

Конструктори структуре

Пример: У структури Zaposleni са једним пољем id, креирати дифолтни конструктор који ће по дифолту повезати поље структуре са вредности унешене као параметар приликом приступа објекта том пољу.

```
using System;
namespace Proba
{
    struct Zaposleni
    {
        public int id;
        public Zaposleni(int zaposleniId)
        {
            id = zaposleniId;
        }
    }
    class Program
    {
        static void Main()
        {
            Zaposleni radnik1 = new Zaposleni(1);
            Console.WriteLine($"Radnik radnik1 ima id: {radnik1.id}");
        }
    }
}
```

Пример: Модификацијом претходног примера омогућити дифолтни конструктор са параметрима ид и име запосленог.

```
using System;
namespace Proba
{
    struct Zaposleni
    {
        public int id;
        public string ime;
        public Zaposleni(int zaposleniId, string zaposleniIme)
        {
            id = zaposleniId;
            ime = zaposleniIme;
        }
    }
    class Program
    {
        static void Main()
        {
            Zaposleni radnik1 = new Zaposleni(1, "Mica");
            Console.WriteLine($"Radnik radnik1 ima id: {radnik1.id}");
        }
    }
}
```

```

        Console.WriteLine($"Radnik radnik1 se zove:
{radnik1.ime}");
    }
}

```

Задаци за самосталан рад:

1. Креирати дифолтни конструктор структуре Osoba са четири поља: висина, тежина, број браће и број сестара. Креирати два различита објекта овог типа и доделити им неке вредности. Приказати успешну доделу вредности на конзоли.

```

using System;
namespace Proba
{
    struct Osoba
    {
        public double visina;
        public int tezina;
        public int brojBrace;
        public int brojSestara;
        public Osoba(double osobaVisina, int osobaTezina, int
osobaBrojBrace, int osobaBrojSestara)
        {
            visina = osobaVisina;
            tezina = osobaTezina;
            brojBrace = osobaBrojBrace;
            brojSestara = osobaBrojSestara;
        }
    }
    class Program
    {
        static void Main()
        {
            Osoba osoba1 = new Osoba(185.45, 91, 1, 1);
            Console.WriteLine($"Osoba1 ima visinu {osoba1.visina} cm i
tezinu {osoba1.tezina} kg.");
            Console.WriteLine($"Osoba1 ima {osoba1.brojBrace} brace i
{osoba1.brojSestara} sestara.");
        }
    }
}

```

2. Направити апликацију која користи структуру Osoba са четири поља: име, презиме, висина, тежина, број браће и број сестара. Креирати дифолтни конструктор који додељује име и презиме креираном објекту. Креирати две методе, једну која приказује телесне карактеристике особе и другу која приказује колико укупно блиских рођака има особа.

```

using System;
namespace Proba
{
    struct Osoba
    {
        public string ime;
        public string prezime;
        public double visina;
        public int tezina;
        public int brojBrace;
        public int brojSestara;
        public Osoba(string osobaIme, string osobaPrezime, double
osobaVisina, int osobaTezina, int osobaBrojBrace, int
osobaBrojSestara)
        {
            ime = osobaIme;
            prezime = osobaPrezime;
            visina = osobaVisina;
            tezina = osobaTezina;
            brojBrace = osobaBrojBrace;
            brojSestara = osobaBrojSestara;
        }
        public void NajbliziRodjaci(int brojBrace, int brojSestara)
        {
            Console.WriteLine($"Osoba ima {brojBrace} brace i
{brojSestara} sestre.");
            Console.WriteLine($"Ukupno ima {brojBrace + brojSestara}
najblizih rodjaka.");
        }
    }
    class Program
    {
        static void Main()
        {
            Osoba osoba1 = new Osoba("Mica", "Mikic", 185.45, 77, 2,
1);
            Console.WriteLine($"Osoba1 se zove {osoba1.ime}
{osoba1.prezime}.");
            Console.WriteLine($"Osoba1 ima visinu {osoba1.visina} cm i
tezinu {osoba1.tezina} kg.");
            osoba1.NajbliziRodjaci(osoba1.brojBrace,
osoba1.brojSestara);
        }
    }
}

```