

2 Više klasa u jednom programu

Napomena za sve zadatke:

- Možete dodavati attribute koji će vam omogućiti da efikasnije uradite program, ali ne smeju da budu javni
- Možete dodati sve potrebne javne metode koje će obezbediti da program radi (uključujući konstruktore i destruktor);
- Ne smete unositi podatke u telu konstruktora (scanf, cin, ...)
- Izbegnite na svaki način dupliranje koda
- **Unos test podataka mora da bude iz fajla, a ne sa standardnog ulaza, osim ako u zadatku nije bas naglašeno.** Štampanje na standardni izlaz je u redu
- Ukoliko se traži testiranje ponašanja nizova, nizovi treba da imaju makar po 5 elemenata
- Ukoliko se traži kreiranje nekoliko objekata, to znaci najmanje 4
- Objekte koje kreirate u glavnom programu, kreirati u dinamičkoj zoni memorije, osim ako nije suprotno naglašeno
- Broj zadatka koji radite se određuje na isti način kao u prvoj vežbi

Zadatak 0.

Kreirati klasu Complex za modelovanje kompleksnih brojeva. Od privatnih članova treba da ima po jednu double promenljivu za realni i imaginarni deo, a od javnih operatore za unos i štampanje, funkcije za sabiranje i oduzimanje koje vraćaju novi objekat, kao i sve neophodne konstruktore i funkcije za pristup privatnim članovima.

Klasu Skup, kreiranu u prvoj vežbi, modifikovati tako da umesto niza elemenata tipa int sadrži niz elemenata tipa Complex, i dodati:

- funkciju koja određuje razliku 2 skupa (rezultat je ovi objekat tipa Skup),
- funkciju koja određuje uniju 2 skupa (rezultat je ovi objekat tipa Skup),
- funkciju koja određuje presek 2 skupa (rezultat je ovi objekat tipa Skup) i
- operatorsku funkciju = koja preslikava jedan skup u drugi.

Funkciju za učitavanje elemenata skupa sa standardnog ulaza zameniti prijateljskom operatorskom funkcijom >> za učitavanje elemenata iz tekstualnog toka podataka, a funkciju za prikaz elemenata skupa na standardni izlaz, prijateljskom operatorskom funkcijom << za upis elemenata skupa u tekstualni tok podataka.

U funkciji main kreirati nekoliko objekata klase Skup u dinamičkoj zoni memorije, učitati njihove elemente i na standardni izlaz prikazati njihovu zajedničku uniju, presek i razliku.

Zadatak 1.

Kreirati klasu Ispit koja od privatnih članova treba da ima jednu promenljivu u kojoj se pamti ime ispita (tip char*), jednu za čuvanje datuma polaganja ispita u formatu yyyy-MM-dd (npr. 2017-11-06) i jednu u kojoj se pamti ocena (tip int). Od javnih članova klasa treba da sadrži funkcije koje treba da porede dva objekta klase Ispit po oceni ili datumu polaganja, funkcije za unos i štampanje, kao i sve neophodne konstruktore i funkcije za pristup privatnim članovima.

U klasi Student, kreiranoj u prvoj vežbi, zameniti vektor ocena vektorom objekata klase Ispit i dodati:

- konstruktor bez argumenata koji broj ispita postavlja na 50,
- funkciju koja poredi dva studenta prema prosečnoj oceni,
- operatorsku funkciju = koja jedan objekat klase Student dodeljuje drugom.

Funkciju za učitavanje podataka o studentu sa standardnog ulaza zameniti prijateljskom operatorskom funkcijom >> za učitavanje podataka o studentu iz tekstualnog toka podataka, a funkciju za prikaz podataka o studentu na standardni izlaz, prijateljskom operatorskom funkcijom << za upis podataka o studentu u tekstualni tok podataka.

U funkciji main kreirati vektor objekata klase Student u dinamičkoj zoni memorije, učitati njihove podatke, urediti niz prema prosečnoj oceni i uređeni niz prikazati na standardni izlaz.

Nakon toga, urediti niz studenata po najkraćem vremenu između prvog i zadnjeg položenog ispita i uređeni niz prikazati na standardni izlaz.

Na kraju sortirati studente po broju položenih ispita i prikazati sortiranu listu na standardni izlaz.

Zadatak 2.

U klasi Poligon, kreiranu u prvoj vežbi, dodati:

- funkciju koja preslikava poligon simetrično u odnosu na x osu,
- funkciju koja preslikava poligon simetrično u odnosu na y osu.

Funkciju za učitavanje koordinata temena sa standardnog ulaza zameniti prijateljskom operatorskom funkcijom >> za učitavanje koordinata iz tekstualnog toka podataka, a funkciju za prikaz koordinata temena na standardni izlaz, prijateljskom operatorskom funkcijom << za upis koordinata u tekstualni tok podataka.

Dodati u projekat klasu Slika koja od privatnih članova ima maksimalan broj poligona na slici i dinamički niz tipa Poligon, a od javnih članova funkcije za dodavanje poligona u niz kao i operator za štampanje.

U funkciji main kreirati objekat klase Poligon u dinamičkoj zoni memorije sa n temena, učitati koordinate temena, i preslikati ga simetrično u odnosu na x, a onda i u odnosu na y osu i nakon svakog preslikavanja prikazivati koordinate temena na standardni izlaz. Isto to uraditi i sa jednim objektom klase Poligon sa m temena. Kreirati jedan objekat klase Slika sa maksimalno k elemenata i dodati mu kreirana dva poligona, i dodatno kreirati još k-2 tako sto ćete njihove podatke uneti sa standardnog ulaza.

Zadatak 3.

U klasu koja opisuje crno bele slike (nadalje će biti zvana Image), koja je kreirana u prvoj vežbi, dodati:

- funkciju koja preklapa sadržaje dveju slika (u rezultujućoj slici piksel na poziciji i,j dobija vrednost 1, ako je bar u jednoj od slika koje se preklapaju na toj poziciji bila vrednost 1),
- funkciju koja rotira sliku za 90 stepeni u smeru kazaljke na satu,
- operatorsku funkciju = za dodelu jednog objekta klase drugom.

Funkciju za učitavanje sadržaja slike sa standardnog ulaza zameniti prijateljskom operatorskom funkcijom >> za učitavanje sadržaja slike iz tekstualnog toka podataka, a funkciju za prikaz sadržaja slike na standardni izlaz, prijateljskom operatorskom funkcijom << za upis sadržaja slike u tekstualni tok podataka.

Dodati u projekat klasu Kolekcija koja od privatnih članova ima maksimalan broj slika u kolekciji i dinamički niz tipa Image, a od javnih članova funkcije za dodavanje slika u kolekciju i za štampanje.

U funkciji main kreirati 2 objekta klase Image u dinamičkoj zoni memorije, učitati njihove sadržaje, invertovati ih i izvršiti njihovo preklapanje. Rezultujuću sliku dodeliti trećem objektu klase Image i njegov sadržaj prikazati na standardni izlaz. Kreirati jedan objekat tipa Kolekcija, dodati mu kreirane slike i odštampati ga na standardni izlaz.

Zadatak 4.

Kreirati klasu Complex za modelovanje kompleksnih brojeva. Od privatnih članova treba da ima po jednu double promenljivu za realni i imaginarni deo kompleksnog broja, a od javnih funkcije za unos, štampanje, sabiranje i oduzimanje, računanje modula, poređenje po modulu, kao i sve neophodne konstruktore i funkcije za pristup privatnim članovima.

Klasu Buffer, kreiranoj u prvoj vežbi, prepraviti da radi sa objektima tipa Complex a ne sa celobrojnim podacima, i u njoj dodati:

- funkciju koja nadovezuje sadržaj jednog na sadržaj drugog bafera,
- funkciju koja iz bafera izbacuje element sa zadate lokacije,
- funkciju koja iz bafera izbacuje sve elemente manje od prosleđene vrednosti,
- funkciju koja računa prosečnu vrednost elemenata u baferu,
- operatorsku funkciju = za dodelu jednog objekta klase Buffer drugom.

Funkciju za prikaz sadržaja bafera na standardni izlaz zameniti prijateljskom operatorskom funkcijom << za upis sadržaja slike u tekstualni tok podataka.

U funkciji main kreirati nekoliko objekta klase Buffer, u njih upisati po nekoliko elemenata, nadovezati ih i rezultujući objekat dodeliti novom objektu klase Buffer. Iz tog objekta izbaciti neki od unetih elemenata i prikazati ga na standardni izlaz.

Zadatak 5.

Kreirati klasu Tacka za modelovanje trodimenzionalnih tačaka. Od privatnih članova klasa treba da sadrži po jednu double promenljivu za svaku od 3 dimenzije, a od javnih funkcije za unos, množenje skalarom, štampanje, sabiranje i oduzimanje i sve neophodne konstruktore i funkcije za pristup privatnim članovima.

U klasi Vektor, kreiranoj u prvoj vežbi, zameniti niz celih brojeva nizom elemenata tipa Tacka i dodati:

- funkciju koja sabira elemente dva niza iste dužine,
- funkciju koja izbacuje sve duplikate iz niza i uređuje preostale vrednosti u rastući redosled,
- funkciju koja rotira sadržaj vektora za jedno mesto udesno,
- funkciju koja rotira sadržaj vektora za jedno mesto ulevo,
- operatorsku funkciju = za dodelu jednog objekta klase Vektor drugom,
- operatorsku funkciju >> za učitavanje sadržaja niza iz zadatog tekstualnog toka,
- operatorsku funkciju << za upis sadržaja niza u tekstualni tok podataka.

U funkciji main kreirati nekoliko objekta klase Vektor i na standardni izlaz prikazati njihov zbir, proizvod, rezultat sabiranja sa konstantom n i rezultat množenja konstantom m i drugog niza. Rezultat svake operacije najpre dodeliti novom objektu i sadržaj tog objekta prikazati.

Zadatak 6.

U klasi Minesweeper, kreiranu u prvoj vežbi, dodati:

- funkciju koja sve mine u minskom polju pomera za jedno mesto udesno (one iz poslednje kolone prenosi u prvu),
- funkciju koja sve mine u minskom polju pomera za jedno mesto nadole (one iz poslednje vrste prenosi u prvu),
- funkciju koja sve mine u minskom polju pomera za jedno mesto ulevo (one iz prve kolone prenosi u poslednju),
- funkciju koja sve mine u minskom polju pomera za jedno mesto nagore (one iz prve vrste prenosi u poslednju),
- funkciju koja svako n-to prazno polje menja minom,
- operatorsku funkciju >> za učitavanje sadržaja minskog polja iz zadatog tekstualnog toka,
- operatorsku funkciju << za upis sadržaja minskog polja u tekstualni tok podataka.

U funkciji main kreirati objekat klase Minesweeper i translirati mine u sva četiri smera. Nakon svakog transliranja novi sadržaj minskog polja prikazati na standardni izlaz.

Zadatak 7.

Kreirati klasu Tacka za modelovanje trodimenzionalnih tačaka. Od privatnih članova klasa treba da sadrži po jednu double promenljivu za svaku od 3 dimenzija, a od javnih funkcije za unos, štampanje, sabiranje i oduzimanje i sve neophodne konstruktore i funkcije za pristup privatnim članovima.

U klasi Matrica, kreiranoj u prvoj vežbi, zameniti niz double brojeva nizom elemenata tipa Tacka i dodati:

- funkciju koja povećava vrednost svake komponente svake tačke za 1,
- funkciju koja sumira sve elementa iz vrste čiji se indeks prosleđuje kao parametar funkcije,
- funkciju koja normalizuje sve koordinate u okviru prosledjenih granica. Funkciji se proslede minimalne i maksimalne vrednosti za svaku od koordinata. Funkcija sve vrednosti manje od zadatih minimuma menja minimumima, i vrednosti veće od odgovarajućih maksimuma menja zadatim maksimumima,
- operatorsku funkciju = za dodelu jednog objekta klase Matrica drugom,
- Funkciju koja vraća Tacku koja se nalazi na prosleđenim koordinatama,
- prijateljsku operatorsku funkciju >> za učitavanje sadržaja matrice iz zadatog tekstualnog toka,
- prijateljsku operatorsku funkciju << za upis sadržaja matrice u tekstualni tok podataka.

U funkciji main kreirati objekat klase Matrica dimenzija $m \times n$, učitati vrednosti ćelija matrice. Zatim sumirati vrednosti kolone k i l kao redova p i q , demonstrirati upotrebu svih implementiranih funkcija, a potom odštampati matricu.

Zadatak 8.

U klasi String, kreiranu u prvoj vežbi, dodati:

- A) funkciju koja vrši nadovezivanje dva stringa,
- B) funkciju koja prvo slovo u stringu konvertuje u veliko,
- C) funkciju koja sva slova u stringu konvertuje u velika,
- D) funkciju PadRight koja sa desne strane proširuje string prosleđenim karakterom dok ne dostigne zahtevanu dužinu,
- E) operatorsku funkciju = koja jedan objekat klase string dodeljuje drugom.

Funkciju za učitavanje stringa sa standardnog ulaza zameniti prijateljskom operatorskom funkcijom >> za učitavanje stringa iz tekstualnog toka podataka, a funkciju za prikaz stringa na standardni izlaz, prijateljskom operatorskom funkcijom << za upis stringa u tekstualni tok podataka.

Dodati u projekat klasu Paragraph koja od privatnih članova ima maksimalan broj stringova i dinamički niz tipa String, a od javnih članova funkcije za dodavanje stringa u paragraf kao i za prikaz.

U funkciji main kreirati nekoliko objekta klase string, učitati njihove vrednosti, i sačuvati kopiju svakog od unetih stringova. Testirati funkcije B, C i D nad unetim stringova, i tako promenjene stringove konkatenerati i odštampati konkatenerani string. Kreirati jedan objekat klase paragraf, dodati mu sve kopije početnih stringova stringove i prikazati paragraf na standardni izlaz.

Zadatak 9.

U klasi Picture, koja je kreirana u prvoj vežbi, dodati:

- funkciju koja preklapa sadržaje dveju slika (u rezultujućoj slici piksel na poziciji i,j dobija vrednost srednju vrednost piksela iz polaznih slika),
- funkciju koja broji različite boje na slici, odnosno vraća broj različitih vrednosti kojima su opisani pikseli,
- funkciju = za dodelu jednog objekta klase Picture drugom,
- funkciju za učitavanje sadržaja slike sa standardnog ulaza zameniti prijateljskom operatorskom funkcijom >> za učitavanje sadržaja slike iz tekstualnog toka podataka,
- funkciju za prikaz sadržaja slike na standardni izlaz zameniti prijateljskom operatorskom funkcijom << za upis sadržaja slike u tekstualni tok podataka.

Dodati u projekat klasu Kolekcija koja od privatnih članova ima maksimalan broj slika u kolekciji i dinamički niz tipa Image, a od javnih članova funkcije za dodavanje slika u kolekciju kao i operator za prikazivanje elemenata kolekcije.

U funkciji main kreirati nekoliko objekta klase Picture reda $m \times n$, učitati njihove sadržaje i testirati sve kreirane funkcije. Kreirati jedan objekat tipa Kolekcija, dodati mu sve kreirane slike i prikazati taj objekat na standardni izlaz.