

Kolokvijum - Priprema

C++

Napraviti program koji evidentira cenu putovanja na osnovu kupljenih karata.

1) Napisati klasu **Karta** koja ima sledeća polja:

- identifikator karte, tipa **int**
- mesto polaska, tipa **DinString**
- mesto dolaska, tipa **DinString**
- cena karte, tipa **float**
- statičko polje **ukupnoKarata**, tipa **int** (reprezentuje ukupan broj instanci klase **Karta**)

Za klasu **Karta** implementirati i sledeće:

- Konstruktor bez parametara - postaviti vrednosti nestatičkih polja na proizvoljne vrednosti
- Konstruktor sa parametrima - **Karta(int, const DinString&, const DinString&, float)**
- Konstruktor kopije
- getere za sva polja

2) Implementirati virtuelnu metodu **bool jeftinaKarta(float)** u klasi **Karta** (realizovati je tako da joj je povratna vrednost uvek **true**).

Iz klase **Karta** izvesti klasu **AvionskaKarta** i za nju implementirati sledeće:

- Konstruktor sa parametrima - **AvionskaKarta(int, const DinString&, const DinString&, float)**
- redefinisati virtuelnu metodu **bool jeftinaKarta(float)** tako da metoda vraća **true** ukoliko je cena manja od prosleđenog parametra, u suprotnom vraća **false**
- metodu **void ispisi()** koja ispisuje sve podatke za avionsku kartu

3) Napisati klasu **Putovanje** koja ima sledeća polja:

- naziv putovanja, tipa **DinString**
- maksimalna cena putovanja, tipa **int**
- lista karti koje su kupljene za putovanje, tipa **List<AvionskaKarta>**

Za klasu **Putovanje** implementirati i sledeće:

- Konstruktor bez parametara - naziv putovanja postaviti na proizvoljnu vrednost, cenu na 0, a listu karti inicijalizovati kao praznu listu
- Konstruktor sa parametrima **Putovanje(const DinString&, int)** - postavlja naziv putovanja i maksimalnu cenu na prosleđene parametre, a listu karti inicijalizovati kao praznu listu
- getere za polja naziv i maksimalna cena
- metodu **bool dodajKartu(const AvionskaKarta&)** koja dodaje kartu na kraj liste, ukoliko je mesto dolaska poslednje karte isto kao mesto polaska karte koja se dodaje (ovaj uslov ne važi ukoliko je lista karata prazna), takođe ne sme se dodati karta ukoliko bi se njenim dodavanjem prekoračila maksimalna cena putovanja (cena putovanja se prekorači u slučaju da je suma cena svih karata u putovanju veća od maksimalne cene putovanja), kao ni karta čiji identifikator već postoji u listi karata
- metoda **bool izbaciKartu(int)** koja izbacuje kartu sa prosleđenim identifikatorom, ukoliko takva karta postoji
- metoda **void ispisi()** koja ispisuje maksimalnu cenu putovanja, naziv putovanja i broj karata u putovanju, kao i podatke o svim kreiranim kartama

Napomene:

- 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**, u suprotnom možete dobiti najviše 36 poena