

PRAKTIKUM IZ PROGRAMIRANJA 1

- domaći zadatak broj 5 –

- Pascal grupa -

Sastaviti program na programskom jeziku Pascal koji radi sa listama i datotekama na sledeći način:

- a) poziva potprogram koji učitava sve potrebne podatke iz ulazne tekstualne datoteke u jednostruko ulančanu listu, tako da redosled podataka u listi bude isti kao u datoteci;
- b) poziva odgovarajući potprogram koji izvrši traženu obradu nad datom listom;
- c) poziva potprogram koji u datoteku upisuje sve rezultate dobijene u toku obrade;
- d) poziva potprogram koji listu briše iz memorije;
- e) ponavlja prethodne korake dok korisnik za ime ulazne datoteke ne unese prazan string.

Zavisno od rednog broja problema, program treba da izvrši sledeću obradu:

0. čita tekstualnu datoteku u kojoj se u svakom redu nalaze sledeće informacije o studentu: godina upisa, redni broj indeksa, ime, prezime, profil (OO ili SI), programski jezik na P1 (Pascal ili Python), da li sluša PP1 (DA ili NE) i programski jezik na PP1 (Pascal ili Python); iz liste izbaciti studente sa neispravnim podacima i od tih podataka kreirati tekstualnu datoteku; svaki red datoteke treba da sadrži redni broj greške i broj indeksa studenta u formatu gg/bbbb; redni brojevi grešaka su 0 (student sa OO sluša Python na P1), 1 (student sa SI ne sluša PP1), 2 (student sa SI sluša različite jezike na P1 i PP1); datoteka sa greškama je uređena rastuće po broju indeksa studenta.
1. čita tekstualnu datoteku u kojoj se u svakom redu nalaze sledeće informacije o rezultatima kolokvijuma: indeks studenta u formatu gg bbbb, poeni na K1, poeni na K1P, poeni na K2_PAS, poeni na K2_PY; ako se polje ne odnosi na datog studenta upisana je vrednost -1, a ako student nije izašao na obavezu upisana je vrednost -2; iz informacija smeštenih u listi pravi statistiku i u zavisnosti od zahteva na standardnom izlazu prikazuje podatke; pre svakog zahteva se bira da li se prikazuje izlaznost (I) ili prosek (P); zahtevi su K1 (izlaznost ili prosek na K1), K1P (izlaznost ili prosek na K1P), K2_PAS (izlaznost ili prosek na K2_PAS), K2_PY (izlaznost ili prosek na K2_PY); zahtevi se obrađuju i statistika prikazuje sve dok se ne unese nepostojeći zahtev.
2. čita tekstualnu datoteku kojoj se u svakom redu nalaze sledeće informacije o poenima na kolokvijumu: indeks studenta u formatu gggg bbbb, tip kolokvijuma (K1_OO, K1_SI, K1_P), sedam pitalica (za svaku vrednost -1, 0 ili 1), zadatak (0-100), ukupan broj poena; ako je pitalica netačna upisana je vrednost -1, ako je tačna 1, a ako student nije odgovorio upisana je vrednost 0; izračunava ukupne poene po odgovarajućoj formuli (polja koja su višak za datu formulu se ignorišu); formule su: 7 pitalica (5p tačna, -1.25p netačna, 30p max) i 0 zadataka za K1_OO, 6 pitalica (12p tačna, -3p netačna, 60p max) i 1 zadatak (40p max) za K1_SI, 4 pitalice (15p tačna, -3.75p netačna, 45p max) i 1 zadatak (55p max) za K1_P; zbir poena na pitalicama i zadatku čini ukupan broj poena; iz liste izbaciti studente koji su osvojili manje od 50% ukupnog broja bodova, a zatim u tekstualnu datoteku upisati preostale studente po nerastućem poretku dobijenih poena; u datoteku upisati indeks studenta u formatu bbbb/gggg i osvojene poene.
3. čita tekstualnu datoteku u kojoj se u svakom redu nalaze sledeće informacije o studentu: godina upisa, redni broj indeksa, ime, prezime, profil (OO ili SI), programski jezik na P1 (Pascal ili Python), da li sluša PP1 (DA ili NE) i programski jezik na PP1 (Pascal ili Python); iz liste kreira dve nove tekstualne datoteke u kojima se nalaze podaci o studentima sa profila OO i SI respektivno; datoteka sa informacijama o studentu sa OO smera u jednom redu sadrži: indeks u formatu bbbb/gg, ime i prezime, jezik koji sluša na PP1 (ne sluša, PAS, Py); datoteka sa informacijama o studentu sa SI smera u jednom redu sadrži: indeks u formatu gggg/bbbb, prezime i ime, jezik koji sluša (PAS, Py); datoteke su uređene po programskom jeziku i po prezimenu i imenu studenta.
4. čita tekstualnu datoteku u kojoj se u svakom redu nalaze sledeće informacije o poenima na kolokvijumu: indeks studenta u formatu gggg bbbb, tip kolokvijuma (K1_OO, K1_SI, K1_P), sedam pitalica (za svaku vrednost -1, 0 ili 1), zadatak (0-100), ukupan broj poena; potrebni podaci su 7 pitalica za K1_OO, prvih 6 pitalica i 1 zadatak za K1_SI, prve 4 pitalice i 1 zadatak za K1_P; iz liste izbaciti studente sa neispravnim podacima i od tih podataka kreirati tekstualnu datoteku; svaki red

datoteke treba da sadrži redni broj greške i broj indeksa studenta u formatu bbbb/gggg; redni brojevi grešaka su 0 (pitalica koja se ne razmatra za dati kolokvijum ima vrednost različitu od 0), 1 (zadatak koji se ne razmatra na datom kolokvijumu ima vrednost različitu od 0), 2 (pitalici je dodeljena vrednost različita od -1, 0, 1); datoteka sa greškama je uređena rastuće po rednom broju greške.

5. čita dve tekstualne datoteke, jednu u kojoj se u svakom redu nalaze sledeće informacije o studentu: godina upisa, redni broj indeksa, ime, prezime, profil (OO ili SI), programski jezik na P1 (Pascal ili Python), da li sluša PP1 (DA ili NE) i programski jezik na PP1 (Pascal ili Python) i drugu u kojoj se u svakom redu nalaze sledeće informacije o rezultatima kolokvijuma: indeks studenta u formatu gg bbbb, poeni na K1, poeni na K1P, poeni na K2_PAS, poeni na K2_PY; iz liste izbaciti studente sa neispravnim podacima i od tih podataka kreirati tekstualnu datoteku; svaki red datoteke treba da sadrži redni broj greške i broj indeksa studenta u formatu bbbb/gg; redni brojevi grešaka su 0 (student sa OO izasao na K1P, K2_PAS ili K2_PY), 1 (student sa SI izasao na pogrešan programski jezik), 2 (student sa SI nije izasao na pravi programski jezik).
6. čita tekstualnu datoteku u kojoj se u svakom redu nalaze sledeće informacije o poenima na kolokvijumu: indeks studenta u formatu gggg bbbb, tip kolokvijuma (K1_OO, K1_SI, K1_P), sedam pitalica (za svaku vrednost -1, 0 ili 1), zadatak (0-100), ukupan broj poena; iz liste kreira tri nove tekstualne datoteke formata u kojima se nalaze podaci o kolokvijumima K1_OO, K1_SI i K1_P respektivno; datoteka sa informacijama o K1_OO u jednom redu sadrži: indeks u formatu bbbb/gggg, 7 pitalica, ukupan broj poena; datoteka sa informacijama o K1_SI u jednom redu sadrži: indeks u formatu bbbb/gg, prvih 6 pitalica, zadatak, ukupan broj poena; datoteka sa informacijama o K1_P u jednom redu sadrži: indeks u formatu gg/bbbb, prve 4 pitalice, zadatak, ukupan broj poena; datoteke su uređene rastuće po broju indeksa studenta.
7. čita tekstualnu datoteku u kojoj se u svakom redu nalaze sledeće informacije o studentu: godina upisa, redni broj indeksa, ime, prezime, profil (OO ili SI), programski jezik na P1 (Pascal ili Python), da li sluša PP1 (DA ili NE) i programski jezik na PP1 (Pascal ili Python); iz informacija smeštenih u listi pravi statistiku i u zavisnosti od zahteva na standardnom izlazu prikazuje podatke; zahtevi su OO (procenat studenata koji su na profilu OO), SI (procenat studenata koji su na profilu SI), OO_PP1 (procenat studenata sa OO koji sluša PP1), SI_PAS (procenat studenta sa SI koji sluša Pascal na P1), SI_PY (procenat studenata sa SI koji sluša Python na P1); zahtevi se obrađuju i statistika prikazuje sve dok se ne unese nepostojeći zahtev.

Elementi liste su zapisi koji imaju dva polja. Prvo polje je tipa zapisa, čija polja sadrže sve bitne podatke, prema tekstu zadatka. Drugo polje je pokazivač na sledeći element liste. Format izlazne tekst datoteke usvojiti prema tekstu zadatka, tako da sadrži relevantne, ali ne i nepotrebne informacije. Imena ulazne i izlazne datoteke učitavati u glavnom programu. 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.

Osmisliti i napisati na papiru nekoliko primera ulaznih podataka sa kojima će se program testirati, kao i očekivani izlaz za odabrane primere. Makar jedan od test primera mora imati više od 5 redova. **Potprogrami ne smeju pristupati promenljivama glavnog programa direktno, već samo putem svojih argumenata.**

Napomene:

1. Odbrana petog domaćeg zadatka je u sredu, 25.12.2019. Tačni termini za sve studente su objavljeni 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):
$$i = (R + G) \bmod 8$$
3. Kao rešenje domaćeg zadatka potrebno je na odbrani pokazati sledeće datoteke:
 - **dz5.PAS**, koja sadrži izvorni tekst osnovnog programa na programskom jeziku Pascal;
 - **dz5_u1.txt, ..., dz5_uN.txt**, koji sadrže nekoliko odabranih test primera kojima će program biti testiran.