

Osnovi računarstva II

Čas 7

Miloš Daković

Elektrotehnički fakultet – Podgorica

29. mart 2021.

Funkcijski fajlovi

- Funkcijske fajlove koristimo kada je Octave/MATLAB okruženje potrebno proširiti novim funkcijama.
- Funkcijski fajlovi su slični skript-fajlovima uz nekoliko razlika:
 - Svaki funkcijski fajl se izvršava u sopstvenom radnom prostoru. Svaka varijabla koja se koristi u funkcijskom fajlu, je nezavisna od osnovnog radnog prostora MATLAB-a (Octave).
 - Sve ulazne podatke funkcijski fajl dobija preko svojih ulaznih argumenata, a kroz izlazne argumente vraća rezultate.
- Octave i MATLAB imaju mnogo ugrađenih funkcija (na primjer trigonometrijske, logaritamske, funkcije za sortiranje, za zaokruživanje realnih brojeva...). Pored toga, često se funkcije vezane za neku oblast distribuiraju u obliku toolbox-ova (na primjer funkcije za rad sa signalima, slikama, neuralnim mrežama...)

Funkcijski fajlovi – kreiranje

- Funkcijski fajl počinje ključnom riječju **function**.
- Poslije toga se navodi lista izlaznih argumenata (**koriste se uglaste zagrade ukoliko ih je više**).
- Dalje stavljamo znak jednakosti i naziv fukcije (**mora biti isti kao i naziv funkcijskog fajla**)
- Na kraju navodimo listu ulaznih argumenata (**navodi se u običnim zagradama a argumenti se odvajaju zarezima**).
- Unutar funkcijskog fajla pišemo komande kojima je cilj da pošavši od ulaznih vrijednosti dobijemo izlazne vrijednosti funkcije.
- Primjer: funkciju sa nazivom **projekcija**, koja ima tri ulazna argumenta: x , y i z , i dvije izlazne vrijednosti: a i b pišemo u funkcijskom fajlu sa nazivom **projekcija.m** i u prvoj liniji fajla stavljamo:

```
function [a,b] = projekcija(x,y,z)
```

Funkcijski fajlovi – korišćenje

- Funkcijski fajl koristimo na isti način kao i ugrađene Octave/MATLAB funkcije. Navedemo ime fajla i u zagradi stavimo argumente. Ukoliko fajl vraća više od jedne izlazne vrijednosti, tada sve izlaze smještamo u varijable.
- Ukoliko odmah poslije definicije funkcijskog fajla stavimo nekoliko linije komentara, tada će se te linije ispisati u komandnom prozoru komandom **help naziv_funkcije**.
- Podsjetimo se načina pozivanja nekih ugrađenih funkcija:

```
y = sin(pi/6) + cos(15*pi/180)
```

```
B = sum(1:2:21) + prod([2, 3, 5, 8])
```

```
[M, p] = max(A)
```

(dva izlazna argumenta)

```
m = min(A)
```

(jedan izlazni argument)

```
size(A)
```

(bez izlaznih argumenta)

Funkcijski fajlovi: primjer

- Korisnički definisana funkcija može izgledati ovako:

```
function R = veza(R1,R2)
% Paralelna veza otpornika R1 i R2
R = (R1*R2) / (R1+R2);
```

- Ova funkcija uzima dvije ulazne vrijednosti, R_1 i R_2 i računa izlaznu vrijednost kao paralelnu vezu otpornika R_1 i R_2 .
- Napisanu funkciju **veza** možemo koristiti ovako:

```
a = veza(100,20)
r5 = veza(r1,r2) + veza(r3,r4)
help veza
```

- Ovu funkciju možemo pozivati samo ako se nalazimo u istom folderu gdje je ona definisana. Ukoliko imamo potrebu da definišemo funkcije koje će nam uvijek biti dostupne, moramo ih smjestiti u specijalne foldere.

Funkcijski fajlovi: nargin i nargout

- Unutar funkcijskog fajla dostupne su nam dvije varijable sa specijalnim nazivima.
- Varijabla **nargin** u sebi sadrži broj ulaznih argumenata zadatih prilikom poziva funkcije.
- Varijabla **nargout** u sebi sadrži broj izlaznih vrijednosti koje se of funkcije traže.
- Korišćenjem ovih varijabli možemo modifikovati ponašanje funkcije u skladu sa zahtjevima definisanim prilikom poziva funkcije.
- U funkcijskim fajlovima treba izbjegavati ispile međurezultata. Izuzetak su komande **error** i **warning** kojima možemo informisati korisnika o eventualnim problemima.
- Bilo gdje u tijelu funkcije možemo navesti naredbu **return** kojom se funkcija završava i kontrola vraća pozivajućem programu.

Funkcijski fajlovi: primjer 2

```
function y = primjer2(x,N)

if nargin==1                                % ako je zadato samo x
    N = 20;                                  % definišimo N ovako
end

s = 1;
for k=1:N
    s = s + (-abs(x)).^k/prod(1:k);
end

if nargin==0 && length(x)>1               % kada je ovo tačno?
    plot(x, s)                             % nacrtajmo grafik
else
    y=s;                                     % u suprotnom vratimo rezultat
end
```

Funkcijski fajlovi: primjer 2

- Izračunajte vrijednosti $\text{primjer2}(1)$, $\text{primjer2}(1, 40)$ i $\text{exp}(-1)$. Uporedite dobijene rezultate.
- Ako je x niz vrijednosti od -5 do 5 sa korakom 0.1 , nacrtajte, na jednom grafiku, funkcije: $y1=\text{primjer2}(x)$, $y2=\text{primjer2}(x, 10)$ i $y3=\text{exp}(-\text{abs}(x))$.
- Napišite varijantu funkcije primjer2 kod koje se umjesto varijable s koristi varijabla y . Analizirajte ponašanje takve funkcije u slučajevima kada se ne traži izlazni argument.
- U komandnoj liniji otkucajte:
 $3*\text{primjer2}(0:0.1:1)-\text{primjer2}(0.5)$
Zašto se ne crta grafik?

Funkcijski fajlovi: završne napomene

- Ako programiramo klasičnu matematičku funkciju (koja liči na funkcije $\sin(x)$, e^x , $|x|$) korisno je obezbijediti da se i naša funkcija ponaša na „očekivan“ način ukoliko joj korisnik prosljedi niz ili matricu kao ulazni argument.
- Naša funkcija bi trebala da se snađe i sa neočekivanim ulaznim podacima i da, na primjer, korisnika upozori na mogući problem. Analizirajte šta će se desiti ako ranije definisanu funkciju **veza** pozovemo ovako: **veza(0, 0)** ili čak ovako: **veza(-80, 30)**. Pokušajte da modifikujete funkciju tako da „ispravno“ reaguje u ovakvim slučajevima.
- Komentar nakon definicije funkcije (help dio) treba biti detaljan. To će biti od velike koristi svima koji koriste i pozivaju funkciju. Uobičajeno je da se nakon help komentara, ostavi jedna prazna linija pa se u nastavku (ponovo kao komentar) upiše ime autora, datum, kontakt adresa, pravila o korišćenju (licenca), ...

Funkcijski fajlovi: zadaci

- Napišite funkciju koja ima dva ulazna argumenta x i y i jedan izlazni argument f . Funkcija računa izlaznu vrijednost f kao:

$$f(x,y) = \begin{cases} 0 & \text{za } x^2 + y^2 \geq 1 \\ 1 - \sqrt{x^2 + y^2} & \text{za } x^2 + y^2 < 1 \end{cases}$$

- Napišite funkciju koja kao argument uzima niz brojeva x . Izlazna vrijednost je zbir kvadrata pozitivnih elemenata ulaznog niza x .
- Napišite funkciju koja ima dva ulazna argumenta: nizove brojeva x i y . Elementi nizova x i y su uređeni u neopadajući poredak. Izlazni argument je niz brojeva z u koji su uključeni svi elementi nizova x i y , uređeni u neopadajući poredak. Ako se od funkcije traže dvije izlazne vrijednosti, tada se druga vrijednost računa kao zbir kvadrata elemenata niza z .