

## 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:

0. izračunava vrednost polinoma predstavljenog listom koeficijenata (za polinom  $a_n x^n + a_{n-1} x^{n-1} + \dots + a_0$  u listi su redom smešteni  $a_n, a_{n-1}, \dots, a_0$ ); funkciji koja računa vrednost polinoma treba preneti samo vrednost **x** i pokazivač na početak liste;
1. izračunava vrednost broja predstavljenog listom cifara brojnog sistema sa osnovom **q** ( $q \leq 16$ ); lista cifara može sadržati tačku koja razdvaja celobrojni i razlomljeni deo broja; prvi element liste sadrži cifru najveće težine;
2. izračunava zbir dva polinoma predstavljenih pomoću jednostruko ulančanih lista, čiji elementi sadrže stepen promenljive i koeficijent polinoma (polinomi ne moraju biti istog reda i liste **ne moraju biti uređene** po stepenima promenljive); zbir treba predstaviti na isti način kao sabirke; vrednost dobijenog polinoma ne treba računati;
3. izračunava proizvod dva polinoma predstavljenih pomoću jednostruko ulančanih lista, čiji elementi sadrže stepen promenljive i koeficijent polinoma (polinomi ne moraju biti istog reda, liste **jesu uređene** po stepenima promenljive); proizvod predstaviti na isti način kao činioce; vrednost dobijenog polinoma ne treba računati;
4. izračunava vrednost broja predstavljenog listom rimskih cifara; rimske cifre koje treba podržati su I,V,X,L,C,D,M;
5. pronalazi i vrši spajanje susednih slobodnih memorijskih blokova; svi memorijski blokovi su predstavljeni ulančanom listom; element liste sadrži adresu početka bloka, veličinu memorije koju blok zauzima u bajtovima i polje koje označava da li je blok slobodan ili ne;

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 i prekidanje programa. 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. Odabрати 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 nazubljanja (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 koeficijenta polinoma, rezultata obrade, vrednosti realnog podatka ili broja učitano sa glavnog ulaza. Opciju za ispis bitova binarne predstave podatka tipa float dodati u meni kao jednu od mogućnosti.**

### **Napomene:**

1. Odbrana četvrtog domaćeg zadatka je u RC ETF, u četvrtak, 12. juna, 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\_ubl.vcproj**, koja sadrži informacije o projektu koji se odnosi na dodatni program;
  - **dz4.sln**, koja sadrži informacije o svim projektima relevantnim za ovaj zadatak.

28.05.2008. godine

*sa predmeta*