

Praktikum iz objektno-orientisanog programiranja (13S112POOP)

Prva laboratorijska vežba – C++

Cilj laboratorijske vežbe jeste upoznavanje studenata sa korišćenjem standardne biblioteke jezika C++ (iteratori, kolekcije, algoritmi), lambda funkcijama i regularnim izrazima.

Zadatak 1.

U fajlu *Zadatak1.cpp* priložena je definicija klase Tačka i Izlomljena. Kreirati klasu iteratora sa svim potrebnim metodama i dodati u klasu Izlomljena potrebne metode kako bi se omogućilo iteriranje kroz tačke koje pripadaju izlomljenoj liniji u obrnutom redosledu od redosleda njihovog zadavanja (*reverse iterator*). Testirati realizovani iterator koristeći naprednu verziju for ciklusa.

Zadatak 2.

Izmeniti implementaciju klase Izlomljena da umesto dinamičkog niza koristi *vector* iz standardne biblioteke jezika C++. Obezbediti sledeće mogućnosti sa klasom Izlomljena:

- rotacija izlomljene linije oko Y-ose,
- prebrojavanje tačaka koje ispunjavaju zadati uslov,
- sortiranje tačaka (uz mogućnost zadavanja načina za poređenje dve tačke),
- pronalaženje dve najudaljenije tačke (vraćaju se pomoću *pair*),
- uklanjanje zadatke tačke (ili svih tačaka koje ispunjavaju zadati uslov).

Koristiti algoritme standardne biblioteke i lamda funkcije gde god je to moguće. U postupku testiranja realizovanih metoda isprobati:

- sortiranje tačaka po:
 - X-osi,
 - Y-osi,
 - rastojanju od koordinatnog početka,
 - ...
- prebrojavanje tačaka:
 - koje se nalaze na pozitivnoj strani X-ose,
 - koje se nalaze na negativnoj strani Y-ose,
 - kojima je udaljenost po X-osi od koordinatnog početka veća od udaljenosti po Y-osi
 - ...
- uklanjanje najudaljenijih tačaka.

Zadatak 3.

Priložena je datoteka *linija_74.txt* u kojoj se nalazi spisak svih stajališta kroz koja prolazi ova gradska linija. U svakom redu dati su podaci za po jedno stajalište i to: šifra stajališta, naziv stajališta, spisak drugih linija koje prolaze kroz to stajalište, geografska lokacija (širina i dužina), kao i redni broj stajališta.

1. Definisati klasu *Stajalište* koja sadrži sve podatke o jednom stajalištu koji su navedeni u priloženoj datoteci, a zatim klasu *Linija* koja predstavlja niz Stajališta i ima svoju oznaku koja je tekstualnog tipa. Koristiti kolekcije standardne biblioteke.
2. Na osnovu datog formata kreirati regularni izraz za parsiranje priložene datoteke i instancirati objekat *lin_74* klase *Linija* sa svim priloženim podacima.
3. Implementirane klase opremiti metodama za ispis po formatu prikazanom na slici 1. U svakom redu ispisuje se po jedno stajalište i to šifra stajališta, naziv, geografski položaj i skup linija koje prolaze kroz dano stajalište.

1985	Narodnih heroja	44.8225, 20.4069	{ 612 70 74 }
1993	Crkva Svetog Dimitrija	44.8204, 20.4117	{ 17 70 74 }
435	Bulevar umetnosti	44.8186, 20.4155	{ 17 68 70 74 }
821	Beogradska arena	44.8145, 20.4192	{ 70 74 }
387	Beogradska arena /antifasisticke borbe/	44.8111, 20.4226	{ 18 601 70 74 88 }

Slika 1. Format ispisa linije sa stajalištima

4. Implementirati metodu u klasi *Linija* koja:
 - a. vraća skup svih linija sa kojima data Linija ima zajednička stajališta,
 - b. vraća vektor od 4 stajališta u kojim je redom smešteno najsevernije, najistočnije, najjužnije i najzapadnije stajalište,
 - c. određuje liniju sa kojom data linija ima najviše zajedničkih stajališta. Povratna vrednost metode je tipa *pair* i sadrži Liniju i ceo broj koji označava koliko je to zajedničkih stajališta,
 - d. određuje parove linija koje imaju zajednička stajališta uzimajući u obzir samo stajališta kroz koja prolazi data linija. Povratna vrednost metode je mapa koja mapira par linija na broj zajedničkih stajališta. Kao argument metode moguće je proslediti i celobrojnu vrednost kojom se vrši filtriranje parova linije na samo one parove koji imaju više zajedničkih stajališta od zadate celobrojne vrednosti.

NAPOMENA: cilje realizacije metoda iz tačke 4. jeste intenzivno korišćenje kolekcija i algoritama standardne biblioteke jezika C++, kao i lambda funkcija.