

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Rezultati i uvidi u kolokvije: Rezultati u srijedu, 28.6., navečer na webu, a uvidi u četvrtak, 29.6., u 14 sati.

Upute: Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni podsjetnik. Kalkulatori, razne neslužbene tablice, papiri i sl., nisu dozvoljeni! **Mobitele isključite i spremite!** Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent. U svim zadacima **zabranjeno je korištenje dodatnih nizova** i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Zadatak 1. (20 bodova) Napišite funkciju

```
char *prosiri(char *tekst, char **skr, char **puni, int n)
```

koja kao parametre prima string `tekst` i nizove od `n` stringova `skr` i `puni`. U nizu `skr` pišu skraćenice, a u nizu `puni` na odgovarajućim mjestima pišu puni oblici od kojih su one skraćenice. Npr. ako na prvom mjestu u nizu `skr` piše "npr.", na istom mjestu u nizu `puni` će pisati "na primjer".

Funkcija vraća pokazivač na string dobiven iz stringa `tekst`, tako da su u njemu sve skraćenice zamijenjene svojim punim oblicima. Riječ u stringu `tekst` je skraćenica ako se javlja u nizu `skr`. Smijete pretpostaviti da nijedna skraćenica neće biti podstring druge. Dozvoljeno je koristiti funkcije iz `string.h`.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 2. (15 bodova) Deklarirajte strukturu `destinacija`, koja označava destinaciju, a sadrži naziv (string od najviše 30 znakova), te cijenu (tipa `int`). Deklarirajte strukturu `student` koja sadrži ime (string od najviše 30 znakova), broj n destinacija na koje bi taj student želio ići (tipa `int`), te pokazivač na niz pripadnih destinacija. Pretpostavljamo da taj niz ima točno n destinacija.

Napišite funkciju `odgovara` koja prima niz od m studenata. Funkcija treba vratiti pokazivač na alociranu destinaciju koja odgovara svim studentima i još ima najmanju cijenu. Ako takvih destinacija ima više, uzmite onu najmanjeg naziva (po leksikografskom uređaju). Ako takvih destinacija nema, treba vratiti `NULL`. Smijete koristiti funkcije iz `string.h`.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 3. (2 + 9 + 9 bodova) Trgovački putnik želi posjetiti niz gradova kako bi prodavao svoju robu. Svaki je grad određen svojim imenom (string od najviše 50 znakova), koordinatama x i y (realni brojevi), te zaradom (pozitivni cijeli broj) koju putnik ostvaruje u tom gradu. Niz gradova koje je potrebno posjetiti bit će čuvan u vezanoj listi.

(a) Napišite definiciju odgovarajućeg tipa podatka za grad, na način da bude moguće definirati varijable naredbom `grad g` i sl. Definirajte samo podatke koji su nužni za čuvanje takve liste u memoriji.

(b) Ponekad trgovački putnik odlučuje posjetiti grad koji nije u originalnom planu, te mora ažurirati svoj raspored. Napišite funkciju `grad* organiziraj_put(grad *first, grad *novi)` koja prima vezanu listu gradova te grad `novi`. Funkcija ubacuje grad `novi` u listu na sljedeći način: traži dva uzastopna grada g_1 i g_2 tako da je suma *težinske zarade* putovanja od g_1 do grada `novi` i *težinske zarade* od grada `novi` do g_2 **maksimalna**, nakon čega se ubacuje čvor `novi` između g_1 i g_2 . *Težinska zarada* je kvocijent zarade u odredišnom gradu i euklidske udaljenosti polaznog i odredišnog grada.

Ukoliko je lista `first` prazna, `novi` postaje jedini čvor, a ukoliko ima točno jedan čvor, tada se `novi` dodaje na kraj.

(c) Napišite funkciju `grad* odvoji(grad **first, char *ime, double rad)` koja će iz dane liste izbaciti sve gradove **unutar** radijusa `rad` od grada s imenom `ime` (uključujući i taj grad). Izbačene gradove potrebno je ubaciti u novu listu koristeći funkciju `organiziraj_put`.

Ako grad s tim imenom ne postoji, vraća se `NULL`, a originalna lista se ne mijenja. Ova funkcija vraća pokazivač na novu listu, a originalnu modificira preko varijabilnog parametra.

Napomena: Smijete koristiti standardne funkcije `strcmp` i `sqrt` iz odgovarajućih biblioteka. Za rješavanje podzadatka (c) nije nužno da riješite podzadatak (b).

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 4. (15 bodova) U ulaznoj tekstualnoj datoteci zapisana je realna kvadratna matrica A , reda $n \geq 2$, na sljedeći način. Prvi broj u datoteci je cijeli broj n , koji označava red matrice. Nakon toga, zapisani su svi elementi matrice, redak po redak, odvojeni razmacima i/ili skokovima u novi red. Elementi matrice su tipa `double`.

- (a) Napišite funkciju `int split(FILE *ulaz, FILE *b)` koja čita matricu A iz već otvorene ulazne datoteke, zadane pokazivačem `ulaz`. Funkcija vraća onaj indeks k ($1 \leq k < n$) za koji se dostiže minimum izraza

$$\left| \sum_{i=1}^n \sum_{j=1}^k a_{i,j} - \sum_{i=1}^n \sum_{j=k+1}^n a_{i,j} \right|.$$

Ako takvih indeksa ima više, uzima se najmanji od njih. Pripadnu podmatricu B_k , koja sadrži prvih k stupaca od A , treba napisati u već otvorenu binarnu datoteku, zadanu pokazivačem `b`, tako da se prvo napišu dva cijela broja n i k , a zatim svi elementi te podmatrice B_k , redak po redak.

- (b) Napišite program koji s komandne linije učitava dva argumenta. Prvi argument je ime ulazne tekstualne datoteke, a drugi je ime izlazne binarne datoteke. Program treba otvoriti obje datoteke, pozvati funkciju iz (a) i ispisati njezin rezultat, te zatvoriti obje datoteke.

Napomena: Smijete koristiti maksimalno jedno (dvodimenzionalno) polje za rješavanje zadatka. Oslobodite svu memoriju koju alocirate. Ignorirajte moguće greške kod čitanja argumenata i rada s datotekama.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Rezultati i uvidi u kolokvije: Rezultati u srijedu, 28.6., navečer na webu, a uvidi u četvrtak, 29.6., u 14 sati.

Upute: Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni podsjetnik. Kalkulatori, razne neslužbene tablice, papiri i sl., nisu dozvoljeni! **Mobitele isključite i spremite!** Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent. U svim zadacima **zabranjeno je korištenje dodatnih nizova** i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Zadatak 1. (20 bodova) Napišite funkciju

```
char *objasni(char *tekst, char **akronim, char **naziv, int n)
```

koja kao parametre prima string `tekst` i nizove od `n` stringova `akronim` i `naziv`. U nizu `akronim` pišu akronimi, a u nizu `naziv` na odgovarajućim mjestima pišu puni nazivi od kojih su oni akronimi. Npr. ako na prvom mjestu u nizu `akronim` piše "SAD", na istom mjestu u nizu `naziv` će pisati "Sjedinjene Američke Države".

Funkcija vraća pokazivač na string dobiven iz stringa `tekst`, tako da su u njemu svi akronimi zamijenjeni svojim punim nazivima. Riječ u stringu `tekst` je akronim ako se javlja u nizu `akronim`. Smijete pretpostaviti da nijedan akronim neće biti podstring drugog. Dozvoljeno je koristiti funkcije iz `string.h`.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 2. (15 bodova) Deklarirajte strukturu `ponuda`, koja označava ponudu neke tvrtke za izgradnju hotela, a sadrži naziv tvrtke (string od najviše 30 znakova), te broj kreveta u hotelu (tipa int). Deklarirajte strukturu `clan` koja sadrži ime osobe (string od najviše 30 znakova), broj n ponuda tvrtki koje ta osoba smatra dobrima (tipa int), te pokazivač na niz pripadnih ponuda. Pretpostavljamo da taj niz ima točno n ponuda.

Napišite funkciju `odabrana` koja prima niz od m članova upravnog odbora tvrtke koja želi investirati. Funkcija treba vratiti pokazivač na alociranu ponudu koja odgovara svim članovima i još ima najviše kreveta. Ako takvih ponuda ima više, uzmite onu najvećeg naziva (po leksikografskom uređaju). Ako takvih destinacija nema, treba vratiti `NULL`. Smijete koristiti funkcije iz `string.h`.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 3. (2 + 9 + 9 bodova) Dostavljač mora posjetiti niz lokacija kako bi dostavio robu. Svaka je lokacija određena svojim nazivom (string od najviše 30 znakova), koordinatama x i y (realni brojevi), te vremenom (pozitivni cijeli broj) koje je potrebno za iskrcavanje robe na toj lokaciji. Niz lokacija koje je potrebno posjetiti bit će čuvan u vezanoj listi.

- (a) Napišite definiciju odgovarajućeg tipa podatka za lokaciju, na način da bude moguće definirati varijable naredbom `lokacija l` i sl. Definirajte samo podatke koji su nužni za čuvanje takve liste u memoriji.
- (b) Ponekad dostavljač mora dostaviti robu na lokaciju koja nije u originalnom planu, te mora ažurirati svoj raspored. Napišite funkciju `lokacija* planiraj_dostavu(lokalacija *first, lokalacija *nova)` koja prima vezanu listu lokacija te lokaciju `nova`. Funkcija ubacuje lokaciju `nova` u listu na sljedeći način: traži dvije uzastopne lokacije l_1 i l_2 tako da je suma *ukupnog vremena dostave* od l_1 do lokacije `nova` i *ukupnog vremena dostave* od lokacije `nova` do l_2 **minimalna**, nakon čega se ubacuje čvor `nova` između l_1 i l_2 . *Ukupno vrijeme dostave* je suma euklidske udaljenosti polazne i odredišne lokacije te vremena provedenog na odredišnoj lokaciji.
- Ukoliko je lista `first` prazna, `nova` postaje jedini čvor, a ukoliko ima točno jedan čvor, tada se `nova` dodaje na kraj.
- (c) Napišite funkciju `lokacija* zanemari(lokalacija **first, char *naziv, double rad)` koja će iz dane liste izbaciti sve lokacije **unutar** radijusa `rad` od lokacije s nazivom `naziv` (uključujući i tu lokaciju). Izbačene lokacije potrebno je ubaciti u novu listu koristeći funkciju `planiraj_dostavu`.

Ako lokacija s tim nazivom ne postoji, vraća se NULL, a originalna lista se ne mijenja. Funkcija vraća pokazivač na novu listu, a originalnu modificira preko varijabilnog parametra.

Napomena: Smijete koristiti standardne funkcije `strcmp` i `sqrt` iz odgovarajućih biblioteka. Za rješavanje podzadatka (c) nije nužno da riješite podzadatak (b).

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 4. (15 bodova) U ulaznoj tekstualnoj datoteci zapisana je realna kvadratna matrica A , reda $n \geq 2$, na sljedeći način. Prvi broj u datoteci je cijeli broj n , koji označava red matrice. Nakon toga, zapisani su svi elementi matrice, redak po redak, odvojeni razmacima i/ili skokovima u novi red. Elementi matrice su tipa `double`.

- (a) Napišite funkciju `int split(FILE *ulaz, FILE *b)` koja čita matricu A iz već otvorene ulazne datoteke, zadane pokazivačem `ulaz`. Funkcija vraća onaj indeks k ($1 \leq k < n$) za koji se dostiže minimum izraza

$$\left| \sum_{i=1}^k \sum_{j=1}^n a_{i,j} - \sum_{i=k+1}^n \sum_{j=1}^n a_{i,j} \right|.$$

Ako takvih indeksa ima više, uzima se najmanji od njih. Pripadnu podmatricu B_k , koja sadrži prvih k redaka od A , treba napisati u već otvorenu binarnu datoteku, zadanu pokazivačem `b`, tako da se prvo napišu dva cijela broja k i n , a zatim svi elementi te podmatrice B_k , redak po redak.

- (b) Napišite program koji s komandne linije učitava dva argumenta. Prvi argument je ime ulazne tekstualne datoteke, a drugi je ime izlazne binarne datoteke. Program treba otvoriti obje datoteke, pozvati funkciju iz (a) i ispisati njezin rezultat, te zatvoriti obje datoteke.

Napomena: Smijete koristiti maksimalno jedno (dvodimenzionalno) polje za rješavanje zadatka. Oslobodite svu memoriju koju alocirate. Ignorirajte moguće greške kod čitanja argumenata i rada s datotekama.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Rezultati i uvidi u kolokvije: Rezultati u srijedu, 28.6., navečer na webu, a uvidi u četvrtak, 29.6., u 14 sati.

Upute: Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni podsjetnik. Kalkulatori, razne neslužbene tablice, papiri i sl., nisu dozvoljeni! **Mobitele isključite i spremite!** Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent. U svim zadacima **zabranjeno je korištenje dodatnih nizova** i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Zadatak 1. (20 bodova) Napišite funkciju

```
char *imena(char *tekst, char **nadimak, char **ime, int n)
```

koja kao parametre prima string `tekst` i nizove od `n` stringova `nadimak` i `ime`. U nizu `nadimak` pišu nadimci, a u nizu `ime` na odgovarajućim mjestima pišu prava imena ljudi s tim nadimcima. Npr. ako na prvom mjestu u nizu `nadimak` piše "Zec", na istom mjestu u nizu `ime` će pisati "Goran Ivanišević".

Funkcija vraća pokazivač na string dobiven iz stringa `tekst`, tako da su u njemu svi nadimci zamijenjeni pravim imenima. Riječ u stringu `tekst` je nadimak ako se javlja u nizu `nadimak`. Smijete pretpostaviti da nijedan nadimak neće biti podstring drugog. Dozvoljeno je koristiti funkcije iz `string.h`.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 2. (15 bodova) Deklarirajte strukturu `predstava`, koja označava kazališnu predstavu, a sadrži naziv (string od najviše 30 znakova), te cijenu (tipa `int`). Deklarirajte strukturu `gledatelj` koja sadrži ime (string od najviše 30 znakova), broj n predstava koje bi taj gledatelj želio vidjeti (tipa `int`), te pokazivač na niz pripadnih predstava. Pretpostavljamo da taj niz ima točno n predstava.

Napišite funkciju `izvedba` koja prima niz od m gledatelja. Funkcija treba vratiti pokazivač na alociranu predstavu koja odgovara svim gledateljima i još ima najmanju cijenu. Ako takvih predstava ima više, uzmite onu najmanjeg naziva (po leksikografskom uređaju). Ako takvih predstava nema, treba vratiti `NULL`. Smijete koristiti funkcije iz `string.h`.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 3. (2 + 9 + 9 bodova) Turist želi posjetiti niz gradova kako bi ih razgledao. Svaki je grad određen svojim imenom (string od najviše 40 znakova), koordinatama x i y (realni brojevi), te vremenom (pozitivni cijeli broj) koje turist provodi pri posjetu u tom gradu. Niz gradova koje je potrebno posjetiti bit će čuvan u vezanoj listi.

(a) Napišite definiciju odgovarajućeg tipa podatka za grad, na način da bude moguće definirati varijable naredbom `grad g` i sl. Definirajte samo podatke koji su nužni za čuvanje takve liste u memoriji.

(b) Ponekad turist odlučuje posjetiti grad koji nije u originalnom planu, te mora ažurirati svoj raspored. Napišite funkciju `grad* organiziraj_izlet(grad *first, grad *novi)` koja prima vezanu listu gradova te grad `novi`. Funkcija ubacuje grad `novi` u listu na sljedeći način: traži dva uzastopna grada g_1 i g_2 tako da je suma *ukupnog vremena putovanja* od g_1 do grada `novi` i *ukupnog vremena putovanja* od grada `novi` do g_2 **maksimalna**, nakon čega se ubacuje čvor `novi` između g_1 i g_2 . *Ukupno vrijeme putovanja* je suma euklidske udaljenosti polaznog i odredišnog grada te vremena provedenog u odredišnom gradu.

Ukoliko je lista `first` prazna, `novi` postaje jedini čvor, a ukoliko ima točno jedan čvor, tada se `novi` dodaje na kraj.

(c) Napišite funkciju `grad* odvoji(grad **first, char *ime, double rad)` koja će iz dane liste izbaciti sve gradove **izvan** radijusa `rad` od grada s imenom `ime`. Izbačene gradove potrebno je ubaciti u novu listu koristeći funkciju `organiziraj_izlet`.

Ako grad s tim imenom ne postoji, vraća se NULL, a originalna lista se ne mijenja. Funkcija vraća pokazivač na novu listu, a originalnu modificira preko varijabilnog parametra.

Napomena: Smijete koristiti standardne funkcije `strcmp` i `sqrt` iz odgovarajućih biblioteka. Za rješavanje podzadatka (c) nije nužno da riješite podzadatak (b).

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 4. (15 bodova) U ulaznoj tekstualnoj datoteci zapisana je realna kvadratna matrica A , reda $n \geq 2$, na sljedeći način. Prvi broj u datoteci je cijeli broj n , koji označava red matrice. Nakon toga, zapisani su svi elementi matrice, redak po redak, odvojeni razmacima i/ili skokovima u novi red. Elementi matrice su tipa `double`.

- (a) Napišite funkciju `int split(FILE *ulaz, FILE *b)` koja čita matricu A iz već otvorene ulazne datoteke, zadane pokazivačem `ulaz`. Funkcija vraća onaj indeks k ($1 \leq k < n$) za koji se dostiže minimum izraza

$$\left| 2 \sum_{i=1}^n \sum_{j=1}^k a_{i,j} - \sum_{i=1}^n \sum_{j=k+1}^n a_{i,j} \right|.$$

Ako takvih indeksa ima više, uzima se najmanji od njih. Pripadnu podmatricu B_k , koja sadrži prvih k stupaca od A , treba napisati u već otvorenu binarnu datoteku, zadanu pokazivačem `b`, tako da se prvo napišu dva cijela broja n i k , a zatim svi elementi te podmatrice B_k , redak po redak.

- (b) Napišite program koji s komandne linije učitava dva argumenta. Prvi argument je ime ulazne tekstualne datoteke, a drugi je ime izlazne binarne datoteke. Program treba otvoriti obje datoteke, pozvati funkciju iz (a) i ispisati njezin rezultat, te zatvoriti obje datoteke.

Napomena: Smijete koristiti maksimalno jedno (dvodimenzionalno) polje za rješavanje zadatka. Oslobodite svu memoriju koju alocirate. Ignorirajte moguće greške kod čitanja argumenata i rada s datotekama.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Rezultati i uvidi u kolokvije: Rezultati u srijedu, 28.6., navečer na webu, a uvidi u četvrtak, 29.6., u 14 sati.

Upute: Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni podsjetnik. Kalkulatori, razne neslužbene tablice, papiri i sl., nisu dozvoljeni! **Mobitele isključite i spremite!** Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent. U svim zadacima **zabranjeno je korištenje dodatnih nizova** i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Zadatak 1. (20 bodova) Napišite funkciju

```
char *artikli(char *tekst, char **kod, char **opis, int n)
```

koja kao parametre prima string `tekst` i nizove od `n` stringova `kod` i `opis`. U nizu `kod` pišu kôdovi artikala, a u nizu `opis` na odgovarajućim mjestima pišu opisi artikala s tim kôdovima. Npr. ako na prvom mjestu u nizu `kod` piše "A1112", na istom mjestu u nizu `opis` će pisati "hrana za mačke".

Funkcija vraća pokazivač na string dobiven iz stringa `tekst`, tako da su u njemu svi kôdovi artikala zamijenjeni njihovim opisima. Riječ u stringu `tekst` je kôd ako se javlja u nizu `kod`. Smijete pretpostaviti da nijedan kôd neće biti podstring drugog. Dozvoljeno je koristiti funkcije iz `string.h`.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 2. (15 bodova) Deklarirajte strukturu `dionica`, koja označava dionicu, a sadrži naziv (string od najviše 30 znakova), te prinos (tipa `int`). Deklarirajte strukturu `birac` koja sadrži ime (string od najviše 30 znakova), broj n dionica koje ta osoba smatra vjerodostojnima (tipa `int`), te pokazivač na niz pripadnih dionica. Pretpostavljamo da taj niz ima točno n dionica.

Napišite funkciju `kupnja` koja prima niz od m članova udruge građana koji bi htjeli kupiti neku dionicu, a imaju pravo biranja. Funkcija treba vratiti pokazivač na alociranu dionicu koja odgovara svim biračima i još ima najveći prinos. Ako takvih dionica ima više, uzmite onu najvećeg naziva (po leksikografskom uređaju). Ako takvih dionica nema, treba vratiti `NULL`. Smijete koristiti funkcije iz `string.h`.

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 3. (2 + 9 + 9 bodova) Vozač autobusa mora posjetiti niz stanica kako bi prevezao putnike do odredišta. Svaka je stanica određena svojim nazivom (string od najviše 60 znakova), koordinatama x i y (realni brojevi), te brojem putnika (pozitivni cijeli broj) koji se očekuju na toj stanici. Niz stanica koje je potrebno posjetiti bit će čuvan u vezanoj listi.

- (a) Napišite definiciju odgovarajućeg tipa podatka za stanicu, na način da bude moguće definirati varijable naredbom `stanica s` i sl. Definirajte samo podatke koji su nužni za čuvanje takve liste u memoriji.
- (b) Ponekad vozač mora posjetiti stanicu koja nije u originalnom planu, te mora ažurirati svoj raspored. Napišite funkciju `stanica* dodaj_stanicu(stanica *first, stanica *nova)` koja prima vezanu listu stanica te stanicu `nova`. Funkcija ubacuje stanicu `nova` u listu na sljedeći način: traži dvije uzastopne stanice s_1 i s_2 tako da je suma *ukupnog troška putovanja* od s_1 do stanice `nova` i *ukupnog troška putovanja* od stanice `nova` do s_2 **minimalna**, nakon čega se ubacuje čvor `nova` između s_1 i s_2 . *Ukupan trošak putovanja* je kvocijent broja putnika na **polazišnoj** stanici te euklidske udaljenosti polazne i odredišne stanice.

Ukoliko je lista `first` prazna, `nova` postaje jedini čvor, a ukoliko ima točno jedan čvor, tada se `nova` dodaje na kraj.

- (c) Napišite funkciju `stanica* izdvoji(stanica **first, char *naziv, double rad)` koja će iz dane liste izbaciti sve stanice **izvan** radijusa `rad` od stanice s nazivom `naziv`. Izbačene stanice potrebno je ubaciti u novu listu koristeći funkciju `dodaj_stanicu`.

Ako stanica s tim nazivom ne postoji, vraća se NULL, a originalna lista se ne mijenja. Funkcija vraća pokazivač na novu listu, a originalnu modificira preko varijabilnog parametra.

Napomena: Smijete koristiti standardne funkcije `strcmp` i `sqrt` iz odgovarajućih biblioteka. Za rješavanje podzadatka (c) nije nužno da riješite podzadatak (b).

Programiranje 2 – drugi kolokvij, 23. 6. 2017.

Zadatak 4. (15 bodova) U ulaznoj tekstualnoj datoteci zapisana je realna kvadratna matrica A , reda $n \geq 2$, na sljedeći način. Prvi broj u datoteci je cijeli broj n , koji označava red matrice. Nakon toga, zapisani su svi elementi matrice, redak po redak, odvojeni razmacima i/ili skokovima u novi red. Elementi matrice su tipa `double`.

- (a) Napišite funkciju `int split(FILE *ulaz, FILE *b)` koja čita matricu A iz već otvorene ulazne datoteke, zadane pokazivačem `ulaz`. Funkcija vraća onaj indeks k ($1 \leq k < n$) za koji se dostiže minimum izraza

$$\left| 2 \sum_{i=1}^k \sum_{j=1}^n a_{i,j} - \sum_{i=k+1}^n \sum_{j=1}^n a_{i,j} \right|.$$

Ako takvih indeksa ima više, uzima se najmanji od njih. Pripadnu podmatricu B_k , koja sadrži prvih k redaka od A , treba napisati u već otvorenu binarnu datoteku, zadanu pokazivačem `b`, tako da se prvo napišu dva cijela broja k i n , a zatim svi elementi te podmatrice B_k , redak po redak.

- (b) Napišite program koji s komandne linije učitava dva argumenta. Prvi argument je ime ulazne tekstualne datoteke, a drugi je ime izlazne binarne datoteke. Program treba otvoriti obje datoteke, pozvati funkciju iz (a) i ispisati njezin rezultat, te zatvoriti obje datoteke.

Napomena: Smijete koristiti maksimalno jedno (dvodimenzionalno) polje za rješavanje zadatka. Oslobodite svu memoriju koju alocirate. Ignorirajte moguće greške kod čitanja argumenata i rada s datotekama.