

**STRUKTURE PODATAKA I ALGORITMI – LAB 2**

Prije rješavanja zadataka preuzmite na svoje računalo sljedeće datoteke:

- cities.csv
- SPA-Lab02.py

Objekte datoteke neka budu u istom folderu te svoja rješenja zadataka pišite u datoteci SPA-Lab02.py.

### Zadatak 1:

Definirana je klasa **Gradovi** te će se unutar konstruktora klase učitati nazivi gradova i broj stanovnika za  $n$  gradova ( $n$  je parametar konstruktora klase). Za svaki grad učitava se naziv grada i broj stanovnika što se pohranjuje u listu **podaci** (svojstvo klase) pri čemu je jedan element liste par vrijednosti (`[naziv grada], [broj stanovnika]`). Napišite metodu **prvi (self)** koja će vraćati naziv prvog grada te broj stanovnika (kao string) za prvi grad iz svojstva **podaci**. Nadalje kreirajte instancu klase za 100 gradova i ispišite naziv prvog grada i broj njegovih stanovnika.

*Primjer:*

Ulaz	Ispis
	dengzhou 85279

### Zadatak 2:

Napišite metodu **najveci (self)** koja će vraćati naziv grada te pripadni broj stanovnika za grad koji ima najviše stanovnika.

### Zadatak 3:

Napišite metodu **poredani (self)** koja će vraćati *True* ako su gradovi poredani prema nazivu grada (uzlazno ili silazno), a *False* inače.

### Zadatak 4:

Definirajte metodu **trazi (self, naziv\_grada)** koja će vraćati *True* ako u listi učitanih gradova postoji grad čiji je naziv `[naziv_grada]`, inače će vraćati *False*. Traženje grada implementirajte slijednim (sekvencijalnim) traženjem.

**STRUKTURE PODATAKA I ALGORITMI – LAB 2****Zadatak 5:**

Definirajte metodu `bubble_sort(self)` koja će sve gradove iz svojstva `podaci` poredati prema nazivu grada rastuće. Metoda vraća sortiranu listu s gradovima pri čemu vrijednost svojstva `podaci` treba ostati nepromijenjena. Sortiranje treba realizirati bubble sortom.

**Zadatak 6:**

Definirajte metodu `binary_search(self, naziv_grada)` koja će prvo poredati gradove, a onda vraćati *True* ako u listi gradova postoji grad čiji je naziv `[naziv_grada]`, inače će vraćati *False*. Traženje grada implementirajte binarnim traženjem.

**Zadatak 7:**

Definirajte metodu `merge_sort(self)` koja će sve gradove iz svojstva `podaci` poredati prema nazivu grada rastuće. Metoda vraća sortiranu listu s gradovima pri čemu vrijednost svojstva `podaci` treba ostati nepromijenjena. Sortiranje treba realizirati merge sortom.

**Zadatak 8\*:**

Definirajte funkciju `vizualiziraj()` koja će grafički prikazati usporedbe trajanje bubble i merge sorta na listama gradova veličine 1000, 2000, 5000, 8000, 10000 gradova. Podatke treba prikazati grafički.