

Programabilni uređaji i objektno orjentisano programiranje

Laboratorijske vježbe 5

Razlike C i C++ programskih jezika

1. Realizovati funkciju **uvecaj** koja ima sljedeće argumente:

- prvi argument je pokazivač na niz cijelih brojeva **A**;
- drugi argument je dužina niza **N**;
- treći argument je cijeli broj **B**.

Funkcija treba da svaki elemenat niza A, ukoliko je djeljiv elementom koji mu prethodi, uveća za broj B. Prvi elemenat niza ne provjeravati. Ukoliko se funkcija pozove sa dva ulazna argumenta, podrazumijevati da je broj B=10. Napisati i glavni program koji testira funkcionalnost navedene funkcije.

```
#include <iostream>
using namespace std;

void uvecaj(int *, int, int=10);

main(){
    int a[20], i, n;
    cout<<"Unesite duzinu niza"<<endl;
    cin>>n;
    cout<<"Unesite elemente niza"<<endl;
    for(i=0;i<n;i++)
        cin>>a[i];
    uvecaj(a,n,3);
    for(i=0;i<n;i++)
        cout<<a[i]<<" ";
    cout<<endl;
}
```

```
void uvecaj(int *a, int n, int b){
    for(int i=1; i<n; i++)
        if((a[i]%a[i-1])==0)
            a[i]+=b;
}
```

2. Realizovati funkciju **najveci** koja ima tri argumenta:

- pokazivač na niz cijelih brojeva **X**;
- dužinu prosljeđenog niza **N**;
- cio broj **K**.

Funkcija treba da pronađe i vrati najveći elemenat niza koji je djeljiv brojem K. Ukoliko se funkcija pozove sa dva ulazna argumenta, smatrati da je K=1. Napisati i glavni program koji testira realizovanu funkciju.

3. Realizovati funkciju **nadovezivanje** koja uz dva argumenta, stringove **S1** i **S2**, može imati i treći argument, cijeli broj **K**. Ukoliko broj K nije prosljeđen, funkcija treba da na string S1 nadoveže sve karaktere iz stringa S2 koji predstavljaju cifre. Ukoliko je broj K prosljeđen, funkcija treba da na string S1 nadoveže sva mala slova iz stringa S2. Napisati i glavni program koji testira realizovanu funkciju.

4. U ovom zadatku je demonstrirano prosljeđivanje argumenata funkcijama po vrijednosti, preko reference i preko pokazivača. Pokušajte da objasnite sličnosti i razlike između ova tri tipa prosljeđivanja argumenata.

```
#include <iostream>
using namespace std;

//prosljeđivanje argumenata po vrijednosti
void fun1(int a, int b){
    a++;
    b++;
}

//prosljeđivanje drugog argumenta preko reference
void fun2(int a, int &b){
    a++;
    b++;
}

//prosljeđivanje drugog argumenta preko pokazivača
void fun3(int a, int *b){
    a++;
    (*b)++;
}
```

```

}

int main()
{
    int a,b;

    cout<<"Unesi vrijednosti a i b."<<endl;
    cin>>a>>b;

    cout<<"Vrijednosti promjenljivih prije poziva funkcije 1 su: a="<<a<<" b="<<b<<endl;
    fun1(a,b);
    cout<<"Vrijednosti promjenljivih nakon poziva funkcije 1 su: a="<<a<<" b="<<b<<endl<<endl;

    cout<<"Vrijednosti promjenljivih prije poziva funkcije 2 su: a="<<a<<" b="<<b<<endl;
    fun2(a,b);
    cout<<"Vrijednosti promjenljivih nakon poziva funkcije 2 su: a="<<a<<" b="<<b<<endl<<endl;

    cout<<"Vrijednosti promjenljivih prije poziva funkcije 3 su: a="<<a<<" b="<<b<<endl;
    fun3(a,&b);
    cout<<"Vrijednosti promjenljivih nakon poziva funkcije 3 su: a="<<a<<" b="<<b<<endl;
}

```

5. Napisati program kojim se demonstrira zamjena vrijednosti dvije cjelobrojne promjenljive koristeći funkciju **zamjena**. Funkcija kao argumente ima dva cijela broja, pri čemu je potrebno realizovati dva načina prosljeđivanja argumenata, preko pokazivača i preko reference. U glavnom programu je potrebno pozvati pomenutu funkciju i štampati vrijednosti promjenljivih nakon poziva funkcije. Da li je funkciju moguće realizovati prosljeđivanjem argumenata po vrijednosti?

Rad sa klasama

6. Realizovati klasu **automobil**, koja ima sljedeće podatke članove i funkcije članice:

- godinu proizvodnje (cijeli broj), cijenu (realan broj) i kubikažu (cijeli broj);
- odgovarajuće konstruktore i destruktor;
- odgovarajuće mutatore i inspektore za sve podatke članove.

U okviru klase potrebno je realizovati i funkciju članicu **poredjenje** koja od dva automobila treba da vrati onaj automobil sa manjim odnosom (količnikom) cijene i kubikaže. U glavnom programu potrebno je inicijalizovati tri objekta klase automobil (koristeći odgovarajuće konstruktore) i nad njima testirati rad realizovanih mutatora, inspektora i funkcije za poredjenje dva automobila.

7. Realizovati klasu koja predstavlja **pravougaonik**, sa dva privatna podatka člana, realnim brojevima koji predstavljaju dužinu i visinu pravougaonika. U okviru klase potrebno je realizovati odgovarajuće konstruktore, destruktor, mutatore i inspektore. Pored pomenutih funkcija članica, klasa posjeduje:

- funkcije za računanje površine i obima pravougaonika;
- funkciju koja od dva pravougaonika vraća pravougaonik sa većom površinom;
- funkciju koja kao rezultat treba da vrati dijagonalu pravougaonika.

U glavnom programu inicijalizovati proizvoljan broj objekata klase pravougaonik i nad njima demonstrirati rad svih realizovanih funkcija.

8. Realizovati klasu **radnik** koja predstavlja trenutno zaposlene radnike u firmi. Klasa posjeduje privatne podatke članove koji predstavljaju identifikacioni broj radnika (cijeli broj) i godinu zaposlenja radnika (cijeli broj).

U okviru klase potrebno je realizovati sljedeće funkcije članice:

- odgovarajuće konstruktore, destruktor, mutatore i inspektore;
- funkciju za računanje koliko je dugo radnik zaposlen u firmi (broj godina).

U glavnom programu potrebno je inicijalizovati niz radnika (niz objekata klase radnik) proizvoljne dužine i u unijetom nizu pronaći radnika sa najdužim radnim stažom, čije je podatke članove potrebno štampati na izlazu.