

# Struktura

*struct()*

Dejan Ljubobratović, mag. educ. math. et inf.

dejan.ljubobratovic@uniri.hr

Fakultet informatike i digitalnih tehnologija, Sveučilište u Rijeci

- derivirani - izvedeni podatkovni tip (stvara ga programer)
- grupa logički povezanih podataka različitih podatkovnih tipova sa zajedničkim imenom
- podaci (članovi strukture) su međusobno povezani identifikatorom - imenom strukture
- deklaracijom članova strukture određuje se ime i tip pojedinih članova strukture
- ako je definiran izvan tijela funkcije main() - sve funkcije programa mogu koristiti taj tip
- između imena strukturne varijable i imena člana te strukture stavlja se operator “.”

- Struktura je korisnički definiran složeni tip sastavljen je od polja ili članova koji mogu biti različitih tipova
- u C++ struktura je isto što i klasa osim što su njeni članovi podrazumijevano javni
- varijable se kod definiranja strukturnog tipa mogu deklarirati nizanjem jednog ili više imena varijabli odvojenih zarezima između zatvorene zagrade i ';'

- Napraviti ćemo strukturu koju ćemo nazvati *osoba\_str* a koja će biti sastavljena od varijabli:
  - *starost* (int)
  - *ime* (string)
  - *tezina* (float)
- te ćemo deklarirati dvije varijable:
  - *brat* i *sestra*.
- Zatim ćemo unijeti par vrijednosti u te varijable i ispisati starost sestre.

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

struct osoba_str
{
    int starost;
    float tezina;
    string ime;
};

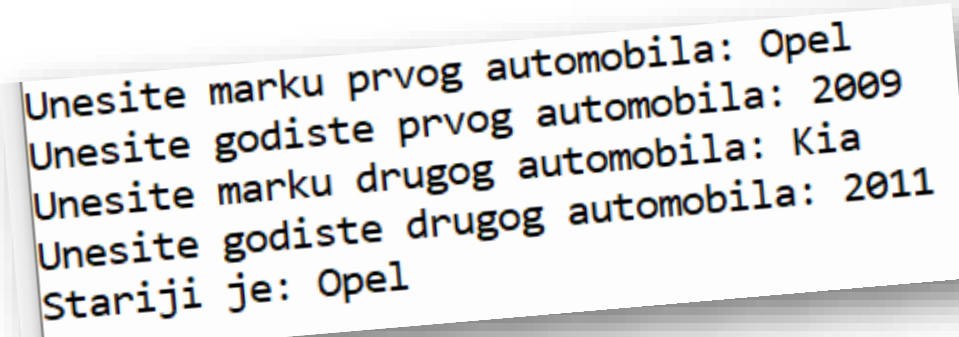
int main ()
{
    int broj;
    osoba_str brat;
    osoba_str sestra;

    brat.starost = 12;
    sestra.starost = 9;
    cout << sestra.starost;
    return 0;
}
```

# Zadatak 1

Napravite program u kojem će korisnik unijeti marku i godište za auto1 i auto2. Program na kraju treba ispisati marku starijeg automobila.

Za izradu programa koristite strukture.

A screenshot of a program's output, showing a sequence of prompts and user inputs. The text is displayed on a white background with a slight shadow, tilted at an angle. The output shows the program asking for the brand and year of two cars, and then determining which one is older.

Unesite marku prvog automobila: Opel  
Unesite godiste prvog automobila: 2009  
Unesite marku drugog automobila: Kia  
Unesite godiste drugog automobila: 2011  
Stariji je: Opel

```
#include <iostream>
using namespace std;
struct automobil_str
{
    string marka;
    int godiste;
};
int main()
{
    automobil_str auto1, auto2;
    cout << "Unesite marku prvog automobila: ";
    cin >> auto1.marka;
    cout << "Unesite godiste prvog automobila: ";
    cin >> auto1.godiste;
    cout << "Unesite marku drugog automobila: ";
    cin >> auto2.marka;
    cout << "Unesite godiste drugog automobila: ";
    cin >> auto2.godiste;
    if (auto1.godiste > auto2.godiste)
        cout << "Stariji je: " << auto2.marka;
    else
        cout << "Stariji je: " << auto1.marka << endl;
    return 0;
}
```

# Zadatak 2

Napravite program u kojem će korisnik unijeti koliko automobila želi unositi.

Zatim unosi marku i godište za svaki od njih. Program na kraju treba ispisati marku najmlađeg automobila.

Za izradu programa koristite strukture.

```
Koliko automobila zelite unesti? 4
Unesite marku i godiste 1. auta: Ford 1998
Unesite marku i godiste 2. auta: Opel 2012
Unesite marku i godiste 3. auta: Mercedes 2020
Unesite marku i godiste 4. auta: Fiat 2017
Najmladji auto je Mercedes
```

```
Process returned 0 (0x0)   execution time : 33.645 s
Press any key to continue.
```

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

struct auto_str
{
    int god;
    string marka;
};

int main()
{
    int n;
    auto_str automobil;
    cout << "Koliko automobila zelite unesti? ";
    cin >> n;
    int naj_god=0;
    string naj_marka;
    for(int i=0;i<n;i++)
    {
        cout<<"Unesite marku i godiste "<<i+1<<". auta: ";
        cin >> automobil.marka >> automobil.god;
        if(automobil.god>naj_god) {
            naj_god=automobil.god;
            naj_marka=automobil.marka;
        }
    }
    cout << "Najmladji auto je " << naj_marka << endl;
}
```

# Zadatak 3

Napravite program u kojem će korisnik unijeti broj studenata, a zatim podatke o tim studentima: ime, prezime, godina i težinu. Program treba ispisati studenta sa najvećom težinom, ali da sačuva podatke o svim studentima.

Ovdje ćemo koristiti **polja struktura**.

```
Unesite broj studenata: 2
Unesi ime 1. studenta: Pero
Unesi prezime 1. studenta: Perić
Unesi godine 1. studenta: 19
Unesi težinu 1. studenta: 78
Unesi ime 2. studenta: Ivo
Unesi prezime 2. studenta: Ivić
Unesi godine 2. studenta: 20
Unesi težinu 2. studenta: 94
```

```
Najteži student je: Ivo Ivic (20 god.) i ima 94 kg
```



# Rješenje zadatka 2

```
#include <iostream>
using namespace std;
struct student_str
{
    string ime, prezime;
    int godina;
    float tezina;
};

int main()
{
    int n;
    cout << "Unesite broj studenata: ";
    cin >> n;
    student_str studenti[n];
    for (int i = 0; i < n; i++)
    {
        cout << "Unesi ime " << i + 1 << ". studenta: ";
        cin >> studenti[i].ime;
        cout << "Unesi prezime " << i + 1 << ". studenta: ";
        cin >> studenti[i].prezime;
        cout << "Unesi godine " << i + 1 << ". studenta: ";
        cin >> studenti[i].godina;
        cout << "Unesi tezinu " << i + 1 << ". studenta: ";
        cin >> studenti[i].tezina;
    }
}
```

```
int maxIndex = 0;
for (int i = 0; i < n; i++)
{
    if (studenti[i].tezina > studenti[maxIndex].tezina)
        maxIndex=i;
}
cout << "Najtezi student je: " << studenti[maxIndex].ime;
cout << " " << studenti[maxIndex].prezime << " (";
cout << studenti[maxIndex].godina << " god.) i ima ";
cout << studenti[maxIndex].tezina << " kg" << endl;
return 0;
}
```