

# Kontrolna vežba

1. Klasa **Grad** sadrži celobrojne  $x$  i  $y$  koordinate grada na dvodimenzionoj mapi, ime grada i broj stanovnika.
2. Klasa **Mapa** sadrži informacije o širini i visini mape. Koordinatni početak  $(0,0)$  se nalazi u gornjem levom uglu mape.
3. Mapa takođe sadrži i listu gradova. Potrebno je da je brzina pristupa elementima te liste u vremenu  $O(1)$ .
4. Pored liste gradova, Mapa sadrži i održava matricu rastojanja između svaka dva grada. Rastojanja su zadata kao realne vrednosti (float). Primiti da dimenziju matrice ne znamo unapred.
5. Klasa Mapa sadrži metodu za učitavanje liste gradova koja se zove **void ucitajGradove()**. Ta metoda učitava listu gradova i formira matricu rastojanja na osnovu datoteke koja se nalazi na lokaciji **"gradovi.txt"**. Format datoteke je takav da je najpre naveden broj gradova, a potom njihov spisak sa  $x$  i  $y$  koordinatama. Na osnovu ovoga treba formirati i matricu rastojanja. Npr:

4

Beograd 50 50

Valjevo 30 100

Zrenjanin 75 30

Kikinda 75 10

6. Postoji i druga metoda koja je abstraktna a koja ima potpis **abstract void ucitajPuteve(String putanja)** koja transformiše učitane matricu rastojanja tako da ostanu samo rastojanja između gradova između kojih postoje direktni putevi.
7. Klasa **ObicnaMapa** implementira metodu za učitavanje puteva tako što čita datoteku koja jednostavno sadrži matricu koja ima 1 na mestu gde direktan put postoji, a 0 na mestu gde ne postoji. Tako npr. za prethodno učitane matricu, datoteka sa putevima bi mogla da izgleda ovako:

0 1 1 0

1 0 0 0

1 0 0 1

0 0 1 0

8. Klasa **NaprednaMapa** učitava iz datoteke sa drugačijim formatom, naime tu su navedeni samo parovi gradova između kojih postoji direktan put. Pritom nisu navedeni indeksi gradova u matrici već samo njihovi nazivi. Potrebno je da pronalaženje njihovih indeksa u matrici bude u vremenu  $O(1)$ , iskoristiti pomoćnu strukturu koja ovo omogućava.

Beograd Valjevo

Beograd Zrenjanin

Zrenjanin Kikinda

9. Napisati metodu **void izracunajOstalePuteve()** koja nakon što su učitani direktni putevi, ažurira sadržaj ostalih rastojanja na osnovu indirektnih puteva. Npr. iako ne postoji direktan put između Kikinde i Beograda, postoji indirektan preko Zrenjanina, tako da će u matrici rastojanja i on da se pojavi.

10. Napišite metodu **void prikaziRastojanja()** koja ispisuje matricu rastojanja.

main funkcija bi mogla ovako da izgleda (za gore opisane datoteke):

```
Mapa mapa = new ObicnaMapa();
```

```
mapa.ucitajGradove();
```

```
mapa.ucitajPuteve("putevi1.txt");
```

```
mapa.prikaziRastojanja();
```

```
mapa.izracunajOstalePuteve();
```

```
mapa.prikaziRastojanja();
```

Prvi poziv za prikaz:

```
0 54 32 0
```

```
54 0 0 0
```

```
32 0 0 20
```

```
0 0 20 0
```

Drugi poziv za prikaz:

```
0 54 32 52
```

```
54 0 86 106
```

```
32 86 0 20
```