

## PRAKTIKUM IZ PROGRAMIRANJA 1

### - domaći zadatak broj 4 -

Sastaviti glavni program koji radi sa **tekst** datotekama po sledećem redosledu izvršavanja:

- poziva potprogram koji učitava imena datoteka sa kojima će se raditi,
- poziva odgovarajući potprogram koji izvrši zahtevanu obradu nad ulaznim podacima,
- poziva potprogram koji ispisuje rezultate izvršavanja obrade (imena datoteka i broj obrađenih redova svake datoteke), i
- ponavlja prethodne korake sve dok se za ime datoteke na zada prazan string.

Zavisno od rednog broja problema koji se rešava, sastaviti **jedan** od sledećih potprograma na programskom jeziku Pascal, kojim se:

- prepisuje spisak studenata iz ulazne datoteke (format `IME PREZIME GGGGBBBB 12 23 34`) u novu datoteku (format `BBBB/GG PREZIME IME 7`); GGGG i GG predstavljaju godinu upisa, odnosno poslednje dve cifre godine upisa, tri cela broja ulazne datoteke uspeh na delovima ispita, celi broj u izlaznoj datoteci konačnu ocenu koja se određuje prema zbiru poena na sva tri dela (51 poen za 6, 61 poen za 7, maksimalan broj poena 100)
- prepisuje spisak studenata iz ulazne datoteke (format `BBBB/GG PREZIME IME 6`) u dve nove datoteke (format `BBBBGGGG Ime Prezime`); smisao podataka je isti kao kod 0. problema, a izlazna datoteka se određuje zavisno od toga da li je student položio ispit ili ne.
- iz ulazne datoteke u kojoj se nalaze podaci o boji, gustini i prečniku kuglica (primer: `crvena 123.4 56.789`) u novu prepisuju samo podaci o kuglicama mase veće od MAX. Podaci o ostalim kuglicama se ispisuju na standardnom izlazu u okviru ove obrade.
- prebrojavaju reči, rečenice i znaci interpunkcije (`. , ! ?`) u ulaznoj tekst datoteci. Reči su razdvojene razmakom, krajem reda ili znakom interpunkcije. Rečenice su razdvojene znakom interpunkcije. Rečenice sa više od 5 reči se prepisuju u novu datoteku.
- podaci o stanju robe u magacinu (šifra, količina, cena, naziv; primer: `BS01 123 45 Bebi sapun`) iz dve tekst datoteke smeštaju u novu datoteku istog formata uz ažuriranje količine za one šifre koje se nalaze u obe datoteke. Podaci u ulaznim datotekama su sortirani rastuće po šifri proizvoda.

Program treba da čita ulaznu datoteku uz vođenje računa o tipu podataka koji se čita. Očuvati prvobitnu uređenost teksta po redovima pri prepisivanju u nove datoteke. Dužine redova smatrati manjim od 80. 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. **Potprogrami ne smeju pristupati promenljivama glavnog programa direktno, već samo putem svojih argumenata.**

**Važno: za sve podatke koje ima smisla grupisati, osmisliti sopstveni tip zapisa i koristiti promenljive tog tipa, tamo gde je to moguće.**

Napomene:

- Raspored termina za odbranu četvrtog domaćeg zadatka je objavljen na Internet stranici predmeta ([http://rti7020.etf.bg.ac.yu/rti/ir1pp1/spiskovi/dz4\\_raspored.htm](http://rti7020.etf.bg.ac.yu/rti/ir1pp1/spiskovi/dz4_raspored.htm)).
- Formula za redni broj problema **i** koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa):  **$i = (R + G) \bmod 5$**
- Kao rešenje domaćeg zadatka potrebno je na odbrani pokazati sledeće datoteke:
  - dz4.PAS, koja sadrži izvorni tekst osnovnog programa na programskom jeziku Pascal;