
PROGRAMSKI JEZICI 1 (13.2.2017.)

1. Napisati generičku klasu **Vector<T, N>** koja predstavlja N dimenzioni vektor podataka tipa **T**. Klasu je potrebno realizovati bez korištenja struktura podataka iz standardne biblioteke, korištenjem dinamičke alokacije niza. Osim pravila petorke, nad klasom je potrebno realizovati sljedeće elemente.
 - Realizovati štampanje na izlazni tok metodom **print(std::ostream&, std::functional<bool(T)>)**, pri čemu će se izvoditi štampanje samo onih elemenata koji zadovoljavaju uslov definisan funkcijom.
 - Operator za ispis na izlazni tok, tako da se omogući štampanje potomaka na standardni izlaz, bez potrebe za definisanjem novih operatora za ispis, pri čemu je potrebno da se na izlaz ispišu svi elementi vektora.
 - Napisati konstruktor koji prihvata inicijalizatorsku listu elemenata vektora. Ukoliko se dimenzionalnost inicijalizatorske liste ne poklapa sa dimenzionalnošću vektora, podići standardni izuzetak sa porukom „*Invalid number of elements.*“. Izbjeći curenje memorije, s obzirom na činjenicu da konstruktor baca izuzetak.
 - Statički onemogućiti instanciranje klase čija je vrijednost parametra **N** manja od 0. Statički onemogućiti instanciranje klase sa tipom **T** koji ne zadovoljava definisanu osobinu **std::is_arithmetic**.
 - Realizovati metodu **size()** koja vraća trenutni broj elemenata u vektoru. Rezultat ove metode treba da bude određen za vrijeme kompajliranja.
 - Realizovati aritmetičke operatore za sabiranje i oduzimanje, kao i sabiranje sa dodjelom i oduzimanje sa dodjelom, bez dupliranja koda, tako da aritmetički operatori bez dodjele vraćaju pokazivač na rezultat koji je u dinamičkoj memoriji, a aritmetički operatori sa dodjelom vraćaju referencu na tekući objekat.
 - Omogućiti nasljeđivanje (pogledati 2. zadatak).
2. Napisati klasu **Polinom<N>**, izvedenu iz **Vector<double, N>**, koja predstavlja polinom **N**-tog stepena. Pored eventualnih konstruktora i destruktora, potrebno je implementirati i sljedeće elemente.
 - Realizovati sve aritmetičke operatore iz natklase, na takav način da nema dupliranja koda, a da operatori vraćaju pokazivače i reference na klasu **Polinom<N>**, a ne na baznu klasu.
 - Napisati konstruktor koji prihvata inicijalizatorsku listu koeficijenata, a ponaša se na isti način kao konstruktor natklase.
 - Preklopiti **operator ()**, tako da vraća vrijednost polinoma za neku proizvoljnu vrijednost promjenljive *x*.
 - 2.1. Kako se zove mehanizam koji omogućava realizovanje prve stavke?
 - 2.2. Da je natklasa imala preklopljen operator indeksiranja koji vraća referencu na *i*-ti element, a izmjena koeficijenata polinoma nije dozvoljena, da li bi se mogao taj operator obrisati u izvedenoj klasi? Zašto?
3. Odgovoriti na sljedeće stavke.
 - 3.1. Napisati elementaran primjer tipa koji nema konstruktor kopije.
 - 3.2. Napisati primjer funkcije čiji rezultat će aktivirati konstruktor sa pomijeranjem povratnog tipa **T**, pod uslovom da takav konstruktor postoji.
 - 3.3. Napisati proizvoljan četvorobajtni binarni podatak u obliku binarnog literala sa separacijom na svakoj četvrtoj cifri.
 - 3.4. Koje metode (naziv i semantika povratnog tipa) mora da ima klasa koje je instancirani objekat *xx* da bi se mogla koristiti for petlja sa sljedećom sintaksom: **for (auto& x : xx)**?

NAPOMENE:

- Minimizirati broj linija koda i izbjeći dupliranje koda.
- Ispravno označiti metode koje sigurno bacaju izuzetke i koje sigurno ne bacaju izuzetke, kao i konstante metode.
- Poštovati konvencije, razdvojiti interfejs od implementacije.