
PROGRAMSKI JEZICI 1

Zadatak 6 – Standardna biblioteka

Postavka

Kreirati projekat pod nazivom *Zadatak6* unutar rješenja (i.e. *Visual Studio solution*) pod nazivom *PJ1Zadaci2020*, te kreirati projekat pod nazivom *Zadatak6.Demo* unutar istog rješenja. Projekat *Zadatak6.Demo* treba da referencira biblioteku iz projekta *Zadatak6* i demonstrira njen rad.

Zadatak

Unutar projekta *Zadatak6*, kao statičku biblioteku, implementirati sve neophodne klase za rješavanje zadatka, a u projektu *Zadatak6.Demo* implementirati rješenje. Pri implementaciji je neophodno koristiti kolekcije i funkcije iz standardne biblioteke. Izbjeći dinamičku alokaciju memorije pri definiciji klasa. Definirati nove klase samo u slučajevima kada je to neophodno i kada ne postoje odgovarajuće klase iz standardne biblioteke za rješavanje nekog podproblema.

- U fajlu pod nazivom *graph.data* upisan je usmjeren težinski graf predstavljen kao matrica susjednosti, gdje su redovi matrice odvojeni novim linijama, a kolone odvojene *tab* znakom.
- Učitati graf u memoriju i predstaviti ga na odgovarajući način, u skladu sa principima OOP programiranja.
 - ↪ Obezbijediti klase za rad sa neusmjerenim i usmjerenim, težinskim i netežinskim grafovima i ispravno ih strukturisati u hijerarhiju. Implementirati samo one metode neophodne za rješavanje zadataka.
 - ↪ Obezbijediti jednosmjerne iteratore za prolazak kroz čvorove grafa, korištenjem BFS i DFS pristupa. Izbjeći dupliranje koda između različitih iteratora.
- Sa standardnog ulaza učitati indekse dvaju čvorova iz grafa i odrediti najkraći put između ta dva čvora.
- Interpretirajući graf kao neusmjeren i netežinski, odrediti da li je moguće sve njegove čvorove označiti sa **true** ili **false**, tako da se dvije iste istinitosne vrijednosti nikada ne nađu na susjednim čvorovima (i.e. tako da dva susjedna čvora nikada nisu oba **true** ili oba **false**).
- Interpretirajući graf kao usmjeren ali netežinski, odrediti najkraće rastojanje od učitanoг početnog čvora do učitanoг krajnjeg čvora.
- Interpretirajući graf kao neusmjeren i težinski, odrediti najkraću zatvorenu putanju kroz sve čvorove grafa.
- Interpretirajući graf kao usmjeren i težinski, odrediti najkraću zatvorenu putanju kroz sve čvorove grafa.

Sve implementirane klase treba da poštuju pravilo trojke i pravilo petorke, budu objedinjene u jedinstven prostor imena i imaju ispravno razdvojena zaglavlja od implementacije. Preklopiti operatore u svim slučajevima kada je to semantički ispravno. Izbjeći svako dupliranje koda. Omogućiti podizanje i obradu izuzetaka u slučajevima kada je to neophodno, te ispravno označiti sve metode s obzirom na korištenje izuzetaka i konstantni kontekst.