

Programski jezici 1 - Pismeni ispit (19.04.2021.)

Napomena: Pridržavati se principa objektno-orijentisanog programiranja i pravilno izvesti enkapsulaciju i skrivanje informacija. Omogućiti duboko kopiranje i optimizacije performansi pri pomjeranju. Pravilno omogućiti upotrebu svih klasa u konstantnom kontekstu. Interfejsi moraju biti apstraktne klase. Koristiti specifikatore **noexcept** i **override**. Za dinamičku alokaciju obavezno koristiti **new** i **delete**. Korištenje STL struktura podataka nije dozvoljeno. Korištenje pametnih pokazivača je dozvoljeno samo za parametre metoda i povratne vrijednosti.

1. **(12)** Definirati šablonski interfejs **ICollection<T>** kojim se predstavlja kolekcija podataka. Interfejs mora da sadrži metodu za dodavanje elementa i za dobavljanje broja elemenata. Interfejs mora da sadrži metode za pretragu i za uklanjanje elemenata kojima se prosljeđuje funkcijski objekat za odabir elementa na osnovu proizvoljnog uslova. Omogućiti proširivanje kolekcije novim elementom korištenjem odgovarajućeg operatora sa dodjelom koji se može ulančano pozivati, kao i korištenjem ekvivalentnog operatora bez dodjele i bez bočnih efekata. Pravilno specificirati ponašanje samo za navedene operatore za dodavanje elementa korištenjem metoda dostupnih u interfejsu. Dodati metodu bez bočnih efekata za filtriranje elemenata kolekcije koja kao rezultat vraća filtriranu kolekciju, a kao parametre prihvata potrebne funkcijske objekte.
2. **(24)** Definirati šablonsku klasu **Set<T>** koja pravilno implementira interfejs **ICollection**. Ova klasa treba da predstavlja dinamički alociran skup elemenata proizvoljnog tipa podatka. Skup ne smije da sadrži ponovljene elemente (duplikate). Omogućiti inicijalizaciju skupa inicijalizatorskom listom i inicijalizaciju praznog skupa. Skup mora da bude proširiv. Dva skupa se smatraju jednakim samo ako svaki element koji pripada jednom skupu pripada i drugom bez obzira na redoslijed u memoriji. Korištenjem koncepata, ograničenja i statičkih pretpostavki obezbijediti ograničenja: 1) za tip **T** mora da postoji kopirajući konstruktor, 2) objekti tipa **T** se mogu ispisati i 3) dva objekta tipa **T** se mogu sabrati i rezultat je tipa **T**. Omogućiti korištenje iteratorske *for* petlje nad skupom.
3. **(18)** Definirati interfejs **IShape** koji predstavlja zatvoreni geometrijski oblik i omogućava dobavljanje njegove površine. Interfejs **IShape** treba da sadrži čistu virtuelnu metodu kojom se provjerava da li se tačka u 2D prostoru koja je parametar metode nalazi unutar zatvorenog geometrijskog oblika predstavljenog objektom implementirajuće klase. Definirati imutabilnu klasu **Rectangle** koja predstavlja pravougaonik u 2D prostoru koji je poravnat sa koordinatnim osama i koja isključivo implementira interfejs **IShape**. Pri inicijalizaciji objekata klase **Rectangle**, koristiti sopstveni tip izuzetka za obradu graničnih slučajeva. Klasa **Rectangle** treba da sadrži nestatičku metodu **intersects(...)** i istoimenu statičku metodu kojima se provjerava da li je presjek dva pravougaonika neprazan skup. Omogućiti skaliranje pravougaonika u odnosu na njegov centar korištenjem komutativne operacije za množenje skalarom. Definirati **operator+** tako da zbir dva objekta tipa **Rectangle** predstavlja pravougaonik minimalne površine koji u 2D prostoru u potpunosti obuhvata oba pravougaonika predstavljena operandima. Omogućiti ispis objekta koji predstavlja pravougaonik, tako da se na izlazni tok ispišu koordinate centra i dužine stranica. Definirati finalnu klasu **Square** koja predstavlja kvadrat u 2D prostoru i omogućiti sve iste prethodno navedene funkcionalnosti koje postoje i kod klase **Rectangle**.
4. **(6)** U glavnoj funkciji programa demonstrirati: a) pomjerajuću konstrukciju kolekcije tipa **Set** kao i korištenje operatora dodjele, b) filtriranje elemenata kolekcije tipa **Set** korištenjem lambda operatora, c) primjenu jedne od implementiranih operacija u klasi **Rectangle**, d) obradu izuzetka proizvoljnog tipa bačenog u nekoj od implementiranih metoda.