

MATLAB

Kontrola toka programa

while petlja

- **while** petlja se koristi kada se unaprijed ne zna koliko puta treba da se izvrši određeni dio koda, ali se zna uslov do kada treba da se izvršava.
- Sintaksa while petlje:

```
while uslov  
    naredbe  
end
```

gdje je uslov logički uslov koji određuje do kad se izvršava petlja.

```
i = 1  
while i < 10  
    disp(i)  
    i = i + 3  
end  
  
1  
4  
7
```

Primjer sa while petljom

- Data je jednakost:

$$\pi^2 = \sum_{n=1}^{\infty} \frac{6}{n^2}$$

Napisati m-fajl koji približno računa vrijednost π^2 koristeći datu sumu. Sumiranje prekinuti kad razlika približne i tačne vrijednosti postane manja od 10^{-5} . Na izlazu ispisati dobijenu vrijednost sume.

```
eps = 1e-5;
n = 1; suma = 0;
while abs(pi^2-suma) > eps
    suma = suma + 6/n^2;
    n = n + 1;
end
disp('Dobijena vrijednost sume je ');
disp(suma);
```

Jedno izvršenje

```
>> piKvadrat
Dobijena vrijednost sume je
```

```
9.8696
```

Drugi primjer sa while petljom

- Napisati funkcijski m-fajl koji za ulazni parametar ima broj **N**. Fajl treba da redom sabira prirodne brojeve (1+2+3+...) sve dok je suma manja od N. Izlazni parametri fajla su dobijeni zbir i broj sabranih brojeva.

```
function [suma, broj] = sabiranje(N)
```

```
% Funkcija koja sabira brojeve
```

Komentar odmah nakon zaglavlja funkcije je help

```
suma = 0;
```

```
n = 0;
```

```
while (suma < N)
```

```
    n = n + 1;
```

```
    suma = suma + n;
```

```
end
```

```
suma = suma - n;
```

```
broj = n - 1;
```

Jedno izvršenje

```
>> help sabiranje
```

```
    Funkcija koja sabira brojeve
```

```
>> [suma, br] = sabiranje(50)
```

```
suma =
```

```
    45
```

```
br =
```

Naredba break

- Izvršenje for i while petlje se može prekinuti naredbom **break**. Nakon naredbe break, prelazi se na prvu naredbu nakon petlje.
- Napisati funkciju `saberidoPrveNule` koja za ulazni parametar ima niz brojeva `X` i koja sumira elemente tog niza sve dok ne naiđe na 0. Funkcija vraća dobijenu sumu i broj sumiranih elemenata.

```
function [suma, broj] = saberidoPrveNule(X)
suma = 0; broj = 0;
for i = 1 : length(X)
    if X(i)==0
        break;
    end
    suma = suma + X(i);
    broj = broj + 1;
end
```

Jedno izvršenje

```
>> X = [1,3,4,3,-2,4,0,6,5];
>> [suma, broj] = saberidoPrveNule(X)

suma =

    13

broj =
```

Naredba error

- Izvršenje programa se može prekinuti naredbom **error**. Ukoliko se aktivira naredba error prekida se izvršavanje programa.

- Primjer: Provjeriti da li je N prirodan broj
N=input('Unijeti prirodan broj: ')

```
if N<0 | round(N)~=N
    error('Nedozvoljen unos');
end
```

Za vježbu

- Napisati funkciju **djeljiv** koja pronalazi sumu svih prirodnih brojeva N , ne većih od broja M , koji imaju osobinu da je N^2+N djeljivo sa 6. Broj M je ulazni parametar fajla.
- Napisati funkciju **parni** koji za ulazni parametar ima niz brojeva X i koja vraća niz koji sadrži samo parne brojeve iz niza X .
- Napisati m-fajl koji izračunava približnu vrijednost broja π koristeći sljedeći razvoj:

$$\pi = 4 - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \frac{4}{11} + \dots$$

Sumiranje prekinuti kada apsolutna vrijednost opšteg člana sume postane manja od broja **eps**, koji se unosi po startovanju fajla.