
PROGRAMSKI JEZICI 1 (16.6.2017.)

1. Napisati program koji omogućava rješavanje sljedećeg problema. Niz riječi učitava se sa standardnog ulaza dok se ne učitava riječ „STOP“. Transformisati taj ulaz na sljedeći način.
 - Napisati klasu **InputReader** koja omogućava učitavanje niza riječi sa ulaznog toka i transformaciju tog niza u oblik **std::vector<std::string>**. Ulazni tok se proslijeđuje kroz konstruktor, a pozivom metode **getWords()** se podaci vraćaju u traženom obliku. Konstruktor ne treba da vrši učitavanje sa ulaznog toka.
 - Napisati generičku klasu **Counter<T>**, privatno naslijeđenu iz klase **std::unordered_set<T>**, koja omogućava prebrojavanje elemenata nekog objekta tipa **std::vector<T>** proslijeđenog kroz konstruktor. Prebrojavanje se vrši pozivom metode **count(const std::function<bool(T)>&)** koja treba da vrati uređeni par čiji elementi sadrže broj jedinstvenih elemenata u kolekciji i broj elemenata koji zadovoljavaju uslov definisan proslijeđenom lambda funkcijom.
 - Statički onemogućiti kompajliranje uz odgovarajuću poruku, ukoliko tip **T** ne zadovoljava uslove neophodne za izračunavanje pomenute uređene trojke, podrazumijevajući da postoji tipska osobina (eng. *type trait*) koja određuje da li se dati tip može porediti (smatrati da ona postoji u standardnoj biblioteci i pozvati je proizvoljnim imenom).
 - Napisati funkciju **filter<T>(const vector<T>&, std::function<bool(T)>&)** koja prihvata vektor elemenata tipa **T** i funkciju za odabir. Funkcija treba da vrati vektor elemenata tipa **T** koji su zadovoljili uslov definisan funkcijom za odabir.
 - Napisati generičku funkciju **convertType(const A&)** u prostoru imena **utilities**, koja prihvata proizvoljan element tipa **A** i pretvara ga u **std::vector**, pretpostavljajući da će proslijeđeni objekat imati definisan iterator.
 - 1.1. Napisati liniju koda kojom se instancira varijabla istog tipa kao tip iteratora nad proslijeđenim objektom klase **A**.
 - Iskoristiti instancu klase **InputReader** za učitavanje traženog niza riječi, a potom, korištenje funkcije **filter(...)**, konstruisati novi vektor stringova koji sadrži samo one riječi koje sadrže samo cifre.
 - Konvertovati taj vektor stringova u vektor vektora karaktera, pozivom napisane funkcije za konverziju.
 - Korištenjem objekta klase **Counter**, prebrojati sve elemente dobijenog vektora čiji su prvi i posljednji elementi jednaki.
 - Sortirati dobijeni vektor u rastućem poretku prema prvom elementu elementa, a potom elemente vektora ispisati na standardni izlaz kao nizove karaktera, pri čemu je za ispis na standardni izlaz potrebno koristiti **for_each** funkciju iz biblioteke **algorithm**.
2. Odgovoriti na sljedeće stavke.
 - 2.1. Napisati primjer funkcije koja vrši alokaciju niza od n elemenata, a u slučaju da alokacija baci izuzetak, ona ne dovodi do propagiranja greške kroz stek, već kao rezultat alokacije vraća prazan pokazivač.
 - 2.2. Napisati elementarne klase **A** i **B**, takve da je **B** izvedena iz **A**. Neka je prazna metoda **f()** redefinisana u izvedenoj klasi. Napisati dva elementarna primjera aktivacije polimorfizma (pri čemu oba samo jedan primjer to treba da ilustruje sa pokazivačima).
 - 2.3. Napisati proizvoljan četvorobajtni binarni podatak djeljiv sa 192 u obliku binarnog literala sa separacijom na svakoj osmoj cifri.
 - 2.4. Ukoliko postoji dinamička alokacija u nekoj klasi, koje tri funkcije moraju da budu implementirane da bi ona ispravno radila?
 - 2.5. Napisati elementaran primjer izvođenja neke klase iz **std::vector<int>**, tako da je omogućen polimorfizam.
 - 2.6. Napisati elementaran primjer nasljeđivanja sa metodom sa kovarijantnim povratnim tipom.
 - 2.7. Na dva načina definisati generičku metodu **add(const A&, const B&)** za sabiranje dva podatka klase **A** i **B**, podrazumijevajući da postoji operator sabiranja nad njima.
 - 2.8. Sljedeću liniju koda napisati na kraći način, tako da naziv klase ostane isti: **class Point{public: int x, y; protected: int z;};**
 - 2.9. Napisati funkciju **f(int n)** koja izračunava vrijednost faktoriijela od n , tako da se izračunata vrijednost dobija za vrijeme kompajliranja, te da se njena povratna vrijednost može iskoristiti kao argument šablona.
 - 2.10. Napisati elementaran primjer definicije klase čiji se objekat ne može instancirati, a koja se ponaša polimorfno. Kako se naziva ovakva klasa?

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.