

Osnovi programiranja

III kolokvijum

15.1.2019.

I grupa

Na **Desktop**-u u direktorijumu **Rad** kreirati direktorijum **ImePrezime_BrIndeksa** i unutar njega sačuvati programe koji sadrže rešenja datih zadataka. Rešenje 1. zadatka **mora** da se nalazi u fajlu **Zadatak1.c**, rešenje 2. zadatka **mora** da se nalazi u fajlu **Zadatak2.c**. Od dva ponuđena zadatka birate jedan koji ćete raditi. Nije dozvoljeno korišćenje globalnih promenljivih.

1. Napisati program u kome se najpre unose ime tekstualne datoteke sa imenima **NBA igrača**, zatim ime tekstualne datoteke sa podacima o **glasovima za ALL-STAR meč** i ime izlazne datoteke.

Definisati strukturu **date** koja ima sledeće podatke:

- **dan (ceo broj)**
- **mesec (ceo broj)**
- **godina (ceo broj)**

Definisati strukturu **player** koja ima sledeće podatke:

- **Ime igrača (jedna reč),**
- **Broj glasova (ceo broj)**

Definisati strukturu **vote** koja ima sledeće podatke:

- **ID glasača (ceo broj),**
- **Datum glasanja (datum),**
- **Ime igrača (jedna reč).**

Napisati sledeće funkcije:

- a) Funkciju **UnosIgraca** koja iz datoteke **datog naziva** čita imena igrača. Na početku datoteke se nalazi broj igrača, a zatim u svakom redu ime igrača. Podaci o igračima se smeštaju u niz igrača i vraća se ceo broj koji predstavlja broj učitanih igrača. Broj glasova koji inicijalno imaju svi igrači je 0.
- b) Funkciju **UnosGlasova** koja iz datoteke **datog naziva** čita podatke o glasovima, smešta ih u niz glasova, i vraća broj učitanih glasova. Nije poznat broj glasova u datoteci. U svakom redu su redom smešteni podaci o glasu, i to, id glasača, dan, mesec, godina glasanja, i ime igrača za koga se glasa (svi podaci odvojeni praznim mestom).
- c) Funkciju **IspisIgraca** koja u datu datoteku ispisuje podatke o svim igračima (ime i broj osvojenih glasova) koji se nalaze u datom nizu igrača, date dužine tako da su podaci o jednom igraču napisani u jednom redu, drugi u drugom redu itd. (datoteka je otvorena u glavnom delu programa)
- d) Funkciju **IzbaciSaK** koja iz datog niza glasova, date dužine, i za dati ceo broj **k**, izbacuje glas koji se nalazi na **k**-tom mestu i vraća nov broj glasova.
- e) Funkciju **IstiDatum** koja ispituje da li su dva data datuma ista i vraća 1 ako jesu a 0 ako nisu.
- f) Funkciju **IzbaciNevazece** koja iz datog niza glasova izbacuje sve nevažeće glasove i vraća nov broj glasova u nizu. Glas je nevažeći ako je istog dana isti čovek već glasao, samo prvi zabeleženi glas određenog glasača nekog datuma je važeći (koristeći funkcije **IzbaciSaK** i **IstiDatum**).
- g) Funkciju **Prebroj** koja za dati niz igrača i dati niz glasova broji glasove svakog igrača i ažurira niz igrača tim podacima.

U glavnom delu programa učitati nazive ulaznih i izlazne datoteke, zatim formirati niz od igrača koji se nalaze u prvoj ulaznoj datoteci koristeći funkciju **UnosIgraca**. Koristeći funkciju **UnosGlasova** formirati niz glasova koji se nalaze u drugoj ulaznoj datoteci.

Zatim:

- koristeći funkciju **IspisIgraca** štampati podatke o učitanim igračima u izlaznu datoteku, a zatim u istu datoteku štampati i podatke o glasovima
- koristeći funkciju **IzbaciNevazece** izbaciti sve nevažeće glasove iz niza glasova. U izlaznu datoteku ispisati niz važećih glasova, podaci o jednom glasu se nalaze u jednom redu.
- koristeći funkciju **Prebroj** prebrojati sve glasove i koristeći funkciju **IspisIgraca** štampati podatke o svim igračima u istu izlaznu datoteku

igraci.txt	glasovi.txt	Izlaz.txt
10 Marej Jokic Milsap Dzejms Harden Kari Bjelica Bogdanovic Vucevic Irving	1 10 1 2019 Jokic 2 10 1 2019 Jokic 2 11 1 2019 Jokic 1 12 1 2019 Bogdanovic 2 12 1 2019 Bjelica 1 11 1 2019 Bjelica 3 10 1 2019 Bjelica 1 12 1 2019 Bogdanovic 1 12 1 2019 Bogdanovic 3 10 1 2019 Bjelica 1 14 1 2019 Jokic	Marej 0 Jokic 0 Milsap 0 Dzejms 0 Harden 0 Kari 0 Bjelica 0 Bogdanovic 0 Vucevic 0 Irving 0 ----- 1 10 1 2019 Jokic 2 10 1 2019 Jokic 2 11 1 2019 Jokic 1 12 1 2019 Bogdanovic 2 12 1 2019 Bjelica 1 11 1 2019 Bjelica 3 10 1 2019 Bjelica 1 12 1 2019 Bogdanovic 1 12 1 2019 Bogdanovic 3 10 1 2019 Bjelica 1 14 1 2019 Jokic ----- 1 10 1 2019 Jokic 2 10 1 2019 Jokic 2 11 1 2019 Jokic 1 12 1 2019 Bogdanovic 2 12 1 2019 Bjelica 1 11 1 2019 Bjelica 3 10 1 2019 Bjelica 1 14 1 2019 Jokic ----- Marej 0 Jokic 4 Milsap 0 Dzejms 0 Harden 0 Kari 0 Bjelica 3 Bogdanovic 1 Vucevic 0 Irving 0

2. Napisati program u kome se najpre unose ime tekstualne datoteke sa šiframa **NBA igrača** (celi brojevi), zatim ime tekstualne datoteke sa podacima o **glasovima za ALL-STAR meč** i ime izlazne datoteke.

Napisati sledeće funkcije:

- Funkciju **UnosIgrača** koja iz datoteke **datog naziva** čita šifre igrača (šifre su celi brojevi). Na početku datoteke se nalazi broj igrača, a zatim u svakom redu šifra igrača. Podaci o igračima se smeštaju u niz šifara igrača i vraća se ceo broj koji predstavlja broj učitanih igrača.
- Funkciju **UnosGlasova** koja iz datoteke **datog naziva** čita podatke o glasovima, smešta ih u niz glasova, i vraća broj učitanih glasova. Nije poznat broj glasova u datoteci. U svakom redu je smeštena po jedna šifra (ceo broj) koja predstavlja glas za igrača čija je to šifra.
- Funkciju **IspisIgrača** koja u datu datoteku ispisuje podatke o igračima (šifra i broj osvojenih glasova) koji se nalaze u datim nizovima šifara i glasova igrača, date dužine (nizovi su iste dužine) tako da su podaci o jednom igraču napisani u jednom redu, drugi u drugom redu itd. (datoteka je otvorena u glavnom delu programa)

- d) Funkciju **IzbaciSaK** koja iz datog niza celih brojeva izbacuje element koji se nalazi na **k**-tom mestu i vraća novu dužinu niza.
- e) Funkciju **IzbaciNevazece** koja iz datog niza glasova izbacuje sve nevažeće glasove i vraća nov broj glasova u nizu. Glas je nevažeći ako je istog dana već glasano za neku šifru igrača, samo prvi zabeleženi glas tog dana je važeći (koristeći funkciju **IzbaciSaK**). Smatrati da dati niz glasova sadrži samo glasove od dva dana. Prva polovina niza pripada prvom danu, a druga drugom (imaće paran broj glasova)
- f) Funkciju **Prebroj** koja za dati niz šifara date dužine, i dati niz glasova date dužine, formira niz vrednost (isti broj elemenata kao i niz šifara) čiji elementi redom odgovaraju šiframa igrača i predstavlja broj glasova koje je dobio svaki igrač. Inicijalno svaki igrač ima nula glasova.

U glavnom delu programa učitati nazive ulaznih i izlazne datoteke, zatim formirati niz od šifara igrača koji se nalaze u prvoj ulaznoj datoteci koristeći funkciju **UnosIgraca**. Koristeći funkciju **UnosGlasova** formirati niz ukupnih glasova koji se nalaze u drugoj ulaznoj datoteci. Niz trenutnih glasova koji ima isto elemenata kao i niz šifara potrebno je inicijalizovati na nulu. To je stanje pre prispelih glasova iz niza ukupnih glasova. Zatim:

- Koristeći funkciju **IspisIgrača** štampati sifre igrača zajedno sa trenutnim glasovima. U izlaznu datoteku štampati i sve ukupne glasove
- koristeći funkciju **IzbaciNevazece** izbaciti sve nevažeće glasove iz niza ukupnih glasova. U izlaznu datoteku ispisati niz važećih ukupnih glasova.
- koristeći funkciju **Prebroj** prebrojati sve glasove i koristeći funkciju **IspisIgrača** štampati podatke o svim igračima i njihovim glasovima u istu izlaznu datoteku

igraci.txt	glasovi.txt	Izlaz.txt
10	11	11 0
11	12	12 0
12	11	19 0
19	19	13 0
13	19	15 0
15	25	16 0
16	25	22 0
22	25	20 0
20	18	18 0
18	12	25 0
25	11	-----
	19	11
	19	12
	25	11
	25	19
	25	19
		25
		25
		25
		18
		12
		11
		19
		19
		25
		25
		25

		11
		12
		19
		25
		18
		12
		11

		19
		25

		11 2
		12 2
		19 2
		13 0
		15 0
		16 0
		22 0
		20 0
		18 1
		25 2

Broj poena: 1. zadatak – 21 poena, 2. zadatak – 15 poena
Vreme izrade: 120 minuta

Osnovi programiranja III kolokvijum

15.1.2019.

II grupa

Na **Desktop**-u u direktorijumu **Rad** kreirati direktorijum **ImePrezime_BrIndeksa** i unutar njega sačuvati programe koji sadrže rešenja datih zadataka. Rešenje 1. zadatka **mora** da se nalazi u fajlu **Zadatak1.c**, rešenje 2. zadatka **mora** da se nalazi u fajlu **Zadatak2.c**. Od dva ponuđena zadatka birate jedan koji ćete raditi. Nije dozvoljeno korišćenje globalnih promenljivih.

3. Napisati program u kome se najpre unose ime tekstualne datoteke sa podacima o **lego kockicama**, zatim ime tekstualne datoteke sa podacima o **lego kutijama** i ime izlazne datoteke.

Definisati strukturu **LegoSockica** koja sadrži sledeće podatke:

- **ID oblika kockice (ceo broj),**
- **Boja kockice (jedna reč),**
- **Kolicina (ceo broj).**

Definisati strukturu **LegoSutija** koja sadrži sledeće podatke:

- **Boja (jedna reč),**
- **Broj različitih kockica (ceo broj),**
- **Kockice (niz LegoSockica).**

Napisati sledeće funkcije:

- a) Funkciju **UnosLegoSockica** koja iz datoteke **datog naziva** čita podatke o kockicama. Na početku datoteke se nalazi broj kockica, a zatim u svakom redu po id oblika kockice, boja i količina te kockice. Podaci o lego kockicama se smeštaju u niz kockica i vraća se ceo broj koji predstavlja broj učitanih kockica.
- b) Funkciju **UnoSutija** koja iz datoteke **datog naziva** čita podatke o lego kutijama, tačnije njihove boje. Boje lego kutija su date jedna ispod druge do kraja datoteke. Nepoznat je unapred broj kutija. Sve kutije su inicijalno prazne, broj različitih kockica je nula i niz kockica nema elemente. Funkcija vraća broj učitanih kutija.
- c) Funkciju **IspisSutije** koja za datu kutiju ispisuje podatke o svim kockicama iz te kutije u fajl čiji se naziv formira od boje kutije i ekstenzije „.txt“.
- d) Funkciju **UbaciLegoSockice** koja dati niz lego kockica date dužine ubacuje u niz kutija date dužine. Ubacivanje se vrši na taj način što sve kockice neke boje se smeštaju u kutiju te iste boje.
- e) Funkciju **LegoSutija SortSutiju(LegoSutija legoSutija)** koja niz lego kockica date kutije sortira prema kolicini, i vraća istu kutiju koja je sada sortirana. (napomena: dato vam je kako treba da izgleda funkcija i kako bi promene

prilikom sortiranja bile vidljive u glavnom delu programa, kada završite sortiranje legoKutija, potrebno je samo da je vratite sa **return legoKutija**)

U glavnom delu programa učitati nazive ulaznih datoteke, zatim formirati niz od lego kockica koji se nalaze u prvoj ulaznoj datoteci koristeći funkciju **UnosLegoKockice**. Potom učitava boje kutija i formira niz kutija koristeći funkciju **UnosKutija**. Zatim:

- Na standardni izlaz štampati uneti niz lego kockica i unete boje kutija.
- Raspoređuje lego kockice u kutije po bojama koristeći funkciju **UbaciLegoKockice**.
- Koristeći funkciju **SortKutiju** sortirati ceo niz kutija, zatim podatke o svim kutijama ispisati u posebnim fajlovima koristeći funkciju **IspisKutije**.

kockice.txt	kutije.txt	Izlaz (standardni)	plava.txt	zuta.txt	crvena.txt
6 12 crvena 10 11 plava 20 4 crvena 15 10 crvena 12 10 plava 5 10 zuta 6	plava zuta crvena	12 crvena 10 11 plava 20 4 crvena 15 10 crvena 12 10 plava 5 10 zuta 6 ----- plava zuta crvena	11 plava 20 10 plava 5	10 zuta 6	4 crvena 15 10 crvena 12 12 crvena 10

4. Napisati program u kome se najpre unose ime tekstualne datoteke sa šiframa **Proizvoda** (celi brojevi), zatim ime tekstualne datoteke sa podacima o **zahtevima za kupovinu datih proizvoda** i ime izlazne datoteke.

Napisati sledeće funkcije:

- g) Funkciju **UnosProizvoda** koja iz datoteke **datog naziva** čita šifre proizvode. Na početku datoteke se nalazi broj proizvoda, a zatim u svakom redu šifra proizvoda. Podaci o proizvodima se smeštaju u niz šifara proizvoda i vraća se ceo broj koji predstavlja broj učitanih proizvoda.
- h) Funkciju **UnosLicitacija** koja iz datoteke **datog naziva i na osnovu niza šifara proizvoda date dužine** čita podatke o licitiranim proizvodima i smešta u niz licitiranih cena proizvoda koji odgovara nizu šifara proizvoda. U svakom trenutku za cenu proizvoda se pamti najveća do tad licitirana cena (pogledati test primer). Nije poznat broj licitacija u datoteci. U svakom redu se nalazi šifra proizvoda i licitirana cena proizvoda.
- i) Funkciju **IspisProizvoda** koja u datu datoteku ispisuje podatke o proizvodima (šifra i licitiranu cenu) koji se nalaze u datim nizovima šifara proizvoda i licitiranim cenama proizvoda, date dužine (nizovi su iste dužine) tako da su podaci o jednom proizvodu napisani u jednom redu, drugi u drugom redu itd. (datoteka je otvorena u glavnom delu programa)
- j) Funkciju **IzbaciNelicitirane** koja iz datih nizova šifara proizvoda, licitiranih cena proizvoda i njihove dužine (imaju istu dužinu) izbacuje sve nelicitirane proizvode i vraća novu dužinu nizova.
- k) Funkciju **SortProizvode** koja za date nizove šifara proizvoda i licitiranih cena proizvoda i njihovu dužinu (ista je dužina) sortira proizvode prema licitiranoj ceni proizvoda (od najveće ka najmanjoj ceni)

U glavnom delu programa učitati nazive ulaznih i izlazne datoteke, zatim formirati niz od šifara proizvoda koji se nalaze u prvoj ulaznoj datoteci koristeći funkciju **UnosProizvoda**. Koristeći funkciju **UnosLicitacija** na osnovu niza šifara proizvoda i podataka u drugoj ulaznoj datoteci formira niz licitiranih cena proizvoda. Zatim:

- Koristeći funkciju **IspisProizvoda** štampati sifre proizvoda zajedno sa licitiranim cenama proizvoda.
- Koristeći funkciju **IzbaciNelicitirane** izbaciti proizvode koji nisu licitirani i iz niza šifara proizvoda i iz niza licitiranih cena proizvoda i koristeći funkciju **IspisProizvoda** štampati sifre proizvoda zajedno sa licitiranim cenama proizvoda.
- Koristeći funkciju **SortProizvode** sortirati niz proizvoda prema licitiranoj ceni.
- Koristeći funkciju **IspisProizvoda** štampati sifre proizvoda zajedno sa licitiranim cenama proizvoda nakon sortiranja.

proizvodi.txt	licitirane cene.txt	Izlaz.txt
7 2222 9999 1111	2222 50 1111 60 2222 60 8888 80	2222 60 9999 0 1111 90 5555 50

5555	8888 50	3333 0
3333	5555 30	4444 0
4444	1111 10	8888 80
8888	5555 50	-----
	1111 90	2222 60
	5555 10	1111 90
	2222 50	5555 50
		8888 80

		1111 90
		8888 80
		2222 60
		5555 50

Broj poena: 1. zadatak – 21 poena, 2. zadatak – 15 poena

Vreme izrade: 120 minuta