

1	2	3	Σ

1. (10 bodova) Teorijski zadatak uraditi na poledini lista.

- Objasniti kada je neophodno definisati podrazumijevani konstruktor.
- Šta je to konstantna referenca? Dati adekvatan primjer upotrebe.
- Objasniti razliku između reference i pokazivača? Šta će biti odštampano (argumentovati) na izlazu nakon izvršavanja narednog koda:

```
int a, b = 3;  
a = b / 2;  
int &c = a;  
b++;  
a++;  
cout << c++;
```

2. (8 bodova) Realizovati klasu **Predmet** koja sadrži:

- naziv predmeta (pokazivač na niz karaktera koji mogu biti slova i brojevi), broj časova predmeta (cijeli broj) i koeficijent predmeta (realan broj);
- podatak o nazivu predmetu koji ima najviše časova (u obzir uzeti sve kreirane predmete u glavnom programu);
- potrebne konstruktore, inspektore i mutatore;
- preklopiti operator + koji kao rezultat treba da vrati objekat koji se dobija tako što se nadovežu nazivi predmeta („Programiranje I & Programiranje II“), sabere broj časova i odabere veći koeficijent od ta dva predmeta;
- preklopiti operatore postfixnog i prefiksnog dekrementiranja kojima se umanjuje broj časova predmeta;
- privatnu metodu koja računa normu za predmet koja predstavlja proizvod broja časova i koeficijenta predmeta;
- metodu koja štampa sve podatke o predmetu.

(2 boda) Realizovati prijateljsku funkciju koja od dva predmeta vraća onaj koji je teži. Teži predmet predstavlja onaj predmet čija je norma veća.

(3 bodova) Napisati glavni program u kojem ćete demonstrirati rad svih realizovanih funkcija.

3. (13 bodova) Za izradu ovog zadatka kreirati novi projekat u kojem možete koristiti kod iz prethodnog zadatka.

Kreirati klasu **Saradnik** koja sadrži sljedeće podatke članove i metode:

- pokazivač na niz predmeta (niz objekata klase Predmet) i njegovu dužinu;
- metodu za štampanje podataka o svim predmetima na kojima saradnik drži nastavu;
- operator [] koji treba da vrati predmet na prosljeđenoj poziciji. Ukoliko je pozicija van opsega niza predmeta, vratiti novi predmet koji se dobija pozivom podrazumijevanog konstruktora;
- metodu koja računa ukupnu normu saradnika. Prilikom implementacije koristiti metodu norma iz klase Predmet;

Napomena: U klasi dodati neophodne konstruktore, prijateljstva, destruktor i operatore (koji eventualno nisu navedeni) kako bi se obezbijedilo pravilno funkcionisanje zadatih funkcija.

(4 bodova) Napisati glavni program u kojem ćete demonstrirati rad svih realizovanih funkcija.

1	2	3	Σ

1. (10 bodova) Teorijski zadatak uraditi na poledini lista.

- a) Preklapanje imena funkcija. Kako se vrši razrješenje poziva? Pretpostavimo da imamo prototipe sljedećih funkcija:
`int fun(int,int)`, `int fun(double,double)` i `int fun(int,double)`.
Koja će se metoda pozvati za `fun(3.1,5)`?
- b) Čemu služi ključna riječ `const`.
- c) Objasniti razliku između reference i pokazivača? Šta će biti odštampano (argumentovati) na izlazu nakon izvršavanja narednog koda:

```
int a, b = 3;  
a = b / 2;  
int &c = a;  
b++;  
a++;  
cout << ++c;
```

2. (8 bodova) Realizovati klasu **Avion** koja sadrži:

- Naziv modela aviona (pokazivač na niz karaktera koji mogu biti slova i brojevi), broj časova leta (cijeli broj) i potrošnju na 100km leta (realan broj);
- Podatak o nazivu aviona koji ima najviše sati leta (u obzir uzeti sve kreirane avione u glavnom programu);
- potrebne konstruktore, inspektore i mutatore;
- preklopiti operator `()` koji kao argument ima cio broj koji predstavlja distancu, a kao rezultat vraća potrošnju goriva za pomenutu distancu;
- preklopiti operatore postfixnog i prefiksnog inkrementiranja kojima se uvećava broj časova leta;
- metodu koja štampa sve podatke o avionu.

(2 boda) Realizovati prijateljsku funkciju koja od dva aviona vraća onaj koji je do sada potrošio više goriva na svim obavljenim letovima.

(3 bodova) Napisati glavni program u kojem ćete demonstrirati rad svih realizovanih funkcija.

3. (13 bodova) Za izradu ovog zadatka kreirati novi projekat u kojem možete koristiti kod iz prethodnog zadatka.

Kreirati klasu **Flota** koja sadrži sljedeće podatke članove i metode:

- pokazivač na niz aviona (niz objekata klase **Avion**) i njegovu dužinu;
- metodu za štampanje podataka o svim avionima flote;
- operator `--` koji treba da iz flote ukloni avion sa najvećom potrošnjom;
- metodu koja računa prosječnu potrošnju aviona u floti.

Napomena: U klasi dodati neophodne konstruktore, prijateljstva, destruktor i operatore (koji eventualno nisu navedeni) kako bi se obezbijedilo pravilno funkcionisanje zadatih funkcija.

(4 bodova) Napisati glavni program u kojem ćete demonstrirati rad svih realizovanih funkcija.