

PRAKTIKUM IZ PROGRAMIRANJA 2

- domaći zadatak broj 4 -

Sastaviti program na programskom jeziku C koji vrši određenu vrstu obrade nad jednom ili više jednostruko ulančanih lista. Program treba da omogući učitavanje svih potrebnih ulaznih podataka, njihovo ispisivanje, obradu, ispis dobijenih rezultata, i ponavljanje navedenih koraka sve dok korisnik ne unese vrednost koja označava kraj programa. **Navedene korake u izvršavanju programa realizovati kao zasebne funkcije.**

Zavisno od rednog broja problema, sastaviti **jedan** od sledećih programa, koji:

- vrši operacije nad skupovima predstavljenim pomoću listi; jedan element skupa odgovara jednom elementu liste; potrebno je podržati operacije **preseka**, **unije** i **razlike** i njihovu primenu na dva skupa koja korisnik unosi preko tastature;
- nalazi i ispisuje svaku trojku kolinearnih tačaka iz liste tačaka u ravni; element liste treba da sadrži koordinate tačke (**x** i **y**); tri tačke su kolinearne ako važi:

$$\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} = 0$$

- nalazi trougao najveće površine među svim trouglovima čija se temena nalaze u listi tačaka; element liste treba da sadrži koordinate tačke; površina trougla se računa Heronovom formulom: $P = \sqrt{p(p-a)(p-b)(p-c)}$, $p = \frac{a+b+c}{2}$, gde su **a**, **b** i **c** dužine stranica trougla;
- vrši operacije nad binarnim brojevima predstavljenim pomoću listi; jedna binarna cifra odgovara jednom elementu liste; glava liste odgovara najstarijoj binarnoj cifri; potrebno je podržati bitske operacije **AND** (i), **OR** (ili) i **XOR** (isključivo ili); navedene operacije primeniti na dva označena cela broja koja korisnik unosi sa tastature;
- nalazi kružnicu koja sadrži najviše drugih kružnica iz liste kružnica; kružnica je predstavljena koordinatama centra i dužinom poluprečnika;
- vrši operacije nad znakovnim nizom (stringom) predstavljenim pomoću liste; jedan znak stringa odgovara jednom elementu liste; potrebno je podržati **određivanje dužine** stringa i **nadovezivanje** stringa drugim stringom (char*);

Sve funkcije smestiti u odgovarajuće **.c** datoteke (prema donjem spisku), a prototipove svih funkcija smestiti u zajedničku **.h** datoteku. Učitavanje liste/lista realizovati pomoću funkcije kojoj će kao argument biti dostavljena adresa pokazivača na početak liste i broj elemenata koji treba učitati, a koja preko povratne vrednosti vraća podatak o uspešnosti učitavanja. Funkcijama koje vrše obradu treba dostaviti samo neophodne podatke (pokazivač na početak liste i, po potrebi, podatke od kojih zavisi obrada). Potrebno je napisati funkcije koje vrše ispisivanje liste, brisanje liste, ubacivanje elementa na početak i na kraj liste i izbacivanje proizvoljnog elementa iz liste. Napraviti interaktivni meni kojim se omogućava učitavanje liste, brisanje liste, izbacivanje elementa iz liste, ubacivanje elemenata na početak i kraj liste, ispisivanje liste, obrada liste i prekidanje programa. **Ukoliko program radi sa više listi (varijante 0 i 3) obezbediti posebne stavke menija za svaku listu. Ukoliko program radi samo sa jednom listom bez nadovezivanja (varijante 1, 2 i 4), dodati u meni stavku koja učitava još jednu listu istih podataka i nadovezuje je na prvobitno unetu. Ukoliko se obrada sastoji iz više celina (varijante 0, 3 i 5) obezbediti posebne stavke menija za svaku celinu.** Voditi računa o pravilnom alociranju i dealociranju dinamičke memorije. **Potprogrami ne smeju komunicirati pomoću globalnih promenljivih, već samo preko liste argumenata i povratne vrednosti.**

Ako nešto u postavci zadatka nije dovoljno precizno definisano ili ako su neki od zahteva međusobno suprotstavljeni, usvojiti razumnu pretpostavku i rešiti zadatak korišćenjem te pretpostavke. Odabrati nekoliko skupova podataka sa kojima će program biti testiran. Odabrane test primere priložiti na listu papira pre odbrane. Kandidati koji ne na odbrani nemaju spremna makar tri suštinski različita test primera ne mogu dobiti maksimalan broj poena. Programski kod rešenja

zadatka treba da bude uredno komentarisano, tako da pri pregledu programa lako može biti uočeno šta radi bilo koja programska celina. Takođe, treba poštovati pravila nazublivanja (identacije) određenih celina prilikom pisanja koda.

Važno: po uspešnom rešavanju osnovnog zadatka, potrebno je omogućiti ispisivanje bitova binarne predstave podatka tipa float. U zavisnosti od dodeljenog problema, treba omogućiti ispis bitova proizvoljnog realnog broja koji već postoji u programu (npr. rezultat neke obrade ili neki od unetih podataka). Opciju za ispis bitova binarne predstave podatka tipa float dodati u meni kao jednu od mogućnosti. Za funkciju koja ostvaruje opisanu obradu dodati posebnu .h i posebnu .c datoteku.

Napomene:

1. Odbrana četvrtog domaćeg zadatka je u lab. 26, u četvrtak, 21. maja, prema rasporedu koji je dostupan preko sistema WebLab.
2. Formula za redni broj problema **i** koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa): **$i = (R + G) \bmod 6$**
3. Kao rešenje domaćeg zadatka potrebno je na odbrani pokazati sledeće datoteke:
 - **dz4.c**, koja sadrži izvorni tekst osnovnog programa na programskom jeziku C;
 - **dz4.h**, koja sadrži prototipove svih funkcija opisanih u postavci zadatka;
 - **dz4_unos.c**, **dz4_ispis.c**, **dz4_obrada.c**, koje sadrže izvorne tekstove potrebnih funkcija;
 - **dz4.vcproj**, koja sadrži informacije o projektu koji se odnosi na osnovni program;
 - **dz4_dodatak.h**, koja sadrži prototip funkcije za ispis bitova realnog broja,
 - **dz4_dodatak.c**, koja sadrži telo funkcije za ispis bitova realnog broja,
 - **dz4_dodatak.vcproj**, koja sadrži informacije o projektu koji se odnosi na dodatni program;
 - **dz4.sln**, koja sadrži informacije o svim projektima relevantnim za ovaj zadatak.

12.05.2009. godine

sa predmeta