

# Vaje pri predmetu Programiranje 1

## Teden 13: Vsebovalniki in lambde

### Članki in avtorji

#### Naloga

Razreda `Clanek` in `Avtor` imata sledeče attribute:

```
public class Clanek {           // članek v reviji
    private List<Avtor> avtorji; // avtorji članka
    private String naslov;       // naslov članka
    private int leto;            // leto objave
}

public class Avtor {
    private String ime;
    private String priimek;
}
```

Dva ločena objekta tipa `Avtor` nista nikoli enaka, tudi če se ujemata v imenu in priimku.

Razred `Glavni` dopolnite z metodami, navedenimi v nadaljevanju. Po potrebi (dodatno) dopolnite tudi razreda `Avtor` in `Clanek`.

- `public static void urediPoLetuInNaslovu(List<Clanek> clanki)`

Podani seznam člankov uredi po *padajočih* letih objave, članke, objavljene v istem letu, pa po naslovih (v leksikografskem vrstnem redu). Uporabite metodo `sort` iz razreda `List`, ustrezeni objekt tipa `Comparator` pa izdelajte kot lambda.

- `public static Set<Avtor> vsiAvtorji(Collection<Clanek> clanki)`

Vrne množico vseh avtorjev, ki nastopajo vsaj pri enem članku v podani zbirki.

- `public static Map<Avtor, List<Clanek>> clankiPoAvtorjih(Collection<Clanek> clanki)`

Vrne slovar, ki vsakega avtorja, ki nastopa vsaj pri enem članku v podani zbirki, preslika v seznam vseh člankov, pri katerih ta avtor nastopa. Seznam naj bo urejen primarno po padajočih letih, sekundarno pa po naslovih (v leksikografskem vrstnem redu).

- `public static Avtor najplodnejšiAvtor(Collection<Clanek> clanki)`

Vrne avtorja, ki nastopa pri največjem številu člankov v podani zbirki. Lahko predpostavite, da je tak avtor vedno enolično določen. Pomagajte si z metodo `max` iz razreda `Collections` (ne `Collection`!), pripadajoči objekt tipa `Comparator` pa zopet izdelajte kot lambda.