

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 prema dole navedenom rasporedu.**

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

0. pomaže korisniku u organizaciji sastanaka u toku jednog dana; ulančana lista sadrži podatke o terminima sastanaka u toku jednog dana; program treba da ispiše sve termine sastanaka koji se preklapaju i da formira i ispiše ulančanu listu sastanaka na koji korisnik treba da ode u okviru zadatog dana; sastanak je određen mestom (zgradom) i salom održavanja sastanka i vremenom početka i kraja; ako se dva ili više sastanaka preklapaju, prednost dati onome koji se odvija na istom mestu kao i prethodni, a u slučaju istog mesta održavanja, prioritet dati onom koji počinje najranije;
1. raspoređuje novinske članke i reklame iz ulančane liste po stranicama novina; podaci o članku se sastoje od imena i procenta prostora na stranici koji zauzima; podaci o reklamama se sastoje od imena oglašivača; svaka novinska stranica može sastojati od članaka čije se dimenzije ne mogu menjati i reklama koje se mogu dimenzionisati da popune ostatak stranice; program treba da formira novinske stranice tako da se na stranicu uvek dodaje najveći članak koji može da stane, sve dok ima odgovarajućih članaka u listi, a zatim prostor dopunjava reklamama; novu stranicu formirati kada postojeća ne može da se dopuni preostalim člancima iz liste; rezultat obrade smestiti u ulančanu listu koja sadrži stranice sa imenima članaka i reklama koje se nalaze na svakoj od njih; ukoliko korisnik ne zada dovoljno reklama, ubaciti na stranicu generičku reklamu sa naslovom: „Prostor za vašu reklamu“;
2. obrađuje ulančanu listu koja sadrži informacije o naučnim člancima; naučni članak je određen jedinstvenim identifikacionim brojem, naslovom članka, naslovom časopisa u kome je objavljen i godinom izdanja; potrebno je za svaki časopis koji se pojavljuje u listi formirati listu njegovih članaka uređenu rastuće po godini izdavanja članaka, a u okviru iste godine leksikografski po naslovu članaka;
3. obrađuje ulančanu listu koja sadrži informacije o opasnim materijama koje se nalaze u okviru nekog skladišta; svaki kontejner koji sadrži opasnu materiju je opisan jedinstvenim identifikacionim brojem, težinom u kilogramima, klasom opasnih materija (ceo broj od 1 do 9) i UN brojem opasne materije; potrebno je za svaku klasu opasnih materija formirati listu kontejnera koji sadrže takvu vrstu materije, uređenu po težini kontejnera i odrediti kojih opasnih materija (prema UN broju) ima količinski najviše po klasama;
4. simulira rad blagajne nekog kluba za prodaju karata za utakmicu; blagajna sadrži spisak ljudi u vidu ulančane liste koji žele da kupe kartu; podatak o jednom čoveku su ime, e-mail adresa, broj karata koje želi da kupi i oznaku da li se radi o članu kluba; blagajna prodaje karte u koracima; najpre se unese maksimalan broj raspoloživih karata; zatim se karte raspodeljuju na sledeći način: najpre se po jedna karta podeli članovima kluba, a zatim se vrši podela preostalih karata prema traženim količinama po redosledu navedenom u ulančanoj listi; raspodela karata se završava kada se sve karte podele ili kada na blagajni nema više karata za prodaju; na kraju podele za svakog prijavljenog ispisati podatke i broj karata koje je dobio;
5. obrađuje ulančanu listu znamenitosti u nekom gradu; podaci o znamenitosti se sastoje od imena, radnog vremena (početno i krajnje vreme u satima i minutima), vremena potrebnog za obilazak i cene ulaznice; potrebno je formirati novu listu koja sadrži spisak znamenitosti koje turista može da obiđe tog dana i ukupnu cenu obilaska, a da se pritom obilasci vremenski ne preklapaju;

Sve funkcije smestiti u odgovarajuće **.c** datoteke (prema donjem spisku), a prototipove svih funkcija smestiti u zajedničku **.h** datoteku. Funkcijama koje vrše učitavanje i 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 obezbediti posebne stavke menija (koje su potrebne) za svaku listu. Ukoliko se obrada sastoji iz više koraka obezbediti posebne stavke menija za svaki korak.** 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 na odbrani nemaju spremna makar tri suštinski različita test primera ne mogu dobiti maksimalan broj poena. Programski kod rešenja zadatka treba da bude uredno komentaran, tako da pri pregledu programa lako može biti uočeno šta radi bilo koja programska celina. Takođe, treba poštovati pravila nazubljenja (identacije) određenih celina prilikom pisanja koda.

Napomene:

1. Odbrana četvrtog domaćeg zadatka je u četvrtak, 30.05.2019. po rasporedu dostupnom na sajtu predmeta.
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, glavnog 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.

21.05.2019. godine

sa predmeta