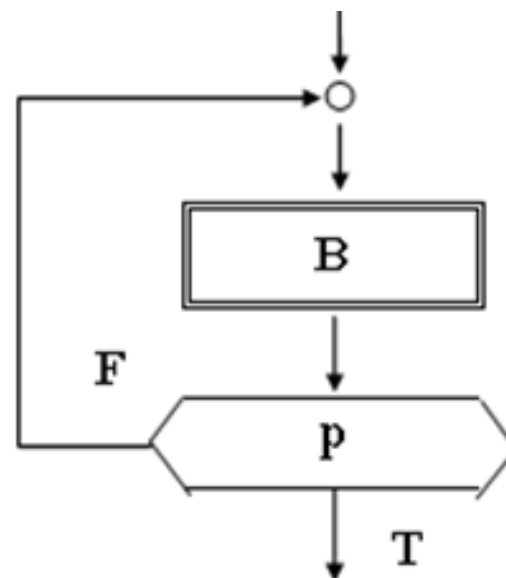


Пример do...while петље - Учитати природан број N и толико пута исписати реч "ALGORITAM".



Пример do...while петље - Еуклидовим алгоритмом одредити NZD два природна броја.





Пример израчунавања x^n

```

s=1
i=n
Do
    s=s*x
    i=i-1
Loop Until i=0
    
```

```

s=x
i=1
Do
    s=s*x
    i=i+1
Loop Until i=n
    
```

```

s=1
If n>0 Then
    i=n
    Do
        s=s*x
        i=i-1
    Loop Until i=0
End If
    
```

Израчунавање x^n

```
Dim i, n, x As Integer
Dim S As Double
S = 1
x = Val (TextBox1.Text)
n = Val (TextBox2.Text)
i = n
Do
    S = S * x
    i = i - 1
Loop Until i <> 0
```

Израчунавање x^n

```
Dim i, n, x As Integer
Dim S As Double
S = 1
i = 0
x = Val(TextBox1.Text)
n = Val(TextBox2.Text)
Do
    S = S * x
    i = i + 1
Loop While i < n
```

- Почетна вредност бројача 0
- Кључна реч **While**
- Услов циклуса $i < n$

```
Dim i, n, x As Integer
Dim S As Double
S = 1
i = 1
x = Val(TextBox1.Text)
n = Val(TextBox2.Text)
Do
    S = S * x
    i = i + 1
Loop While i <= n
```

- Почетна вредност бројача 1
- Кључна реч **While**
- Услов циклуса $i \leq n$

Израчунавање x^n

За $n \geq 1$ ради добро, за $n \leq 0$ се изврши бар једном

```
Dim i, n, x As Integer
Dim S As Double
S = 1
i = 1
x = Val(TextBox1.Text)
n = Val(TextBox2.Text)
Do
    S = S * x
    i = i + 1
Loop While i <= n
```

Petlja sa izlazom na vrhu While ... End While

While *Uslov*

[naredbe]

[Continue While]

[naredbe]

[Exit While]

[naredbe]

End While

Petlja Do While ... Loop, Do Until ... Loop

Do { While | Until } uslov

[naredbe]

[Continue Do]

[naredbe]

[Exit Do]

[naredbe]

Loop

Do

[naredbe]

[Continue Do]

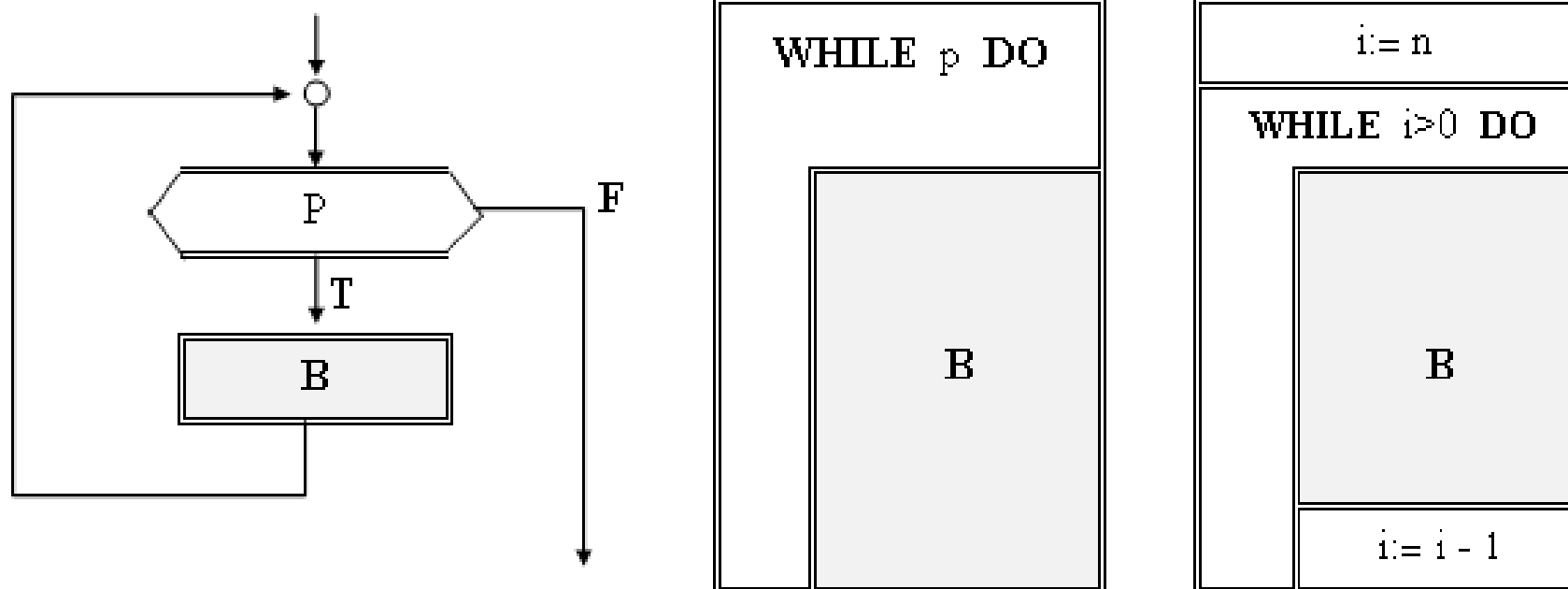
[naredbe]

[Exit Do]

[naredbe]

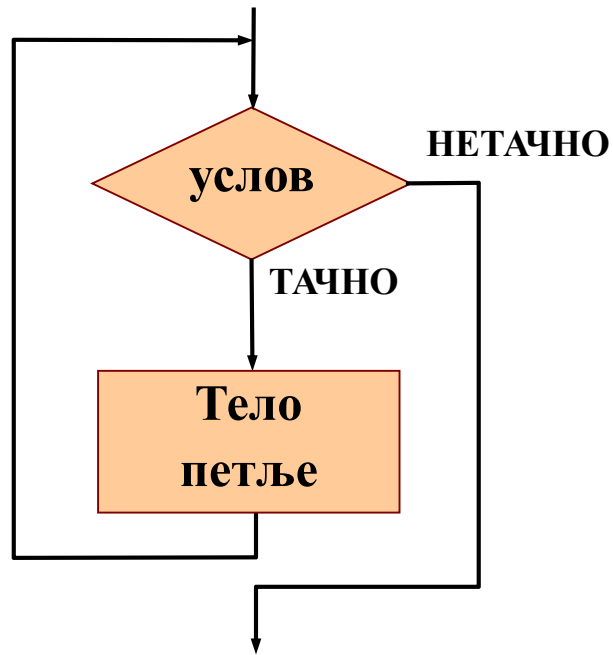
Loop { While | Until } uslov

Стандардни и структурирани дијаграм тока – наредбе са контролом услова на врху



while - петља

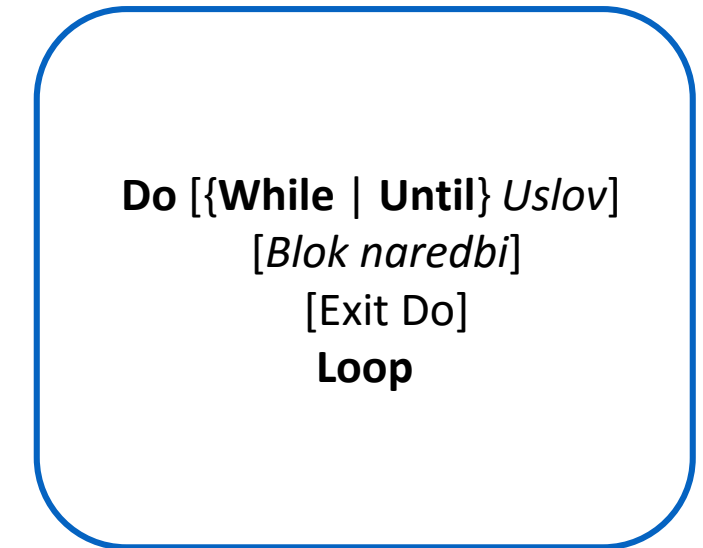
Омогућава реализацију петље са излазом на врху!



Петља ће се понављати све док је услов испуњен (тачан)

(не мора се извршити ни једанпут)

Општи облик

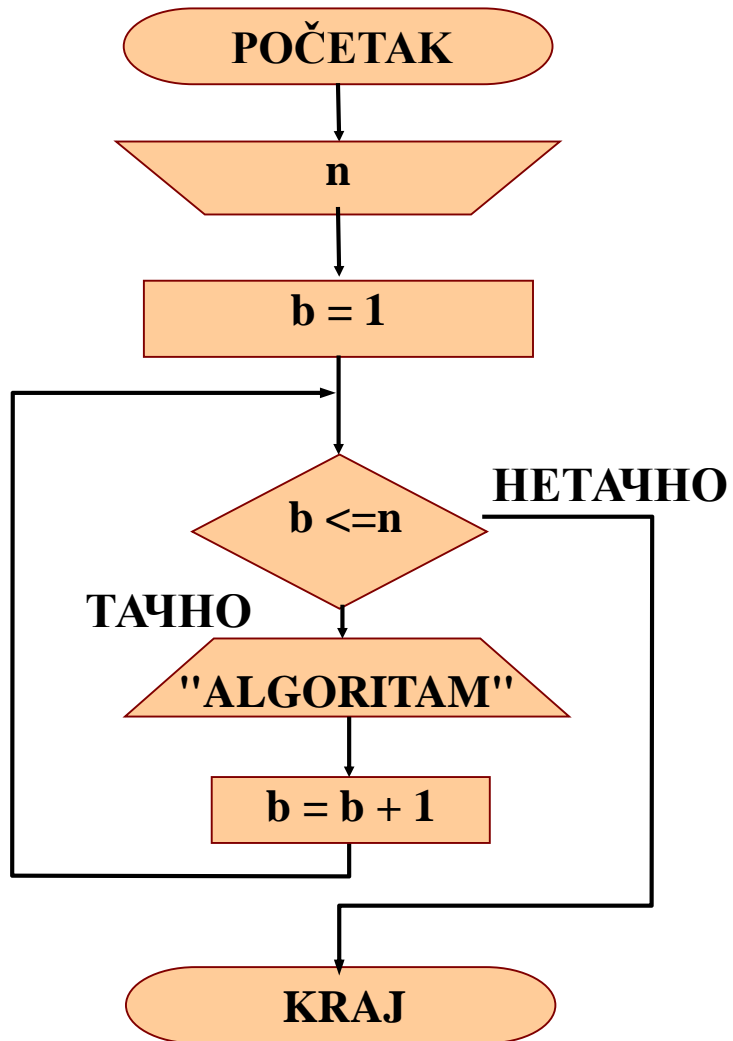


Све док је услов ИСТИНА\ЛАЖ извршава се блок наредби између [{**While** | **Until**} *Uslov*] и **Loop**

Кад **izraz** постане ЛАЖ\ИСТИНА прекида се извршавање петље

Пример while петље

Петља која N пута исписује реч "ALGORITAM".



```
Dim b, n As Integer
```

```
n=Val(TextBox1.Text)
```

```
While b<=n
```

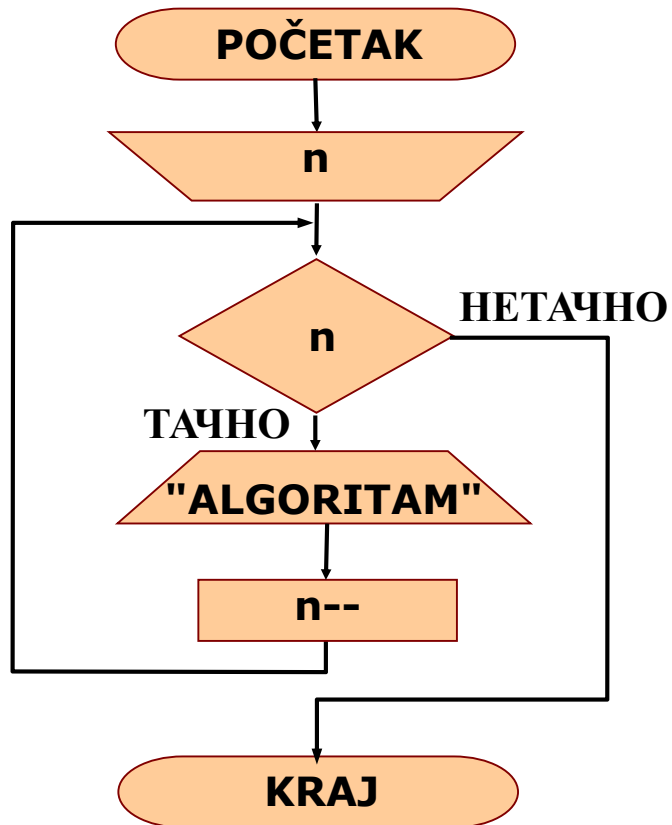
```
txtRezultat.Text= txtRezultat.Text+"Algoritam "
```

```
b++
```

```
End While
```

Пример while петље

Петља која N пута исписује реч "ALGORITAM". Без коришћења помоћног броја.



```
Dim n As Integer
```

```
n=Val(TextBox1.Text)
```

```
While n >= 0
```

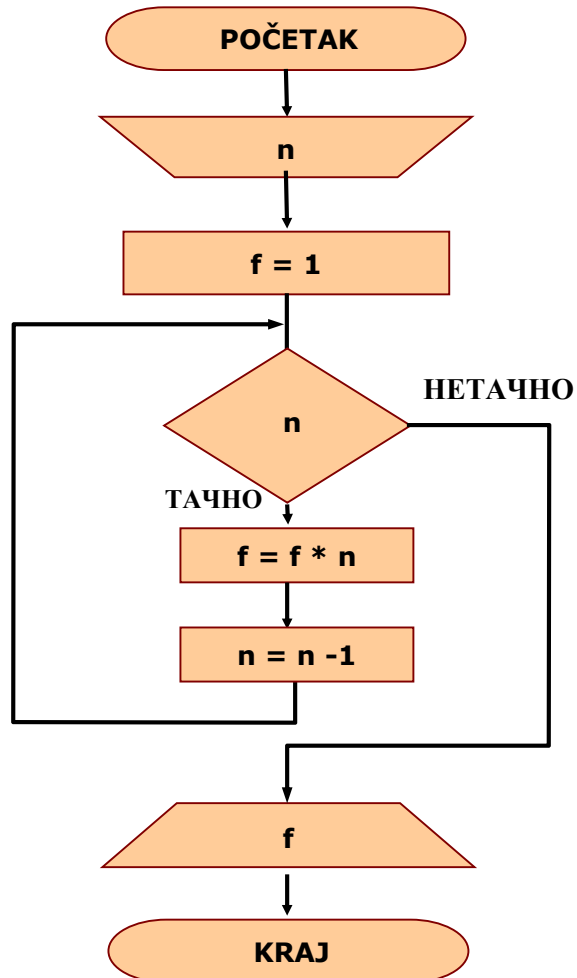
```
    txtRezultat.Text= txtRezultat.Text+"Algoritam "
```

```
    n=n-1
```

```
End While
```

Пример while петље-излаз на врху

Учитати природан број n , а затим израчунати и исписати $n!$

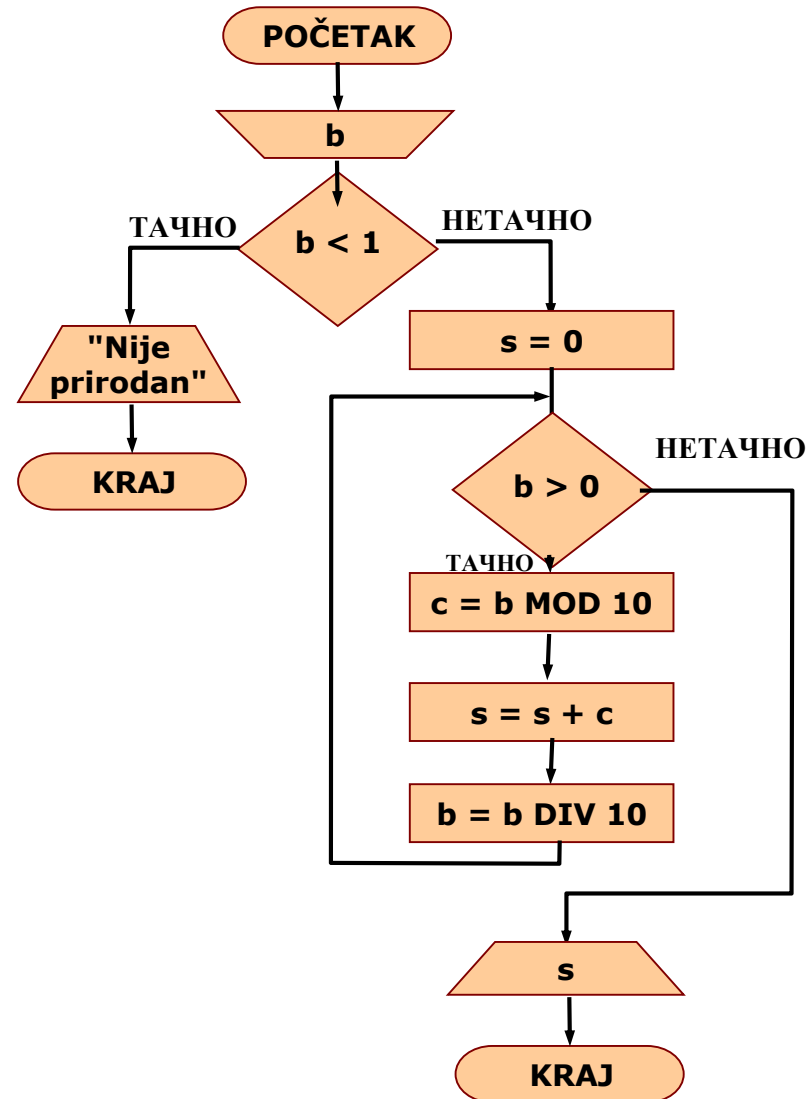


```

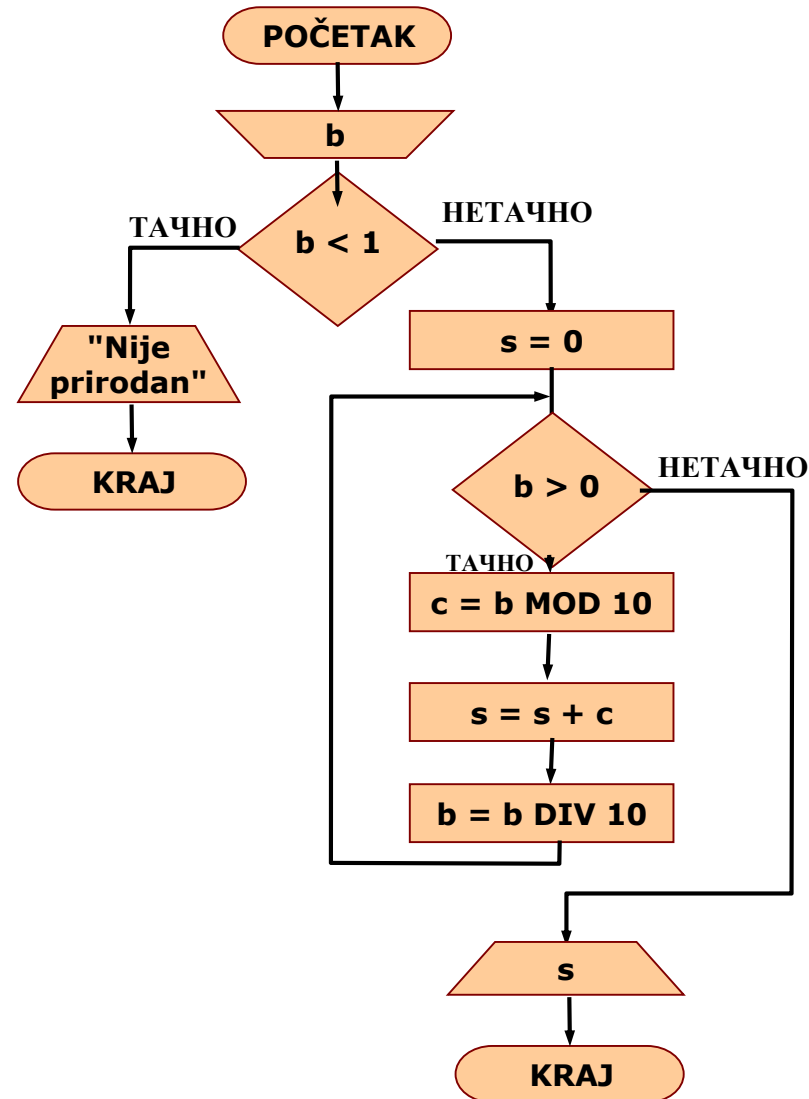
Dim n As Integer
n=Val (TextBox1.Text)
f=1
While n>0
    f=f*n
    n=n-1
End While
txtRezultat.Text= Str(f)
    
```

Напомена: Уништен је почетни податак $n!!!$

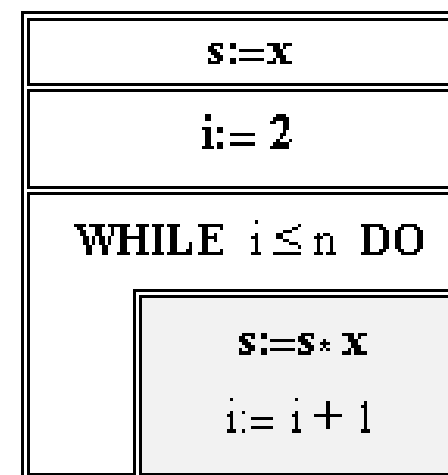
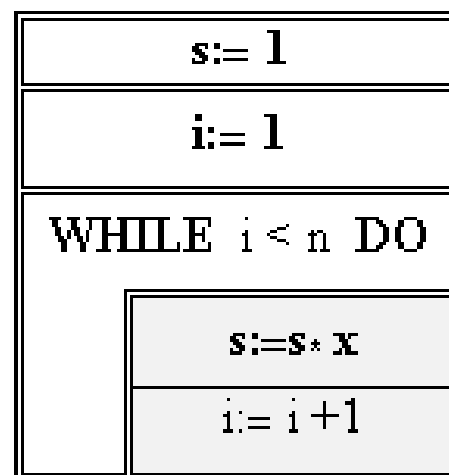
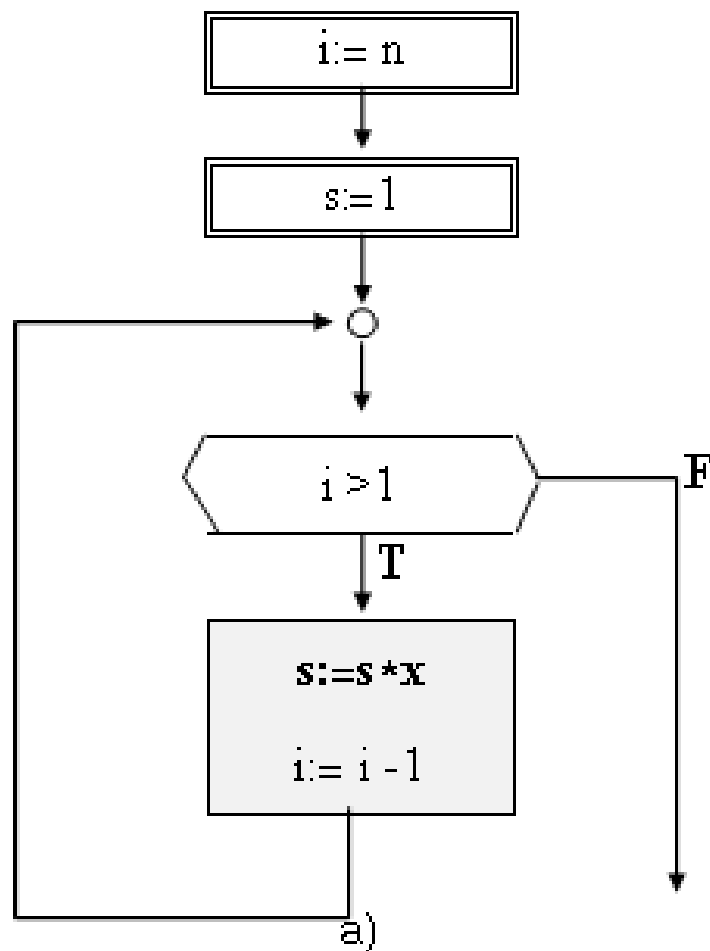
Пример while петље-излаз на врху: Учитати природан број и одредити и исписати збир његових цифара.



Пример while петље-излаз на врху: Учитати природан број и одредити и исписати збир његових цифара.



Израчунавање $s=x^n$

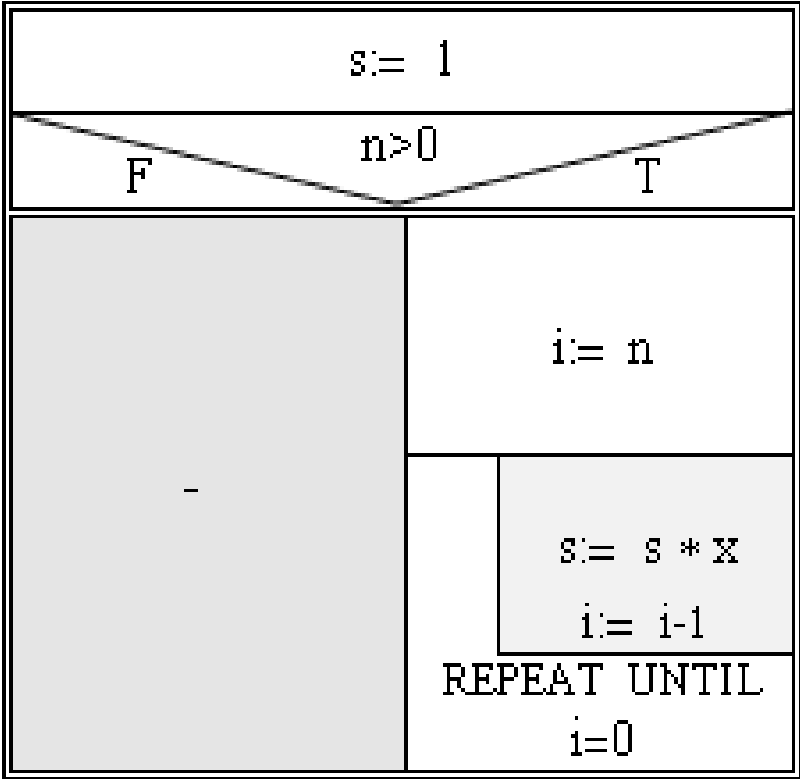


Израчунавање $s=x^n$

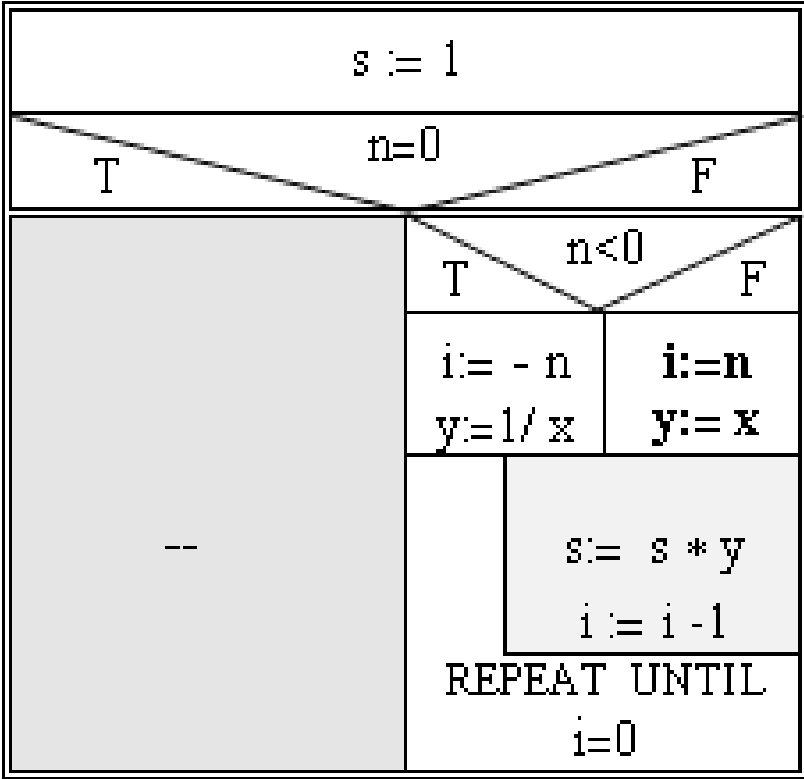
Супротно од \leq , је $>$, а од \geq је $<$ и обрнуто.

<pre> Dim Brojac As Integer Dim Osnova As Single Dim Eksponent As Integer Dim Rezultat As Double Brojac = 1# Rezultat = 1# Osnova = 2# Eksponent = 8 Do While Brojac <=Eksponent Rezultat = Rezultat * Osnova Brojac = Brojac + 1 Loop </pre>	<pre> Dim Brojac As Integer Dim Osnova As Single Dim Eksponent As Integer Dim Rezultat As Double Brojac = 1# Rezultat = 1# Osnova = 2# Eksponent = 8 Do Until Brojac > Eksponent Rezultat = Rezultat * Osnova Brojac = Brojac + 1 Loop </pre>
Uslov za izlaz iz petlje: Brojac <=Eksponent	Uslov za izlaz iz petlje: Brojac > Eksponent

Развој алгоритма за израчунавање x^n корак по корак

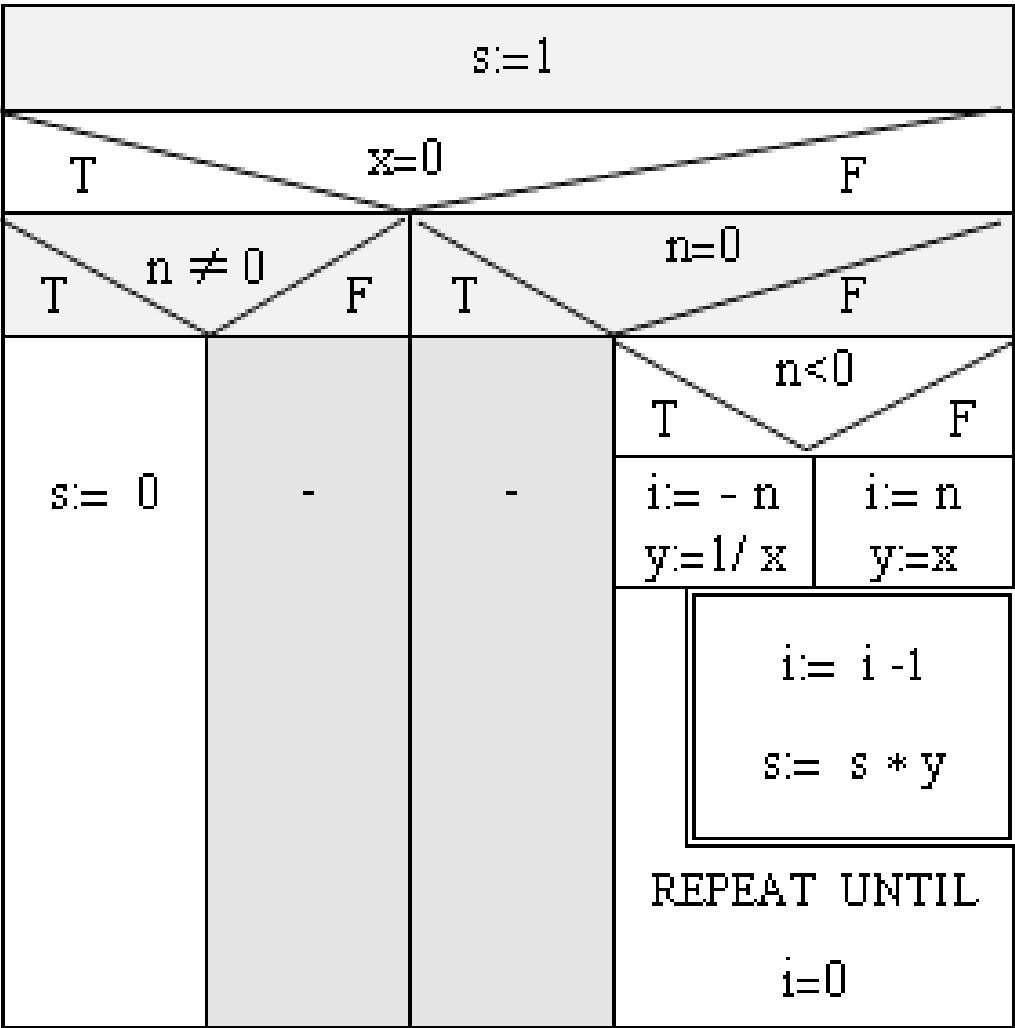


a) sprečavanje pojave beskonačne petlje



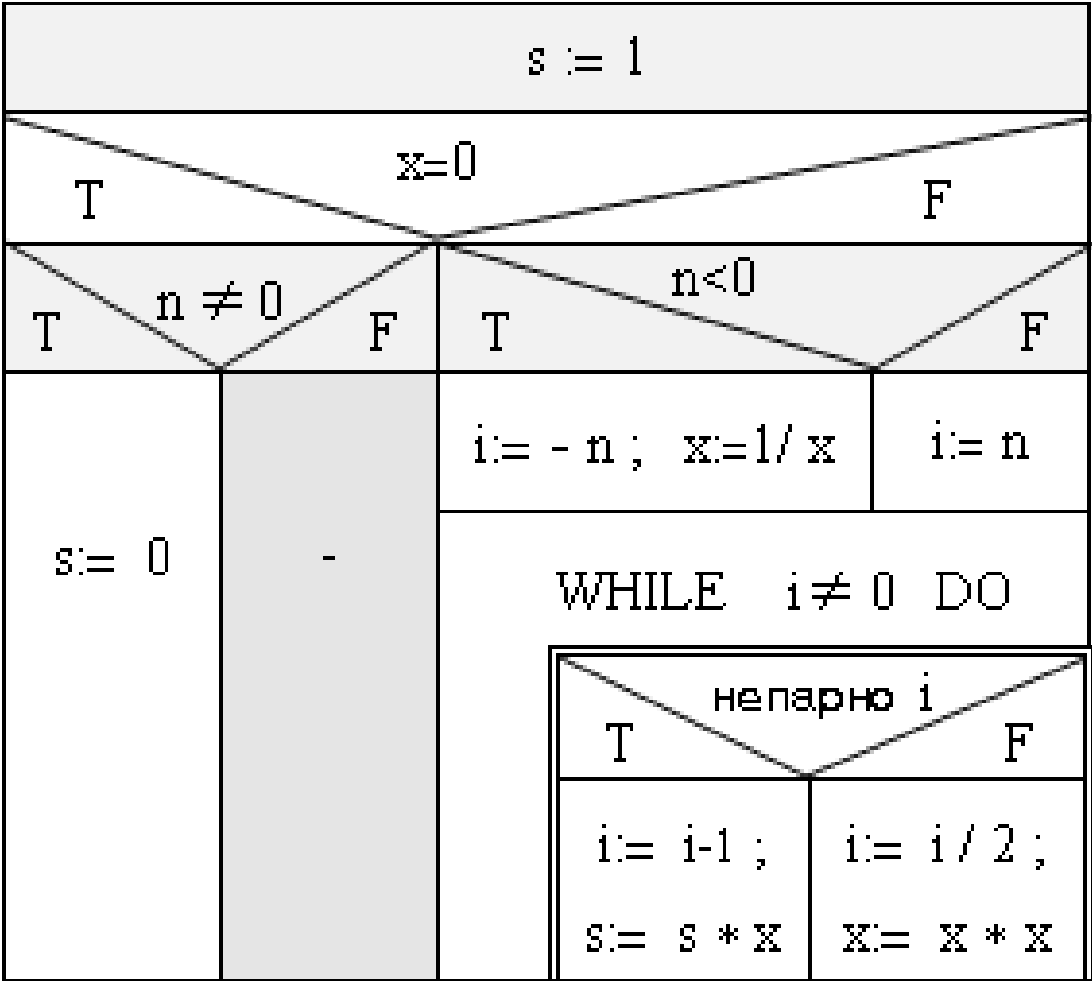
b) algoritam koji radi za sve celobrojne eksponente

Развој алгоритма за израчунавање x^n корак по корак



c) uključivanje provjere vrednosti osnove

Развој алгоритма за израчунавање x^n корак по корак

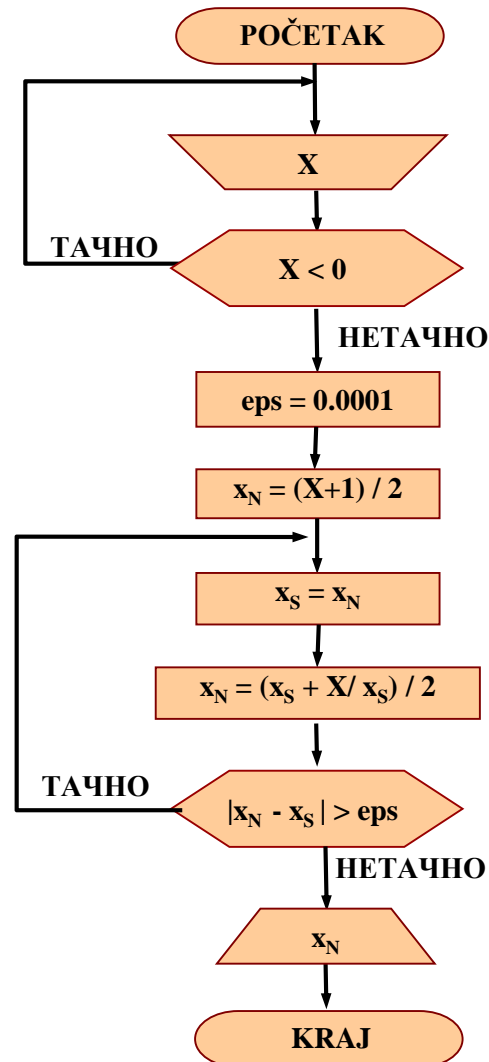


d) optimizacija-ubrzanje obrade

Пример do...while петље: Израчунати, на четири децимале, квадратни корен позитивног реалног броја X , користећи Њутн-Рафсонову итаративну формулу.

$$x_0 = \frac{X+1}{2}, \quad x_{n+1} = \frac{x_n + \frac{X}{x_n}}{2}, \quad n = 0, 1, 2, \dots$$

Do While ... Loop,
Do ... Loop While



Бројачке петље For... Next

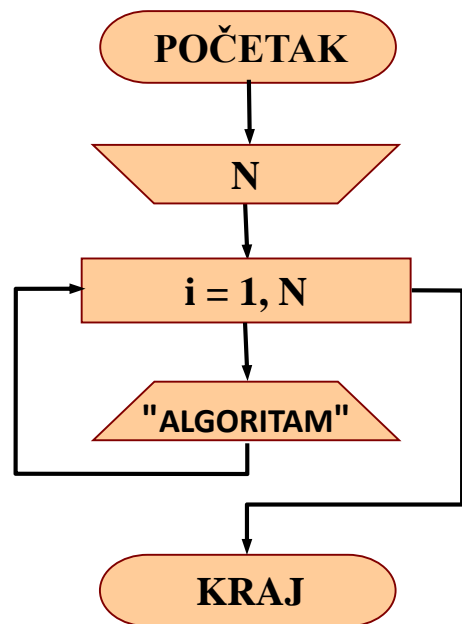
Бројање унапред

```
p=1  
for i=1 to n  
    p=p*i  
next i
```

Бројање уназад

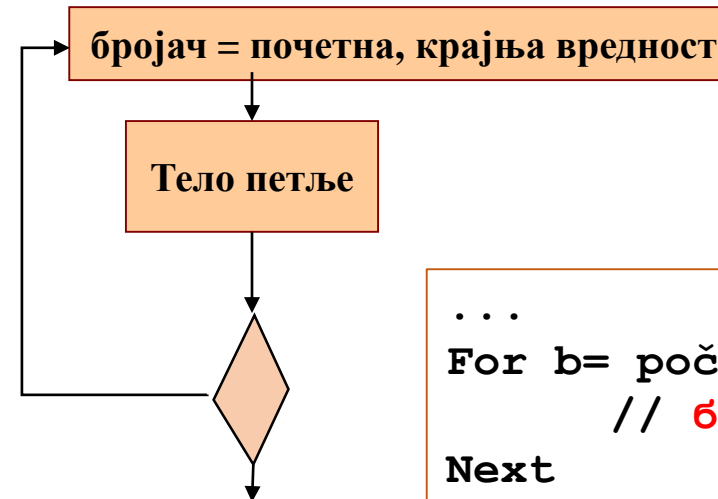
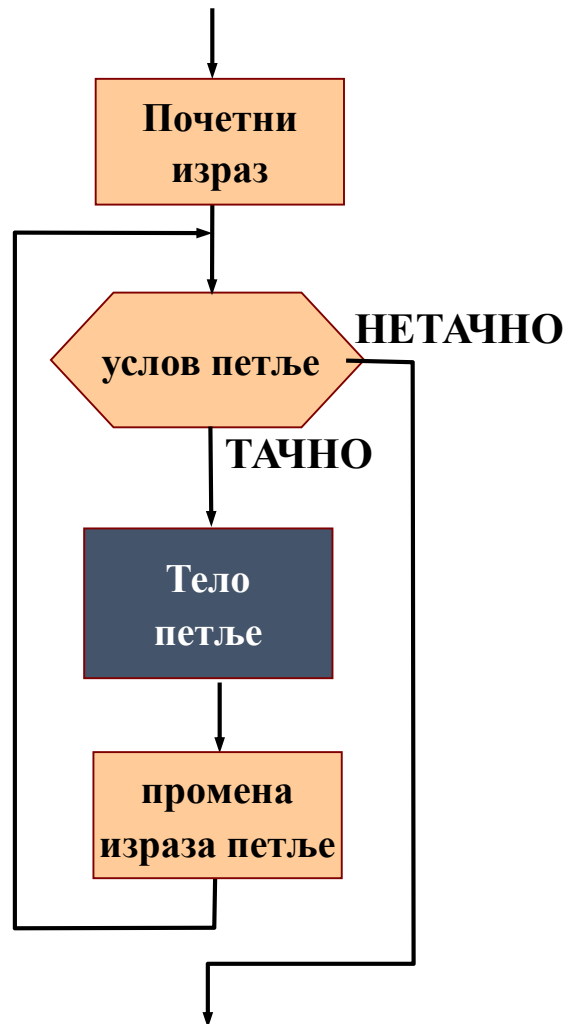
```
p=1  
for i=n to 1 step -1  
    p=p*i  
next i
```

Пример for-петље: Учитати број N и толико пута исписати реч "ALGORITAM".



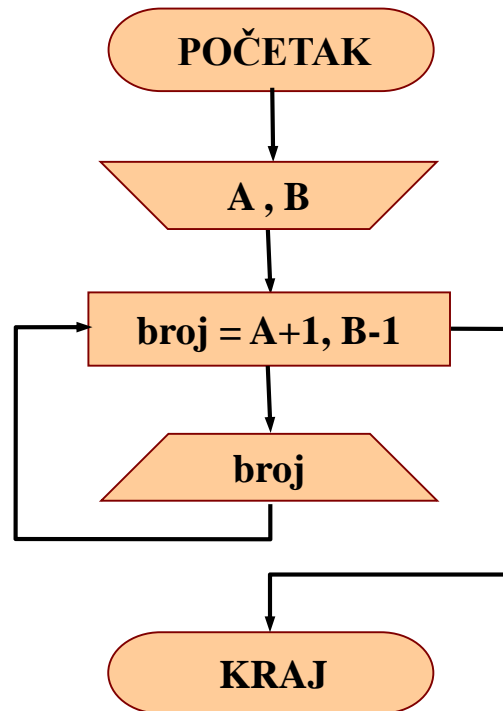
```
For i=1 To N  
    Radni blok  
Next i
```

for-петља са експлицитним бројачем



```
...  
For b= početna To krajnja Step n  
    // блок наредби  
Next  
...
```

Пример for-петље: Учитати границе интервала A,B, и исписати целе бројеве у интервалу <A,B>.



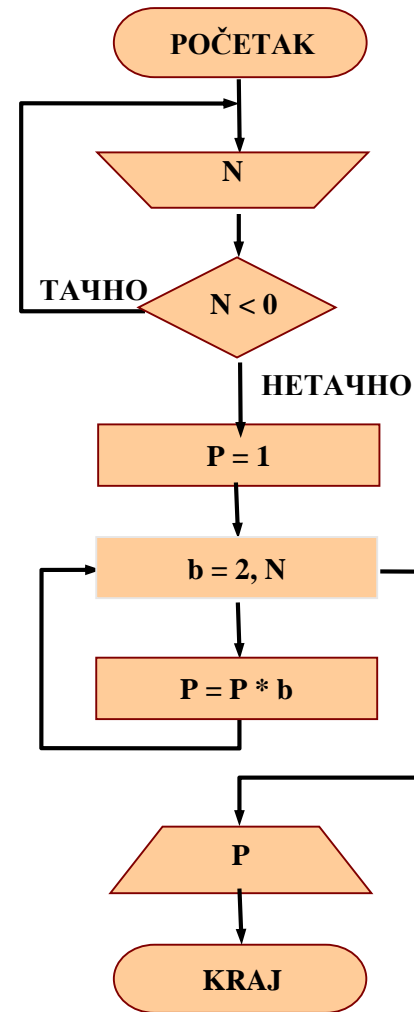
```
Dim A, B, broj As Integer
Unos podataka
For broj=A+1 To B-1 Step 1

Next
```

Пример for-петље (са грешкама): Учитати природни број и израчунати његов факторијел - $n!$

```

...
P=1
For b= 1 To N
    P=P*b
Next
...
    
```



Continue наредба

- **Continue** наредба прекида извршавање циклуса и контролу тока програма враћа на испитивање услова петље, тј извршава петљу за наредну итерацију.
- Може користити у свим претходно помињаним наребама циклуса, с тим што се после кључне речи *Continue* наводи име петље: *Continue Do*, *Continue While*, *Continue For*.

Пример: Написати бројеве из интервала [a, b] дељиве са k

```
broj=a
While broj <= b
    broj = broj + 1
    If Not broj Mod k = 0 Then
        Continue While
    End If
    Console.Write($"{broj}    ")
End While
```

Exit наредба

- **Exit** наредба прекида извршавање циклуса и контролу тока програма враћа на прву наредбу после петље тј. терминира рад петље брз обзира да ли је услов задовољен.
- Може се користити у свим претходно помињаним наребама циклуса, с тим што се после кључне речи *Exit* наводи име петље: *Exit Do*, *Exit While*, *Exit For*.

Пример: Написати први број из интервала $[a, b]$ дељив бројем k

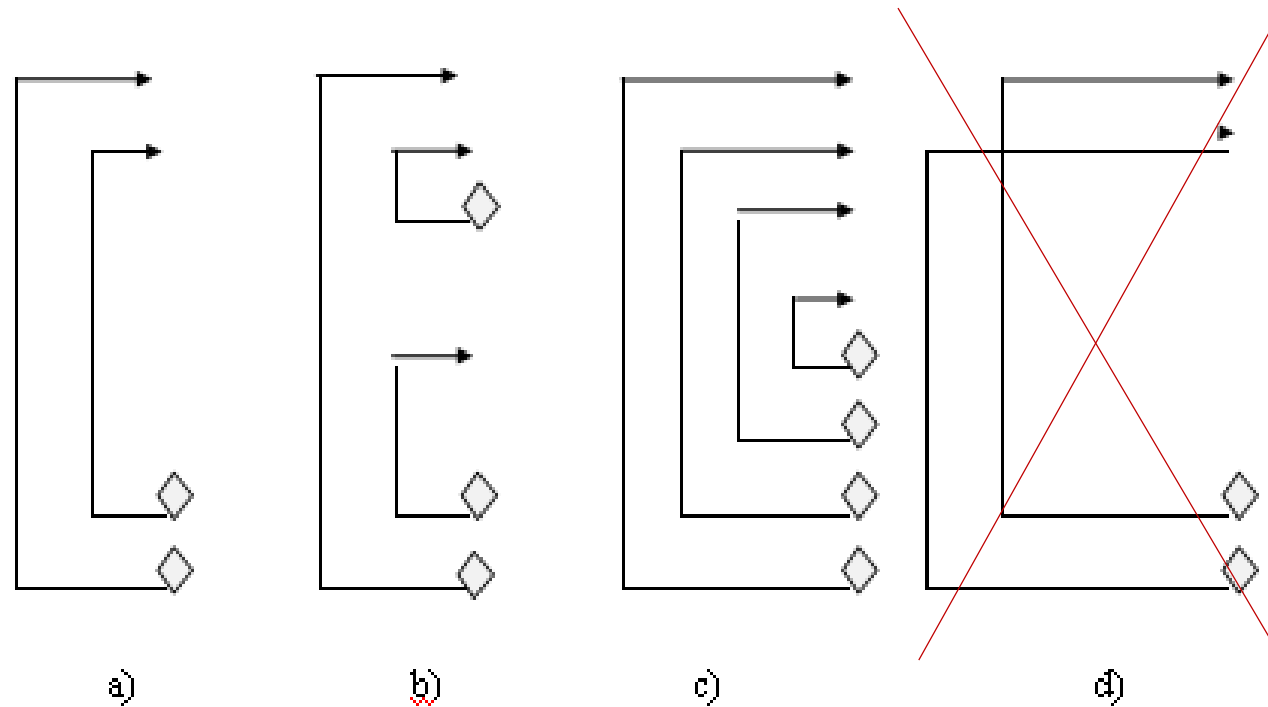
```
...
broj=a
While broj <= b
    broj = broj + 1
    If broj Mod k = 0 Then
        Console.WriteLine($"{broj}    ")
        Exit While
    End If
End While
...
```

Угњеждене петље

- У оквиру једне програмске петље могуће је да се укаже потреба за понављањем неке наредбе или неких наредби више пута. За реализацију тог захтева може се у оквиру једне петље употребити друга петља.

```
S=0
F=1
For i=1 To n
    P=1
    For j= 1 To m
        P=P*(x-i)
    Next
    S=S+P/F
Next
```

Међусобни положај петљи



Примери различитих цикличних структура