

Uvod u informacione tehnologije – vežba4

1. Napravite novi projekat Vežba4. U okviru *src* foldera projekta napravite paket *vozila*. U okviru paketa *vozila* napravite apstraktnu klasu *Vozilo*:

```
publicabstractclass Vozilo {
    boolean radi;
    short ks;
    String regBr;

    publicvoid upali() {
        radi = true;
    }

    publicvoid ugasi() {
        radi = false;
    }

    publicabstractfloat platiPutarinu(int km);

    publicabstractvoid vozi();

    publicint compareTo(Vozilo g) {
        if(ks < g.ks) {
            return -1;
        } elseif(ks == g.ks) {
            return 0;
        } else {
            return 1;
        }
    }
}
```

2. Napišite konstruktor klase *Vozilo* koji prima parametre *radi*, *konjskeSnage* i *registarskiBroj*. Napomena: klasa *Vozilo* je apstraktna. Ovako definisani konstruktor ne može da se poziva direktno, ali može da se poziva u klasama naslednicama.
3. U klasu *Vozilo* dodajte *toString()* metodu.
4. Napraviti klasu *Tocak* koja od atributa ima *oznaku* i *pritisak*. U okviru klase napraviti dva konstruktora – jedan koji prima *oznaku* i *pritisak* kao parametre i default konstruktor (koji ne prima parametre već oznaku i pritisak postavlja na default vrednosti).
5. Napraviti *toString()* metodu u klasi *Tocak*.
6. U paketu *vozila* napraviti klasu *Automobil*:

```
publicclass Automobil extends Vozilo {
    bytebrojSedista;
    Tocak prednjiLevi;
    Tocak prednjiDesni;
    Tocak zadnjiLevi;
    Tocak zadnjiDesni;
}
```

```

public Automobil(boolean radi, short ks, String regBr, byte brojSedista,
String oznaka, float pritisak) {
    super(radi, ks, regBr);
    this.brojSedista = brojSedista;
    prednjiLevi = new Tocak(oznaka, pritisak);
    prednjiDesni = new Tocak(oznaka, pritisak);
    zadnjiLevi = new Tocak(oznaka, pritisak);
    zadnjiDesni = new Tocak(oznaka, pritisak);
}

```

7. U klasu *Automobil* dodati default konstruktor (koji ne prima parametre) i koji postavlja attribute klase na default vrednosti: *radi* na *false*, *ks* na 0, *regBr* na prazan string, *brojSedista* na 4, oznaku točka na prazan string i pritisak točka na 0.
8. Implementirati metode *platiPutarinu* i *vozi* u klasi *Automobil*. Putarina se za automobil izračunava tako što se pređeni broj km pomnoži sa 2.5. Metoda *vozi* bi trebala ispisati "Vozim automobil" ukoliko je vrednost atributa *radi* *true*, a "Automobil nije upaljen" ukoliko je vrednost atributa *radi* *false*.
9. Dodati *toString()* metodu u *Automobil*. U okviru ove metode iskoristiti *toString()* metodu klase *Vozilo*.
10. Napraviti klasu *Kamion* koja takođe nasleđuje klasu *Vozilo*. *Kamion* od atributa ima nosivost (float) i niz od 6 točkova. Metoda *vozi* bi u *Kamionu* trebala ispisati da se vozi kamion ili da kamion nije upaljen (u zavisnosti od vrednosti atributa *radi*). U slučaju kamiona, putarina se računa tako što se pomnože pređeni km, nosivost i ta vrednost se pomnoži još sa 2.5.
11. Dopuniti klasu *Kamion* konstruktorima (bez parametara i sa prosleđenim parametrima) i metodom *toString()*.
12. U klasu *Kamion* dodati i konstruktor koji kao parameter prima drugi objekat tipa *Kamion*. Konstruktor bi trebao da podesi vrednosti atributa na vrednosti odgovarajućih atributa prosleđenog objekta.
13. U metodu *Kamion* dodati i metodu *getNosivost()* koja vraća vrednost atributa nosivost.
14. Za testiranje napraviti klasu *NaplatnaRampa*.
 - a. Napraviti niz od 4 vozila (dva automobila i dva kamiona).
 - b. Ispisati sva vozila
 - c. Svim vozilima naplatiti putarinu na deonici od 80 km. Prikazati ukupan iznos naplaćene putarine.
 - d. Koristeći metodu *compareTo()* klase *Vozilo* uporedite jedan automobil i kamion
 - e. Ispisati nosivost jednog od kamiona iz noza vozila (napomena: koristiti "type cast").
15. U klasu *Vozilo* dodati konstruktor koji prima parametre *radi* i *procenatKvara*. Ukoliko je *procenatKvara* > 55, *radi* dobija vrednost *false*, u suprotnom *radi* postaviti na prosledjenu vrednost. Svi ostali atributi se unutar ovog konstruktora postavljaju na podrazumevane vrednosti.