

OP K2

C++

Modelirati prevoz žičarom na nekoj planini. Žičara radi bez prekida i prevozi po nekoliko putnika odjednom u pravilnim razmacima od po nekoliko minuta. Koliko putnika se prevozi žičarom u jednoj turi zavisi od tipa žičare (dvosjed, četvorosjed, šestosjed,...). Žičara se koristi u različitim periodima godine i svaki period može imati drugačiju vrstu karte.

Ovo omogućiti implementacijom sledećih tipova podataka.

Nabrojivi tip podataka **StanjeZicare** ima vrednosti: **OTVORENA**, **ZATVORENA**.

Nabrojivi tip podataka **VrstaKarte** ima sledeće vrednosti: **LETO**, **SKI**, **SKI_OPENING**.

Klasa **Karta** ima:

- **int brojKarte**
- **VrstaKarte vrsta**
- konstruktor bez parametara
- konstruktor sa parametrima
- metod **getVrsta**
- metod **getBroj**
- operator ispisa <<

Klasa **Putnik** ima:

- **Karta karta**
- **DinString mb**
- **DinString ime**
- konstruktor bez parametara
- konstruktor sa parametrima - svi neophodni parametri za izgradnju objekta
- metod **getMb**
- metod **getVrstaKarte**
- virtualni metod **toString const** - kreira i vraća novi string spajanjem informacija o objektu (mb i ime); dodati neki karakter za radvajanje (npr: -, |, space,...)
- operator ispisa << - ispisuje rezultat metode **toString** i ispisuje Kartu

Klasa **PutnikDete** nasleđuje klasu **Putnik** i ima:

- dodatno polje **int mbRoditelja**
- konstruktor bez parametara
- konstruktor sa parametrima za sva polja (**Putnik** i **PutnikDete**)

- konstruktor sa parametrima gde je prvi parametar objekat klase Putnik koji predstavlja roditelja
- redefinisani metod **toString** - dodaje tekst "Dete" i ispisuje sve informacije o objektu (Putnik i PutnikDete)

Generička klasa **Zicara<VrstaKarte SEZONA, int BROJ_SEDISTA, int INTERVAL>** je određena sezonom (vrstom karte), brojem sedišta koji se odnosi na vrstu žičare (koliko putnika se odjednom može prevesti) i intervalom izraženim u minutima koji se odnosi na vremenski period između puštanja dve ture putnika.

Klasa ima:

- **List<Putnik*> putnici**
- **StanjeZicare stanje**
- **int ukupnoVoznji** - broji ukupan broj voznji od otvaranja (pogledaj metod **preveziTuru**)
- **int ukupnoMinuta** – broji koliko dugo zicara radi u minutima od otvaranja (pogledaj metod **preveziTuru**)
- konstruktor bez parametara - postavlja stanje na **ZATVORENA**
- metod **otvori** - postavlja stanje na **OTVORENA** i ukupan broj voznji i minuta na 0
- metod **zatvori** - koji postavlja stanje na **ZATVORENA** (ukupan broj voznji i minuta se ne menja) i prazni listu putnika koji čekaju u redu za prevoz .
- metod **dodajPutnika** - dodaje zadatog putnika na kraj reda ukoliko taj putnik vec nije u listi. Proveravati po maticnom broju.
- metod **brojPutnika** - vraća koliko putnika čeka u redu
- metod **preveziTuru** - prevozi prve putnike na redu i to samo ukoliko je žičara u stanju **OTVORENA**. Putnik se ne prevozi i uklanja se iz reda ukoliko njegova karta nije važeća (vrsta karte se ne poklapa sa sezonom). Ovom prilikom se prevozi onoliko putnika odjednom koliko žičara ima sedišta. Prilikom prevoženja putnici se uklanjaju iz reda čekanja. Ukoliko u redu ima manje putnika nego što je potrebno da se popuni jedna tura, prevozi se manje putnika. Polje **ukupnoVoznji** se uvećava za broj prevezenih putnika. Takođe, ukoliko je bilo prevoza, polje **ukupnoMinuta** se uvećava za zadati interval.
- operator ispisa **<<** - ispisuje sezonu, broj sedišta na žičari, trajanje jedne vožnje, stanje, ukupan broj vožnji, ukupno radno vreme žicare u satima, broj putnika u redu i listu svih putnika koji čekaju u redu.

NAPOMENE

- Za listu putnika koristiti klasu List koje je data i koju treba uključiti u projekat.
- Za nabrojive tipove podataka se, takođe, može realizovati operator ispisa, ali nije obavezan.

- Pretpostaviti da je unos validan odnosno zanemariti slučajeve kada je interval ili broj sedišta negativna vrednost i slično.
- DinString klasa ima preklopljen operator + što znači da se stringovi mogu sabirati
- DinString klasa ima preklopljen operator == što znači da se stringovi mogu upoređivati
- Izbegavati ponavljanje koda
- Sve što nije testirano u `main()` funkciji (implicitno ili eksplicitno) ne donosi bodove.
- Program se mora uspešno kompajlirati, u suprotnom možete dobiti najviše jedan bod.

