

## PROGRAMSKI JEZICI 1 (28.09.2015.)

- ❶ (20 bodova) Definirati klasu **CelestialBody** sa privatnim članovima masa, poluprečnik i naziv redom tipova **double**, **double** i **char\***. Unutar klase **CelestialBody** deklarirati virtuelnu funkciju **print(ostream&, int)**, te preklopiti operator **<<** za ispis na standardni izlaz korištenjem funkcije **print(ostream&, int)**, pri čemu drugi parametar te funkcije određuje koliko se tab znakova ispisuje na početku linije, a funkcija ispisuje naziv, masu i poluprečnik tijela.

Klasa **CelestialBody** treba da preklopi operator **<**, kao funkciju članicu koji poredi dva nebeska tijela prema njihovoj masi.

- ❷ (45 bodova) Definirati generičku klasu **List<T>** koja predstavlja uvezanu listu elemenata tipa **T**. Kako se radi o uvezanoj listi, broj elemenata nije ograničen. Klasa treba da sadrži privatni slog **Element** koji za podatke članove treba da ima pokazivač na podatak tipa **T**, pokazivač na sljedeći element (**Element\***) i konstruktor koji prihvata referencu na podatak. Neophodno je implementirati i odgovarajući destruktore. Pored toga, klasa treba da ima:

- privatni podatak član tipa **Element\*** koji predstavlja pokazivač na početak uvezane liste.
- operator **+=(T& t)** koja u uvezanu listu dodaje novi element sa podatkom **t**.
- operator **[]** koji vraća referencu na podatak na datom indeksu u listi. Ukoliko je traženi indeks izvan opsega liste podići izuzetak sa odgovarajućom porukom (dovoljno je podići izuzetak u obliku stringa „Index out of range.“). Izuzetak uhvatiti u glavnoj funkciji (pri pozivu).
- operator **()** koja vraća broj elemenata u listi.
- metodu **sort()** koja sortira elemente pozivanjem operatora poređenja (operatora **<**) nad odgovarajućim podacima (funkcija treba da koristi operator **()** za određivanje broja elemenata pri sortiranju).

- ❸ (20 bodova) Naslijediti klasu **LargeBody** iz klase **CelestialBody**. Klasa **LargeBody** treba da, za podatak član, ima listu satelita koji su tipa **CelestialBody**, te da preklopi funkciju **print(ostream&, int)** iz natklase tako da ispisuje svaki od svojih satelita uvučen za jedan tab znak u odnosu na početak ispisa date klase. S obzirom da se radi o virtuelnoj metodi, pozvati i funkciju **print(ostream&, int)** iz natklase prije poziva ispisa nad satelitima, tako da se ispišu i podaci o datom tijelu. Pored navedenog, klasa **LargeBody** treba da ima:

- funkciju **addSatellite(CelestialBody&)** koja dodaje satelit u listu satelita.
- funkciju **sortSatellite()** koja sortira satelite prema masi.
- operator **[]** koji vraća **CelestialBody** na datom indeksu (ukoliko klasa **List<CelestialBody>** podigne izuzetak pri pozivu istog operatora, izuzetak ne treba obraditi unutar koda operatora klase **LargeBody**, već u glavnoj funkciji).

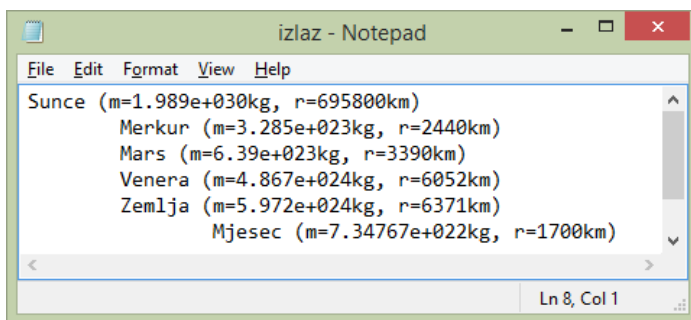
- ❹ (15 bodova) U glavnom programu treba napraviti dva velika tijela: Sunce i Zemlja; te nebeska tijela: Merkur, Venera, Mars, Mjesec, sa podacima koji se mogu zaključiti iz slike. Potom:

- U satelite Sunca dodati redom Merkur, Veneru, Zemlju i Mars.
- U satelite Zemlje dodati Mjesec.
- Sortirati satelite Sunca prema njihovoj masi.
- U datoteku *izlaz.txt* upisati Sunce.
- Pokušati u isti fajl upisivanje redom trećeg, jedanaestog i četvrtog satelita Sunca (svi pozivi treba da budu unutar istog try bloka i navedenim redom). U slučaju izuzetka uhvatiti poruku i ispisati je na standardni izlaz.
- Poslije izvršavanja programa, šta se nalazi na prikazu standardnog izlaza, a šta u dijelu datoteke koji se na slici ne vidi?

### Napomena:

Razdvojiti interfejs i implementaciju.

Primjer izlazne datoteke:



```
File Edit Format View Help
Sunce (m=1.989e+030kg, r=695800km)
Merkur (m=3.285e+023kg, r=2440km)
Mars (m=6.39e+023kg, r=3390km)
Venera (m=4.867e+024kg, r=6052km)
Zemlja (m=5.972e+024kg, r=6371km)
Mjesec (m=7.34767e+022kg, r=1700km)
```