

KOLOKVIJUM

C++/Java

Napraviti program koji simulira rad playliste.

Delove 1 i 2 implementirati u C++ i u Javi, dok deo 3 implementirati u C++

1) Napisati klasu **Video** koje ima sledeca polja:

- *naziv*, tipa **DinString** (**String** u Javi)
- *trajanje*, tipa **int**
- *rezolucijaVidea*, tipa **DinString** (**String** u Javi)

Za klasu **Video** implementirati sledece:

- Konstruktor bez parametara (postavite polja na vrednosti po želji)
- Konstruktor sa parametrima (za sve parametre):

Video (const DinString&, int, const DinString&) - C++

Video(String, int, String) - Java

- getere za sva polja
- setere za sva polja

2) Napisati klasu **VideoList** koja ima sledeca polja:

- *listaVidea*, tipa **List<Video>** (**ArrayList<Video>** u Javi)

Za klasu **VideoList** implementirati sledeće:

- Konstruktor bez parametara koji listu inicijalizuje kao praznu listu
- Metodu **bool dodajVideo(Video&) (boolean dodajVideo(Video)** u Javi) koja dodaje muzicki video na kraju liste videa ukoliko vec ne postoji video sa istim nazivom. Ukoliko se prosledjeni video uspesno ubaci u listu, vratiti **true**, u suprotnom **false**.
- Metodu **bool obrisiVideo(const DinString&) (boolean obrisiVideo(String)** u Javi) koja brise video sa prosledjenim nazivom, ukoliko takav video postoji. Vratiti **true** ako je uspesno pronadjen i obrisan video u suprotnom vratiti **false**.

3) Napisati klasu **YoutubePlaylist** koja nasledjuje klasu **VideoList** i ima sledeca polja:

- *nazivPlayliste*, tipa **DinString**
- *nazivAutora*, tipa **DinString**

Za klasu **YoutubePlaylist** implementirati sledece:

- Konstruktor bez parametara(postavite polja na vrednosti po želji)
- Konstruktor sa parametrima (za sve parametre):

YoutubePlaylist(const DinString&, const DinString&)

- Metodu **int najduziVideo()const** koja vraca index sa pozicijom najduzeg videa u playlisti, vratiti -1 ako je lista prazna
- Preklopiti **operator<<** za ispis koji ispisuje vrednosti svih polja klase, informacije o svim trenutnim videima u listi koji imaju duzinu trajanja vecu od prosečne duzine trajanja videa u listi, kao i podatak o najduzem videu iz liste

Napomene:

- U C++ za stringove koristiti klasu **DinString** , a za listu klasu **List** koje su date.
- Sva polja možete implementirati kao private ili protected (ne public) u zavisnosti od vašeg izbora načina implementacije konstruktora i metoda u zadatku.
- program se **mora uspešno kompajlirati kad u main.cpp uključite (include) sve hpp fajlove za klase iz zadatka**, u suprotnom možete dobiti najviše 1 poen.