



Programski paketi u matematici

Programiranje u Matlabu

Naredba if

- if uslov

....

....

end – OBAVEZNO na kraju!

- Jednostavan primer:

```
x = input('Pogodite zamisljeni broj: ');
```

```
if x == 2
```

```
fprintf('Bravo! To je pogodjen broj.\n');
```

```
end
```

If – else -naredba

- if uslov

....

....

else

....

.....

end

Jednostavan primer:

```
x = input('Pogodite zamisljeni broj: ');  
if x == 2  
    fprintf('Bravo! To je pogodjen broj.\n');  
else  
    fprintf('Niste pogodili, ali dobar  
    pokusaj.\n');  
end
```

“Nested” if naredbe – if naredba u drugoj if naredbi
if uslov

....

....

elseif

....

.....

else

.....

.....

end

Primer:

```
if x == 2
```

```
fprintf('Tacno.\n');
```

```
elseif x < 2
```

```
fprintf('Premalo. Pokusajte ponovo.\n');
```

```
else
```

```
fprintf('Preveliko. Pokusajte ponovo.\n');
```

```
end
```

“Nested” verzija prethodnog primera:

```
if x == 2
    fprintf('Tacno.\n');
else
    if x < 42
        fprintf('Premalo. Pokusajte ponovo.\n');
    else
        fprintf('Preveliko. Pokusajte ponovo.\n');
    end
end
end
```

Switch naredba:

```
switch uslov
  case vrednost1
    ....
  case vrednost2
    ...
  otherwise
    .....
end
```


Zadaci:

1. Napisati skript koji obračunava platu radnika. Korisnik unosi broj radnih sati i zaradu po satu. Preko 40 sati rada nedeljno se plaća 50 procenata vise.
2. Napisati program u skriptu koji odredjuje realne korene kvadratne jednačine $ax^2+bx+c=0$. Korisnik unosi a, b i c. Najpre se izračunava diskriminanta i za $D<0$ ispisuje se poruka 'Nema realnih korena', za $D=0$ poruka 'Jednacina ima jedan dvostruki koren jednak:' i ispisuje vrednost korena u sledecem redu i za $D>0$ ispisuje se poruka 'Jednacina ima dva realna korena:' i ispisuje korene u sledećem redu.

3. Napisati skript koji obračunava cenu vožnje taksijem to tako što je cena prvog kilometra 200 dinara. Svaki sledeći kilometar do 10 kilometara je 25 dinara. Svaki kilometar preko 10 km je 10 dinara. Kilometri se zaokružuju na najbliži ceo broj, osim za prvi kilometar, gde mora biti plaćen bar 1 km vožnje (start). Deca mlađja od 18 godina i stariji od 60 godina dobijaju popust 20 %. Korisnik unosi uzrast i dužinu putovanja, a kao informaciju dobija koliko će ga koštati vožnja.

4. Koristeći switch naredbu napisati skript koji unetu dužinu u km, m, dm,cm prebacuje u mm, tako što korisnik unosi jedinicu mere i dužinu u toj jedinici, a skript vraća odgovor kolika je dužina u mm.

5. Napisati skript koji sortira u rastućem redosledu niz od tri elementa koji unosi korisnik.

6. Napisati skript u kome korisnik unosi redni broj dana u nedelji, počev od nedelje kao prvog dana i zatim se koristeći if elseif naredbu ispisuje poruka koji dan je u pitanju i da li je radni dan ili vikend. Na primer ako korisnik unese 2 treba da dobije poruku „Ponedeljak - radni dan.“, ako unese 7 „Subota - neradni dan.“, a ukoliko ne unese broj između 1 i 7 poruku „Broj mora biti od 1 do 7.“

Petlje

- Omogućavaju da se komanda ili grupa komandi izvrši više puta za redom
- Već smo se sretali sa petljama:
>> x = 1:5
>> suma = sum(x)
- Prethodna naredba zapravo se izvršava u petlji

For petlja

for indeks = prvi : korak : poslednji
grupa Matlab komandi
end

- Prethodni primer sa for petljom:
suma = 0;
for n = 1:5
suma = suma + n;
end
fprintf('suma iznosi %i \n',suma);

For petlja - primeri

- `for i = 1:2:9, disp(i), end`
- `for i = 10:-2:3, disp(i), end`
- `for i = 0:0.5:4, disp(i^2), end`
- `for i = [25 9 81], disp(sqrt(x)), end`
- `for i = 3:0, disp(i), end`

Još jedan primer for petlje:

```
lista = rand(1,5); % može i bez liste
for x = lista % for x = rand(1,5)
if x > 0.5
fprintf('Slucajan broj %f je velik.\n',x)
else
fprintf('Slucajan broj %f je mali.\n',x)
end
end
end
```


Komande break i continue

- break – obično se umeće unutar uslovnog iskaza i tako se prekida izvršavanje petlje ili skripta ako se ispuni određeni uslov. Može se upotrebljavati i unutar petlji.
- continue - kad Matlab naidje na ovu naredbu u petlji ili uslovnom iskazu ne izvršava preostale naredbe u petlji već prelazi na komandu end na kraju te petlje ili uslovnog iskaza.

Zadaci:

- Izračunati sumu $\sum_{k=1}^n \frac{1}{k^2}$ sa for petljom i bez petlje, gde n unosi korisnik.
- Šta je rezultat sledećeg bloka komandi:

```
N = 7;  
for m = 1:N  
    for n = 1:m  
        fprintf('*');  
    end  
    fprintf('\n');  
end
```

Izbegavanje petlji vektorizacijom

- Napraviti 500 x 500 matricu ciji su svi elementi jednaki pi.

```
for i =1:500
    for j =1:500
        A(i,j)=pi;
    end
end
```

Isti zadatak bez petlji:

`A = pi*ones(500,500);`

- Uporediti vremena potrebna za resenja sa i bez petlji, koristeći naredbu `etime(t0,t1)` ili `tic toc` naredbu. Pogledati `help` za više informacija o ovim naredbama.

Zadaci:

- Izračunati sumu $\sum_{k=0}^n \frac{(-1)^k}{2^k} k$ sa for petljom i bez petlje, gde n unosi korisnik.
- Napisati program u skriptu koji odredjuje i ispisuje n-ti Fibonačijev broj, gde n unosi korisnik. Fibonačijevi brojevi su 1,1,2=1+1,3=1+2,5=2+3,...f(n) =f(n-2)+f(n-1)
- Koristeći for petlju napisati program u skriptu koji iscrtava grafike funkcije f(x)=cos(nx) za n = 1,2,3,4,5,6 i 0 <=x<=2*Pi.
- Korisnik unosi dva niza x i y. Koristeći for petlju kreirati matricu A čiji elementi su definisani kao $A_{ij} = x_i y_j$.

While petlja

while uslovni izraz

grupa Matlabovih komandi

end

- Koristi se kada nije unapred poznat broj prolazaka kroz petlju
- Prestaje da prolazi kroz petlju kada uslovni izraz postane netačan
- Pažnja: beskonačna petlja

Primer while petlje

- Izračunati sumu prvih n prirodnih brojeva manju od unapred date granice limit.

```
suma = 0;
```

```
n = 0;
```

```
while suma <= limit
```

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

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

```
end
```

```
fprintf('suma : %i broj: %i \n', suma, n);
```

Zadaci:

1. Koristeći Tejlorov polinom funkcije $\sin(x)$

$\sum_{k=0}^{\infty} \frac{(-1)^k x^{2k+1}}{(2k+1)!}$ izračunati sa tačnošću od

0.0001 vrednost $\sin(x)$ gde je x vrednost ugla u stepenima koji unosi korisnik.

2. Napisati program koji određuje kusur koji kasa za samousluživanje treba da vrati korisniku. Program generiše cenu kao slučajanu celobrojnu veličinu manju od 2000. Zatim mašina pita korisnika kojom novčanicom plaća – i to mora biti jedna novčanica iznosa 2000, 1000, 500, 200, 100,50, 20, 10, 5, 2 ili 1 dinar. Ako je uneta novčanica manja od cene ispisuje se odgovarajuća poruka. Ako je novčanica dovoljna, program izračunava kusur i ispisuje kojim novčanicama se kusur isplaćuje, tako da se upotrebi najmanji mogući broj novčanica.