

Programski jezici

-Laboratorijska vežba 2-

NAPOMENA: *Ovom lab vežbom je obuhvaćen materijal koji je obrađen u prva četiri termina računskih vežbi. Svaki od zadataka treba rešiti korišćenjem ili interfejsa ili apstraktne klase, u zavisnosti od sadržine zadatka. Svaka od klasa koja se kreira treba da sadrži konstruktor u kome će se postavljati vrednosti svih atributa. Atributi klase treba da budu minimalno vidljivi (najbolje private, a atribut koji mora da bude vidljiv iz izvedene klase protected). Sve zadatke odraditi na programskom jeziku Java.*

1. Nadograditi zadatak iz prethodne laboratorijske vežbe kreiranjem generičke klase Kolekcija koja treba da implementira sve neophodne metode (dodavanje, brisanje, rad sa trakama,...) i u klasi Fabrika umesto običnog niza kreirati Kolekciju traka. Ograničiti da parametarski tip Kolekcije može da bude samo Traka ili klase izvedene iz nje.

2. Nadograditi zadatak iz prethodne laboratorijske vežbe kreiranjem generičke klase Kolekcija koja treba da implementira sve neophodne metode (dodavanje, brisanje, rad sa zaposlenima,...) i u klasi Kompanija umesto običnog niza kreirati Kolekciju zaposlenih. Ograničiti da parametarski tip Kolekcije može da bude samo klasa Zaposleni ili klase izvedene iz nje. Omogućiti da kolekcija može da sortira zaposlene po vrednosti plate i testirati sortiranje u Main metodi.

3. Nadograditi zadatak iz prethodne laboratorijske vežbe kreiranjem generičke klase KvadratnaMatrica koja treba da implementira sve neophodne metode (dodavanje na zadatu poziciju, brisanje sa zadate pozicije, prikaz,...) i u klasi Sajam kreirati KvadratnuMatricu automobila.

4. Nadograditi zadatak iz prethodne laboratorijske vežbe tako da se umesto niza Takmičara u klasi Prvenstvo čuva generička lista (`ArrayList<E>` iz Javine biblioteke, paket `java.util`). Sve metode koje su do sada radile sa nizom izmeniti tako da koriste metode iz liste. Za sortiranje liste takmičara moguće je iskoristiti `sort` metodu iz klase `ArrayList<E>`. Najbolji način je da se metoda zove sa parametrom `null` (`list.sort(null)`), a prethodno obavezno modifikovati klasu `Takmičar` tako da implementira generički interfejs `Comparable<Takmičar>` iz paketa `java.lang`.

5. Nadograditi zadatak iz prethodne laboratorijske vežbe kreiranjem generičke klase Kolekcija koja treba da implementira sve neophodne metode (dodavanje, brisanje, rad sa Terminalima,...) i u klasi Vremenska stanica kreirati Kolekciju terminala. Ograničiti da parametarski tip kolekcije može da bude samo Terminal ili klase izvedene iz nje.

6. Nadograditi zadatak iz prethodne laboratorijske vežbe tako da se umesto niza Putovanja u klasi Agencija čuva generička lista (`ArrayList<E>` iz Javine biblioteke, paket `java.util`). Sve metode koje su do sada radile sa nizom izmeniti tako da koriste metode iz liste. Za sortiranje liste putovanja moguće je iskoristiti `sort` metodu iz klase `ArrayList<E>`. Najbolji način je da se metoda poziva sa parametrom `null` (`list.sort(null)`), a prethodno obavezno modifikovati klasu Putovanje tako da implementira generički interfejs `Comparable<Putovanje>` iz paketa `java.lang`.

7. Nadograditi zadatak iz prethodne laboratorijske vežbe kreiranjem generičke klase Kolekcija koja treba da implementira sve neophodne metode (dodavanje, izbacivanje iz kolekcije, rad sa objektima klase Radnik,...) i u klasi Firma umesto običnog niza kreirati Kolekciju radnika. Ograničiti da parametarski tip Kolekcije može da bude samo klasa Radnik ili klase izvedene iz nje.

8. Nadograditi zadatak iz prethodne laboratorijske vežbe kreiranjem generičke klase Matrica koja treba da implementira sve neophodne metode (dodavanje na zadatu poziciju, brisanje sa zadate pozicije, provera da li je pozicija popunjena ili ne, prikaz,...) i u klasi Parking umesto obične koristiti generičku Matricu automobila.

9. Nadograditi zadatak iz prethodne laboratorijske vežbe kreiranjem generičke klase Kolekcija koja treba da implementira sve neophodne metode (dodavanje, brisanje, rad sa objektima klase Stavka,...) i u klasi Literatura umesto običnog niza kreirati Kolekciju stavki. Ograničiti da parametarski tip Kolekcije može da bude samo klasa Stavka ili klase izvedene iz nje. Omogućiti da kolekcija može da sortira stavke po godini izdanja ili po tekstu, u rastućem ili opadajućem redosledu i testirati sortiranje u `Main` metodi.

10. Nadograditi zadatak iz prethodne laboratorijske vežbe kreiranjem generičke klase Kolekcija koja treba da implementira sve neophodne metode (dodavanje, izbacivanje iz kolekcije, rad sa objektima klase Poruka,...) i u klasi Telefon umesto običnog niza kreirati Kolekciju poruka. Ograničiti da parametarski tip Kolekcije može da bude samo klasa Poruka ili klase izvedene iz nje.