

KOLOKVIJUM

C++/Java

Napisati program koji simulira parking za autobuse.

C++

1) Napisati apstraktnu klasu **Vozilo** koje ima sledeća polja:

- **regbr** koje označava registarski broj vozila tipa **DinString**
- **vrsta** nabrojivog tipa **VrstaVozila** (ima vrednosti LAKO, MOTORNO, TERETNO, PRIKLJUCNO)
- **godiste** vozila tipa **int**

Za klasu **Vozilo** implementirati sledeće:

- konstruktor bez parametara (postavite polja na vrednosti po želji)
- konstruktor sa parametrima (za sva polja)
Vozilo(const DinString&, VrstaVozila, int);
- getere za polja: **regbr** i **godiste**
- apstraktnu metodu **bool voznoStanje()**

Iz klase **Vozilo** izvesti klasu **Autobus**. Klasa **Autobus** ima dodatno polje:

- **brojMesta** tipa **int** koja definiše broj sedišta koje autobus ima
- statičko polje **brojAutobusa** tipa **int** koje predstavlja broj objekata klase **Autobus**

Za klasu **Autobus** implementirati sledeće:

- konstruktor bez parametara (postavite polja na vrednosti po želji)
- konstruktor sa parametrima (za sve parametre)
Autobus(const DinString&, VrstaVozila, int, int);
- napisati realizaciju metode **bool voznoStanje()** tako da metoda vraća **true** ukoliko je godište veće od 2000. U suprotnom metoda vraća **false**.
- implementirati i operator ispisa koji ispisuje sve podatke o autobusu (uključujući i polja iz vozila). Polje **vrsta** nabrojivog tipa treba ispisati rečima.

2) Napisati klasu **Parking** koja ima sledeća polja:

- naziv parkinga, tipa **DinString**
- lista parkiranih autobusa, tipa **List<Autobus*>**

Za klasu **Parking** implementirati sledeće:

- Konstruktor sa parametrima **Parking(const DinString&)**
- metodu **bool uparkiraj(Autobus&)** koja dodaje autobus na parking na kraj liste ako je autobus koji se parkira u voznom stanju.
- metoda **void ispisiNove()** koja ispisuje podatke o parkingu: naziv i sve autobuse na parkingu čija je godina proizvodnje (polje **godiste**) jednaka godištu najmlađeg autobusa na parkingu.

Java:

3) Napisati klasu **Autobus** koja ima polja:

- **regbr** koje označava registarski broj vozila tipa **String**
- **godiste** vozila tipa **int**

Za klasu **Autobus** implementirati sledeće:

- konstruktor sa parametrima (za sve parametre)

Autobus(String, int);

- metodu **boolean voznoStanje()** koja vraća **true** ukoliko je godište veće od 2000. U suprotnom metoda vraća **false**.
- getere za polja: regbr i godiste

Napisati klasu **Parking** koja ima sledeca polja:

- naziv parkinga, tipa **String**
- heš mapu parkiranih autobusa, tipa **HashMap<String, Autobus>** gde je ključ vrednost polja registarske tablice autobusa

Za klasu **Parking** implementirati sledeće:

- Konstruktor sa parametrima **Parking(String)**
- metodu **boolean uparkiraj(Autobus)** koja dodaje autobus u mapu ukoliko je autobus u voznom stanju i ukoliko u mapi nema autobusa sa tim registarskim brojem.
- metodu **Autobus isparkiraj(String)** koja izbacuje autobus iz liste sa prosleđenim brojem registarske tablice, ukoliko takav autobus postoji. Ako je uspešno izvršena operacija vratiti izbačeni autobus, ukoliko nije vratiti null.
- metodu **String toString()** koja ispisuje naziv parkinga i registarske tablice svih autobusa na parkingu.

Napomene:

- U C++ za stringove koristiti klasu **DinString**, a za listu klasu **List** koje su date.
- Sva polja možete implementirati kao private ili protected (ne public) u zavisnosti od vašeg izbora načina implementacije konstruktora i metoda u zadatku.
- Program se **mora uspešno kompajlirati kad u main.cpp uključite (include) sve hpp fajlove za klase iz zadatka**, u suprotnom možete dobiti najviše 1 poen.