

Programski jezici 2 - 2246

14.02.2019.

1. **(30)** Napisati aplikaciju za obradu poslovnih podataka. Podaci koje aplikacija treba analizirati nalaze se u fajlu na Moodle stranici predmeta. Aplikacija treba da radi sa narudžbenicama i fakturama. Narudžbenice imaju šifru, datum kupovine (*string*) i niz stavki koje imaju šifru narudžbenice naziv proizvoda, količinu i cijenu po komadu. Fakture imaju šifru, naziv kupca, ukupan iznos i niz stavki koje imaju šifru fakture, naziv proizvoda, količinu i cijenu po komadu. Napraviti generički interfejs koji omogućava da se iz fajla sa podacima ulazni string pretvori u objekat za sve tipove podataka koje se čuvaju (parsiranje). Pretvaranje stringa u objekat implementirati u svakoj klasi modela gdje je to potrebno. Osim toga, potrebno je napraviti i enumeraciju koja ima elemente koji označavaju tip podatka sa mogućim vrijednostima „*Narudzbenica*“, „*NarudzbenicaStavka*“, „*Faktura*“, i „*FakturaStavka*“.

U glavnom dijelu programa potrebno je napraviti parsiranje ulaznog CSV fajla i kreiranje listi objekata odgovarajućeg tipa. Za parsiranje se koriste dvije niti: jedna za narudžbenice, a druga za fakture. Rezultat parsiranja su odgovarajuće liste koje će se dalje koristiti u glavnom dijelu programa. Niti se pokreću istovremeno, obrađuju liniju po liniju i grupišu podatke u jedan objekat sa stavkama za svaki tip (npr. sve stavke i detalji fakture se parsiraju kao jedan objekat i dodaju u kolekciju). Tip objekta se prepoznaje na osnovu prefiksa linije i enumeracije. Kreiranje jednog objekta treba trajati 1s.

Korisnik može tokom obrade podataka pauzirati izvršavanje unosom naredbe „*WAIT*“, koja zaustavlja izvršavanje obrade podataka sve dok korisnik ne unese naredbu „*GO*“. Kada sve niti završe obradu podataka ispisuje se odgovarajuća poruka i broj obrađenih objekata svake vrste.

2. **(30)** U rješenje prvog zadatka potrebno je dodati element za validaciju obrađenih podataka. Klasa validator treba da ima samo jednu generičku metodu koja za proslijeđeni objekat izvršava niz pravila validacije koji su u obliku *Predicate*-a. Metoda vraća *boolean* vrijednost ako su sva pravila ispunjena.

Upotrebom validatora izvršiti validaciju dobijenih nizova tako da se u nove nizove upišu fakture i narudžbenice sa greškom. Faktura je validna ako je ukupan iznos fakture jednak sumi proizvoda polja količina i cijene po komadu kod stavki, i ako je broj stavki najmanje jedan. Narudžbenica je validna ako je broj stavki najmanje jedan i ako je ukupan iznos svake stavke pozitivan.

Nakon validacije ispisati broj objekata koji nisu validni i sve njihove šifre za oba tipa podataka. Sortirati sve validne fakture po ukupnom iznosu za plaćanje u opadajućem redoslijedu i ispisati 5 najskupljih. Ove operacije izvesti upotrebom *Stream* API-ja.

3. **(10)** U tekstualni fajl upisati broj parsiranih faktura i ukupan iznos za plaćanje, zatim broj narudžbenica i prosječan broj stavki u validnim narudžbenicama. Naziv tekstualnog fajla se unosi kao argument komandne linije prilikom pokretanja glavnog programa.

Vrijeme rada: 180 minuta.