

PRAKTIKUM IZ PROGRAMIRANJA 1

- domaći zadatak broj 5 -

Sastaviti program na programskom jeziku Pascal koji vrše zadatu obradu. Zavisno od rednog broja problema koji se rešava, sastaviti sledeći program i pomoćni program za formiranje potrebnih binarnih datoteka:

1. Upisna administracija: Program treba da izračuna koliko bruceša koji su upisali Elektrotehnički fakultet je prethodno maturiralo u nekoj od srednjih škola sa spiska. Na raspolaganju su dve **binarne** datoteke: *brucosi.bin* i *skole.bin*. Jedan zapis u datoteci *brucosi.bin* se odnosi na jednog bruceša i sadrži šifru škole koju je završio (ceo broj), ime i prezime (po 20 znakova), godina upisa na fakultet i registarski broj (dva cela broja). Jedan zapis u datoteci *skole.bin* se odnosi na jednu školu i sadrži šifru škole i naziv škole (najviše 40 znakova). Program najpre treba da čita podatke iz navedenih datoteka i na osnovu njih formira dve ulančane liste, jednu za škole i jednu za bruceše. Program zatim treba da ponudi korisniku izbor vrste izveštaja koji treba napraviti. Predvideti sledeće izveštaje:
 - a. pregledni izveštaj za sve škole u kome se navode podaci za svaku školu koju je završio najmanje jedan bruceš. Podaci su sledeći: šifra škole, naziv škole, broj bruceša koji su završili tu školu (apsolutni iznos i procenat u odnosu na ukupan broj bruceša). Podatke za sve bruceše za koje škola koju su završili nije evidentirana u datoteci *skole.bin* ispisati nakon ispisa svih škola pod stavkom "ostalo".
 - b. detaljan izveštaj za jednu školu u kome se, pored informacija navedenih u prethodnoj tački, nalaze podaci o brucešima (ime, prezime, broj indeksa) koji su završili datu školu. Izbor škole korisnik vrši unosom šifre škole. Šifra 0 je rezervisana za stavku "ostalo" iz prethodne tačke.Obezbediti da se izveštaji mogu ispisivati na standardnom izlazu ili snimati u **tekstuelne** datoteke čiji naziv zadaje korisnik.
2. Hotel: Program treba da proveri prispele zahteve za rezervaciju soba u hotelu i da napravi potvrdu rezervacije odnosno obaveštenje da data rezervacija nije moguća. Opis trenutnog stanja u hotelu se nalazi u **binarnoj** datoteci *hotel.bin*. Svaki zapis u datoteci se odnosi na jednu sobu u hotelu, a soba ima onoliko koliko ima zapisa. Podaci o sobi koji se beleže u datoteci su sledeći: oznaka sobe, broj kreveta, da li sadrži luksuznu opremu, status koji može biti *slobodna*, *rezervisana*, *zauzeta* i *renoviranje*. Program treba da pročita podatke iz datoteke *hotel.bin* i da od tih podataka formira ulančanu listu soba. Zatim program treba da čita **binarnu** datoteku *rezervacije.bin* u kojoj se nalaze podaci o zahtevanim rezervacijama soba. Svaki zapis u datoteci se odnosi na jednu rezervaciju, a sadrži sledeće podatke: šifra rezervacije, potreban broj kreveta, potrebna luksuzna oprema. Za svaku pročitlanu rezervaciju program treba da proveri da li je odgovarajuća soba slobodna. Program treba da snima rezultate provere u dve **tekstuelne** datoteke *potvrda.txt* i *odbijeno.txt*, u zavisnosti od ishoda provere. U obe datoteke svaki red sadrži podatke o jednoj rezervaciji i to šifru rezervacije i oznaku sobe za koju je rezervacija izvršena.
3. Firma za isporuku paketa: Program treba da razvrsta prispele pakete prema prioritetu njihove isporuke a zatim izvrši njihovo raspoređivanje u transportna vozila. Podaci o prispelim paketima se nalaze u **binarnoj** datoteci *paketi.bin*, u kojoj svaki zapis sadrži podatke o jednom paketu: identifikacioni broj, zapremina, prioritet (od 1 do 3, gde je 3 najveći prioritet). Program treba da učita sve pakete i smesti ih u jednu ulančanu listu. Nakon toga, program treba da izvrši razvrstavanje paketa tako što će ih smestiti u jednu od tri ulančane liste, prema njihovom prioritetu (svi paketi istog prioriteta treba da budu smešteni u jednu ulančanu listu). Nakon toga, program treba da čita **binarnu** datoteku *vozila.bin* u kojoj se nalaze podaci o vozilima kojima firma raspolaže. Svaki zapis u

datoteci se odnosi na jedno vozilo i sadrži sledeće podatke: registarski broj vozila, korisna zapremina za transport i status koji može biti *kvar* ili *operativan*. Program treba da čita podatke o vozilima i u svako operativno vozilo raspoređi pakete. Najpre se raspoređuju paketi većeg prioriteta, a za sve pakete istog prioriteta raspoređivanje se vrši sve dok ima mesta u vozilu. Podatke o izvršenom raspoređivanju snimati u tri **tekstuelne** datoteke *rasp_n.txt* (n = 1, 2 ili 3) u kojoj svaki red sadrži podatke o rasporedu paketa u jednom vozilu: registarski broj vozila a zatim identifikacione brojeve raspoređenih paketa. Na kraju datoteke treba ispisati informaciju o tome da li su svi paketi uspešno raspoređeni ili je potrebno angažovati dodatna vozila. Smatrati da nema paketa koji ne može da se smesti u bilo koje vozilo.

Napomene:

1. **Obavezno** treba koristiti potprograme prilikom sastavljanja programa. Takođe, obavezno treba osloboditi svu dinamički alociranu memoriju i zatvoriti sve datoteke nakon njihove upotrebe (čitanje ili pisanje).
2. Odrbane petog domaćeg zadatka će se održati u sedmici od 14.01. do 18.01. u sali 60 prema rasporedu će biti objavljen na posebnom obaveštenju. Studenti treba da se prijave za odbranu zadatka korišćenjem sistema Weblab. **Obavezno pročitati uputstvo na predmetnom veb sajtu u vezi prijavljivanja aktivnosti.**
3. Priznavaće se samo oni radovi koje studenti unesu u računar na licu mesta. Gotovi radovi, donešeni na bilo kojoj vrsti medijuma (CD, USB disk, itd) će biti odbijeni. Na odbrani treba pokazati dva programa: pomoćni program, pomoću kojeg se prave binarne datoteke koje će se koristiti u drugom, glavnom programu.
4. Formula za redni broj problema **i** koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa):

$$i = (R + G) \bmod 3 + 1$$

30.12.2007. godine

sa predmeta