

ZADACI ZA VJEŽBU – OOP

1. Implementirati sistem za praćenje trka.
 - a. Kreirati klasu Vrijeme, koja ima privatne parametre sati, minuti i sekunde, koji su tipa int.
 - b. Generisati sve getere i setere.
 - c. U klasi Vrijeme, prazan konstruktor koji sve vrijednosti postavlja na 0.
 - d. U klasi Vrijeme, kreirati konstruktor sa sva tri argumenta.
 - e. U klasi Vrijeme, kreirati funkciju prikaziVrijeme() koja prikazuje vrijeme u sljedećem formatu: hh:mm:ss. Dakle, ako su vrijednosti za sate, minute i sekunde 1, 2, 51, respektivno, funkcija prikazuje: 01:02:51.
 - f. Kreirati klasu Vozac, koja ima privatne podatke članove: ime (String), prezime (String), kao i objekat tipa Vrijeme, koji predstavlja njegovo vrijeme za određenu trku.
 - g. Generisati sve getere i setere.
 - h. U klasi Vozac, kreirati konstruktor sa svim argumentima.
 - i. U klasi Vozac, kreirati funkciju prikaziRezultat() koja štampa informacije o vozaču i njegovom vremenu.
 - j. U klasi Vozac, kreirati funkciju kazni(int sekunde), koja treba da poveća vrijeme trke Vozaca za određeni broj sekundi, ukoliko je napravio neku grešku.
 - k. Kreirati klasu Trka, koja ima jedan privatni argument, a to je niz objekata tipa Vozac.
 - l. U klasi Trka, kreirati funkciju koja će iz niza vozača izdvojiti onog koji ima najbolje vrijeme.
 - m. Kreirati Test klasu gdje ćete testirati sve navedene funkcije.

2. Implementirati jednostavni sistem za prodavnicu.
 - a. Kreirati klasu *Proizvod*, koja ima tri privatna podatka člana: naziv (*String*), količina (*double*), kao i cijena (*double*).
 - b. U klasi *Proizvod*, kreirati sve getere i setere.
 - c. U klasi *Proizvod*, kreirati konstruktor sa svim argumentima.
 - d. Kreirati klasu *Korpa*, koja ima jedan privatni podatak član, a to je niz objekata tipa *Proizvod*.
 - e. U klasi *Korpa*, kreirati prazni konstruktor koji inicijalizuje niz proizvodi na niz od 10 null elemenata.
 - f. U klasi *Korpa*, kreirati funkciju *dodajProizvod (Proizvod p)* koja dodaje proizvod na prvo slobodno mjesto u niz proizvoda.
 - g. U klasi *Korpa*, kreirati funkciju *cijena()* koja štampa ukupnu cijenu korpe.
3. Firma želi da implementira softver za prepoznavanje objekata na slikama. Vaš dio zadatka je da:
 - a. Implementirate klasu *Bbox*, koja sadrži privatne podatke tipa *int xmin, ymin, xmax, ymax*.
 - b. Napisati gere i setere za sve podatke
 - c. Napisati prazan konstruktor koji postavlja sve vrijednosti na 0
 - d. Napisati konstruktor koji ima 4 argumenta.
 - e. Napisati metode *int width()* i *int height()*, koji vraćaju visinu i širinu boxa.
 - f. Napisati metod *int area(Bbox arg)*, koji vraća površinu presjeka boxa i boxa koji je predat kao argument.
 - g. Napisati metod *int union(Bbox arg)*, koji vraća površinu unije boxa i boxa koji je predat kao argument.
 - h. Napisati metod *double iou(Bbox arg)*, koji vraća količnik presjeka i unije boxa i boxa koji je predat kao argument.
 - i. Kreirati klasu *Image*. Klasa *Image* treba da sadrži sljedeće privatne podatke *int[][] image* i *Bbox[] boxes*, *int numBoxes*, *int capacity*, *int width*, *int height*. Podatak *image* je matrica koja opisuje sliku. Niz *boxes* sadrži boxove koji ograničavaju objekte na slici koje treba da prepoznamo, *numBoxes* predstavlja broj boxova u nizu *boxes*, a podatak *capacity* predstavlja kapacitet niza *boxes*. Podaci *width* i *height* predstavljaju dimenzije matrice koja opisuje sliku.
 - j. Napisati konstruktor koji ima tri argumenta koji sadrži vrijednost za *kapacitet*, *visinu* i *širinu slike*. Na osnovu zadatih vrijednosti alocirati memoriju za matricu *image* i niz.
 - k. Napisati konstruktor koji koji uzima *image*, *boxes*, *capacity*, *width* i *height* i postavlja odgovarajuće vrijednosti. Neophodno je napraviti kopiju niza *boxes* i matrice *image*.

- l. Napisati metodu *void add(Bbox arg)*. Ova metoda dodaje element tipa *Bbox* u niz *boxes*.
- m. Napisati metodu *Bbox get(Bbox arg)*. Ova metoda vraća box iz niza *boxes* koji ima najveći *iou* sa boxom koji je predat kao argument. Ukoliko postoji više takvih boxova vratiti onaj koji se nalazi prvi u nizu.
- n. Napisati metodu *void clean()* uklanja duplikate iz niza *boxes*. (5 poena)