

## PRAKTIKUM IZ PROGRAMIRANJA 1

### - domaći zadatak broj 5 -

#### - Python grupa -

Sastaviti program na programskom jeziku Python, koji radi sa **datotekama i složenim strukturama podataka** po sledećem redosledu izvršavanja:

- učitava nazive jedne ili više datoteka sa kojima će se raditi,
- poziva odgovarajuću funkciju koja čita datoteke i izvršava traženu obradu,
- ponavlja prethodne korake sve dok korisnik ne zada prazan tekst kao naziv datoteke.

Zavisno od rednog broja problema, sastaviti program, koji obavlja sledeću obradu:

- čita datoteku formata .txt u kojoj se u svakom redu nalaze sledeće informacije o studentu odvojene separatorom tačka-zarez (;): 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); pronalazi redove sa neispravnim informacijama i od tih podataka kreira datoteku formata .log; svaki red log fajla 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.
- čita datoteku formata .csv 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 pročitanih informacija 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.
- čita datoteku formata .csv 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; 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; za svakog studenta se na standardnom izlazu ispisuje ukupan broj poena pri uređen po tipu kolokvijuma i opadajuće po ukupnom broju poena.
- čita datoteku formata .txt u kojoj se u svakom redu nalaze sledeće informacije o studentu odvojene separatorom tačka-zarez (;): 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); kreira dve nove datoteke formata .csv u kojima se nalaze podaci o studentima sa profila OO i SI respektivno; datoteka sa informacijama o studentu sa OO smeru 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 smeru 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.
- čita datoteku formata .csv 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; pronalazi redove sa neispravnim informacijama i od tih podataka kreira datoteku

formata .log; svaki red log fajla 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 datoteke, jednu formata .txt u kojoj se u svakom redu nalaze sledeće informacije o studentu odvojene separatorom tačka-zarez (;): 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 formata .csv 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; pronalazi neispravne informacije i od tih podataka kreira datoteku formata .log; svaki red log fajla 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 datoteku formata .csv 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; kreira tri nove datoteke formata .txt 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 datoteku formata .txt u kojoj se u svakom redu nalaze sledeće informacije o studentu odvojene separatorom tačka-zarez (;): 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 pročitanih informacija 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.

Za čuvanje podataka koristiti strukture podataka kao što su rečnici, skupovi, torke, gde je to moguće Program treba da čita podatke uz vođenje računa o tipu podataka koji se čita. 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. Test primere otkucati u bilo kom tekst editoru i priložiti zajedno sa tekstom programa. **Funkcije 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.py**, koja sadrži izvorni tekst programa na programskom jeziku Python;
  - **dz5\_u1.txt, ..., dz5\_uN.txt**, koje sadrže nekoliko odabranih test primera kojima će program biti testiran.