

Lokalni atletski maraton klub odlučio je koristiti računalni program koji sugerira prijavu atletičara na jednu od tri moguće discipline (maraton, polumaraton, utrka građana) na koje se natjecatelji prijavljuju i to temeljem broja pretrčanih kilometara tijekom jedne godine. Svaki je natjecatelj modeliran apstraktnom razredom `Natjecatelj` koja ima attribute: `duzina` (cijeli broj, označava broj pretrčanih kilometara, koja se kontinuirano povećava sa svakim treningom) i `ime` (`String`). Razred ima metodu: `void trening (int x)` kojom se trening karakterizira odgovarajućim brojem pretrčanih kilometara `x` koji, u metodi, uvećavaju ukupnu dužinu (atribut `duzina`). Atribut `duzina` mora biti dohvatljiv iz drugih razreda. Razred implementira sučelje:

```
interface Prijava
{
    String odabranaDisciplina (String ime);
}
```

Kreirati razred s opisanim karakteristikama, te dodati odgovarajući konstruktor s dva parametra i konstruktor kopije. Nadjačati metodu `toString()` koja ispisuje karakteristike natjecatelja (`ime` i dužinu pretrčanih kilometara).

Kreirati razred `Maratonac` koja nasljeđuje razred `Natjecatelj` i implementira metodu sučelja tako što poziva metodu apstraktnog razreda `toString()` i uz njen ispis dodaje „Maratonac“. Osim toga, razred `Maratonac` ima atribut koji je instanca razreda `ObjavljeneUtrke` (iz zadatka 3). Ta razred ima metode: `dodajMaratonca(Prijava p)`, `dodajPolumaratonca (Prijava p)`, `dodajGradanina(Prijava p)`. U konstruktoru razreda `Maratonac` treba omogućiti prijavu na utrku pozivanjem metode `dodajMaratonca(Prijava p)`, tako da se kao argument prenesu odgovarajući podaci o natjecatelju.

Kreirati razred `ObjavljeneUtrke` koja kao atribut ima tri liste:

```
public static ArrayList<Maratonac> p1 = new ArrayList<Maratonac>();
public static ArrayList<Polumaratonac> p2 =
    new ArrayList<Polumaratonac>();
public static ArrayList<Gradanin> p3 = new ArrayList<Gradanin>();
```

koje se popunjavaju pozivima metoda `dodajMaratonca(Prijava p)`, `dodajPolumaratonca(Prijava p)` i `dodajGradanina(Prijava p)` – implementirati jednu metodu po volji. Osim toga imaju, za tri karakteristične utrke, odgovarajuće statičke metode: `objaviUtrku1()`, `objaviUtrku2()` i `objaviUtrku3()` koje ispisuju odgovarajuća imena natjecatelja (pozivom metode `odabranaDisciplina(String)`) iz svih kategorija, ali prema svojim kriterijima temeljenima na atributu `duzina` svakog natjecatelja (zbog toga atribut `duzina` mora biti vidljiv). To znači, ako je natjecatelj prijavljen kao maratonac, a nema odgovarajući broj pretrčanih kilometara, za karakterističnu utrku njegovo ime neće biti objavljeno. Kriterije definirati proizvoljno i implementirati metodu `objaviUtrku1()`.

Kreirati testni razred u kojoj se kreira 100 natjecatelja koji pripadaju različitim kategorijama (maratonci, polumaratonci, građani - nasumično), svakom natjecatelju ime dodati kao string formiran slučajnim brojem (bilo kojim, definirati ga po volji) i broj pretrčanih kilometara kao nasumičan broj od 100 do 10000. Omogućiti u testnom programu objavljivanje imena natjecatelja za sve kategorije.