

Osnovi programiranja

V termin

28.12.2017.

1. Napisati program u kome se najpre unose ime tekstualne datoteke sa podacima o **trakama** i ime izlazne datoteke. Definirati strukturu **traka** koja sadrži sledeće podatke:

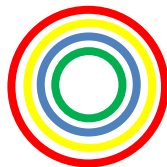
- **Boja trake (jedna rec - "CRVENA", "ZELENA", "PLAVA" i "ZUTA" sigurno mogu biti samo date vrednosti),**
- **Duzina trake (realan broj u centimetrima)**

Napisati sledeće funkcije:

- Funkciju **unosTraka** koja iz datoteke **datog naziva** čita podatke o trakama sve dok ne dođe do kraja datoteke i podatke smešta u niz traka i vraća ceo broj koji predstavlja broj učitanih traka. U datoteci se u svakom redu nalaze boja trake1, duzina trake1, broja trake2, duzina trake2...
- Funkciju **ispisTraka** koja u dati **fajl** ispisuje SVE podatke o trakama koje se nalaze u nizu traka date dužine tako da su podaci o jednoj traci napisani u jednom redu, o drugoj u drugom redu itd. *Jedan od argumenata funkcije je i fajl (nije naziv datoteke) u koji se upisuju podaci.*
- Funkciju **duzinaT** koja za dati niz traka date dužine i datu boju vraća ukupnu dužinu trake date boje. (Dva stringa su jednaka ukoliko su im svi karakteri jednaki.)
- Funkciju **formBoju** koja na osnovu dve date boje formira novu boju kao **boja1-boja2**.
- Kružna meta se pravi od traka, postavljanjem trake jedne boje po obimu svake kružnice (zanemaruje se širina trake, trake mogu da se nastavljaju, obim kružnice je $2r\pi$). Poluprečnik prve kružnice je 10cm, a svaka naredna kružnica ima poluprečnik za 10% veći od prethodnog. Boje se postavljaju sledećim redosledom: CRVENA, ZELENA, PLAVA, ZUTA, pa ponovo istim redosledom. Sa formiranjem mete se staje kada nije moguće napraviti sledeću kružnicu. Napisati funkciju **brojK** koja na osnovi niza traka date dužine određuje koliko kružnica može da se formira.

U glavnom delu programa učitati nazive ulazne i izlazne datoteke, zatim formirati niz od traka koje se nalaze u ulaznoj datoteci koristeći funkciju **unosTraka** i ispisuje ga u izlazni fajl koristeći funkciju **ispisTraka**. Zatim:

- koristeći funkciju **duzinaT** u izlazni fajl za svaku boju odštampati njenu dužinu, ispod već odštampanog niza u formatu:
boja1 duzina1
boja2 duzina2
...
- koristeći funkciju **formBoju** formirati novu boju od boja CRVENA i PLAVA i odštampati je u izlazni fajl ispod već navedenog.
- Koristeći funkciju **brojK** odrediti koliko kružnica se može formirati na meti. Odštampati dati broj ispod već ispisanog u fajlu.



2. Napisati program koji:

dobija ime tekstualne datoteke sa podacima o kartama (ime kupca, tip karata (broj 1, 2, 3), i broj kupljenih karata) i to po jedan podatak u jednom redu. Definirati strukturu **karte** koja sadrži navedene podatke. Arsenal fest traje tri dana, karta tipa 1 predstavlja ulaznicu za prvo veče i košta 900din, karta tipa 2 predstavlja ulaznicu za drugo veče i košta 700din i karta tipa 3 za treće veče i košta 500din. Program treba da sadrži funkciju **Kupac** koja za zadato ime kupca određuje koliko je ukupno novca taj kupac potrošio. Organizatori Arsenal festa žele da naprave niz kupaca, gde će za svakog kupca u okviru strukture **kupac** čuvati njegovo ime i ukupan potrošen novac. U ulaznoj datoteci jedan kupac može da se javi više puta a u nizu kupaca samo jednom. Štampati dobijeni niz kupaca.

3. Napisati program u kome se najpre unose ime tekstualne datoteke sa podacima o **igračima** i ime izlazne datoteke. Definirati strukturu **igrac** koja sadrži sledeće podatke:

- **Ime igrača (jedna rec),**
- **ID igrača (ceo broj),**
- **Izabrani broj (ceo broj)**

Napisati sledeće funkcije:

- f) Funkciju **unosigraca** koja iz datoteke **datog naziva** čita podatke o igračima sve dok ne dođe do kraja datoteke i podatke smešta u niz igrača i vraća ceo broj koji predstavlja broj učitanih igrača. U datoteci se u svakom redu nalaze ime1, izabran broj1, ime2, izabran broj2... ID igrača se ne nalazi u datoteci već se dodeljuje na osnovu rednog broja učitanoig igrača.
- g) Funkciju **ispisigraca** koja u dati **fajl** ispisuje SVE podatke o igračima koji se nalaze u nizu date dužine tako da su podaci o jednom igraču napisani u jednom redu, o drugom u drugom redu itd. *Jedan od argumenata funkcije je i fajl (nije naziv datoteke) u koji se upisuju podaci.*
- h) Funkciju **izbaciNajmanji** koja niz igrača date dužine transformiše tako što iz njega izbacuje **sve** one igrače čiji je izabrani broj najmanji od svih izabranih brojeva i vraća broj igrača transformisanog niza.
- i) Funkciju **formRunda** koja na osnovi niza igrača date dužine formira nov niz igrača koji prelaze u sledeću rundu i vraća broj igrača koji su prešli u novu rundu. Igrač prelazi u novu rundu ako je njegov izabrani broj jedinstven.
- j) Funkciju **najduzelme** koja na osnovi niza igrača date dužine vraća ID igrača čije je ime najduže (ukoliko ima više imena iste dužine vratiti ID prvog igrača koji ima ime najveće dužine).

U glavnom delu programa učitati nazive ulazne i izlazne datoteke, zatim formirati niz od igrača koji se nalaze u ulaznoj datoteci koristeći funkciju **unosigraca** i ispisuje ga u izlazni fajl koristeći funkciju **ispisigraca**. Zatim:

- koristeći funkciju **najduzelme** na standardni izlaz odštampati podatke o igraču koji ima najduže ime u datom nizu.
- transformiše formiran niz koristeći funkciju **izbaciNajmanji** i ispisuje ga koristeći funkciju **ispisigraca** u istu izlaznu datoteku ispod već ispisanog početnog niza.
- na osnovu **transformisanog** niza igrača formirati niz igrača koji prelaze u novu rundu koristeći funkciju **formRunda** i ispisati ga koristeći funkciju **ispisigraca** u istu izlaznu datoteku ispod već ispisanih nizova.

Na osnovu zadatka u izlaznoj datoteci će se nalaziti tri ispisan niza, njihove ispise **OBAVEZNO** razdvojiti tako što ćete u jednom redu ispisati samo -----.