Vektorska reprezentacija dokumenta

20. oktobar 2015

Stringovi u MATLAB-u se navode korištenjem jednostrukih navodnika ili apostrofa (’). String u MATLAB-u se ponaša kao niz znakova, što znači da je indeksiranjem niza moguće pristupiti pojedinim znakovima, odnosno, podstringovima. Pri ovome je potrebno voditi računa da u MATLAB-u indeksiranje nizova počinje od 1. Dužina stringa se može odrediti korištenjem funkcije `length`. Na primjer,

```
>> s = 'string';
>> length(s)

ans =

     6

>> s(2)

ans =

t

>> s(2:5)

ans =

trin

>> s(3:end)

ans =

ring
```

Pošto je string predstavljen kao niz znakova, ukoliko želimo da kreiramo niz stringova, dobili bismo dvodimenzionalnu matricu. Međutim, u ovom slučaju svi stringovi bi morali biti jednake dužine, što je rijetko. Umjesto toga koristićemo ćelijski niz. Ćelijski niz je sličan običnim nizovima, odnosno, vektorima u MATLAB-u, ali elementi ćelijskog niza mogu pripadati proizvoljnoj klasi i ne moraju čak ni svi elementi jednog niza biti iz iste klase. Naravno, elementi ćelijskog niza mogu biti i stringovi proizvoljnih dužina. Ćelijski niz se navodi i indeksira korištenjem velikih zagrada:

```
>> niz = {'eci', 'peci', 'pec'}

niz =

    'eci'    'peci'    'pec'

>> niz{2}

ans =

peci

>> length(niz)

ans =

     3
```

U ovoj vježbi će biti korisno definisanje praznog ćelijskog niza

```
niz = {};
```

i dodavanje elementa na kraj postojećeg niza

```
niz{end+1} = 'string';
```

U ovom slučaju se niz proširuje za jedan element i to onaj koji je dodat.

MATLAB ima bogatu biblioteku funkcija za rad sa stringovima čijoj se dokumentaciji može pristupiti pomoću

```
help strfun
```

U ovoj vježbi koristićemo funkcije `strfind(text, pattern)` i `strcmp(str1, str2)`. Prva funkcija vraća startne indekse pojava stringa `pattern` u stringu `text`, a druga vraća logičku 1 kada su stringovi koji se porede jednaki. Uočite

da jedan ili oba argumenta funkcije `strcmp` mogu biti ćelijski nizovi. Od značaja je i funkcija `strtok` koja tokenizuje string korištenjem zadatih znakova kao granica riječi.

Tekst se može pročitati iz fajla pomoću funkcije `fscanf` čija je sintaksa slična kao u programskom jeziku C. Pomoću

```
sadrzaj_fajla = fscanf(fid, '%c', inf);
```

moguće je sadržaj čitavog tekstualnog fajla unijeti u promjenljivu `sadrzaj_fajla`. U prethodnoj naredbi `fid` je referenca na otvoreni fajl koju vraća funkcija `fopen`. Fajl se zatvara funkcijom `fclose`.

Ukoliko se želi obraditi svaki element ćelijskog niza mogu se koristiti `for` i `while` petlje u MATLAB-u.

Zadaci

1. Kreirajte promjenljivu `str` koja sadrži string 'zdravo'.
2. Kreirajte ćelijski niz `str_niz` koji sadrži sljedeće stringove: 'zdravo', 'zbogom', 'zdravo', 'zbogom', 'zdravo', 'zdravoratumski'. Svaki element niza treba da sadrži jedan od navedenih stringova.
3. Napišite kod koji broji koliko puta se string 'zdravo' javlja u ćelijskom nizu `str_niz`.
4. Napišite kod koji broji koliko puta se riječ 'zdravo' javlja u ćelijskom nizu `str_niz`.
5. Napišite kod kojim će u ćelijskom nizu `str_niz` pronalaziti string najsličniji riječi 'zdravorazumski', pri čemu se sličnost mjeri brojem istih slova. Korisna funkcija u ovoj tački je `intersect`.
6. Kreirajte promjenljivu koja sadrži string 'eci peci pec ti si mali zec'. Tokenizujte ovu promjenljivu na riječi koje sadrži. Rezultat treba da bude ćelijski niz čiji elementi sadrže pojedine riječi zadatog stringa.
7. Učitajte tekst iz fajla (npr. `http://www.textfiles.com/humor/101nos.txt`) u promjenljivu `sadrzaj_fajla`.
8. Tokenizujte sadržaj sadržaj promjenljive `sadrzaj_fajla` na sastavne riječi. Sačuvajte sve riječi iz čitavog dokumenta (bez obzira na ponavljanja) u ćelijskom nizu `rijeci` tako da jedna riječ bude u jednom elementu niza. Eksperimentišite sa znacima koji predstavljaju granice riječi.

9. Formirajte rječnik koji sadrži sve jedinstvene riječi u dokumentu pomoću funkcije `unique`.

10. Kreirajte vektor čiji je svaki element broj pojavljivanja odgovarajuće riječi iz rječnika u dokumentu. To je vektorska reprezentacija dokumenta.