

Низови у петљама

Пример: Написати програм који омогућава кориснику унос n целобројних вредности као елемената низа и рачуна збир сваког другог елемента низа.

```
using System;
namespace Proba
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Uneti koliko elemenata niza je potrebno:");
            int n = Convert.ToInt32(Console.ReadLine());
            int[] niz = new int[n];
            Console.WriteLine("Uneti elemente niza: ");
            for(int i = 0; i < n; i++)
            {
                Console.WriteLine($"niz[{i}] = ");
                niz[i] = Convert.ToInt32(Console.ReadLine());
            }
            int suma = 0;
            for (int i = 0; i < n; i++)
            {
                if (i % 2 != 0)
                {
                    suma += niz[i];
                }
            }
            Console.WriteLine($"Zbir svakog drugog elementa niza je: {suma}");
        }
    }
}
```

Дaje:

Uneti koliko elemenata niza je potrebno: 5

Uneti elemente niza:

niz[0] = 0

niz[1] = -2

niz[2] = 3

niz[3] = 1

niz[4] = 0

Zbir svakog drugog elementa niza je: -1

Пример: Написати програм којим се употребом foreach петље и switch структуре извршава match pattern над стринг низом.

```
using System;
namespace Proba
{
    internal class Program
    {
        static void Main(string[] args)
        {
            string[] imena = { "Todor Novic", "Marija Cilic", "Ana
                               Mikic", null, "" };
            foreach (var ime in imena)
            {
                switch (ime)
                {
                    case string t when t.StartsWith("T"):
                        Console.WriteLine("Ime ove osobe pocinje sa
                                          'T': " + $"{ime} i ima {t.Length - 1}
                                          slova");
                        break;
                    case string e when e.Length == 0:
                        Console.WriteLine("U nizu postoji string bez
                                          vrednosti");
                        break;
                    case null:
                        Console.WriteLine("Postoji 'null' vrednost u
                                          stringu");
                        break;
                    case var x:
                        Console.WriteLine("Ovo je var sema za tip: " +
                                          $"{x.GetType().Name}");
                        break;
                    default:
                        break;
                }
            }
        }
    }
}
```

Дaje:

Ime ove osobe pocinje sa 'T': Todor Novic i ima 10 slova

Ovo je var sema za tip: String

Ovo je var sema za tip: String

Postoji 'null' vrednost u stringu

U nizu postoji string bez vrednosti

У примеру се користи switch исказ тако да уместо да реагује на посебну вредност, case команда врши поређење елементима низа са наведеним филтрима.

Исказ `case string t when t.StartsWith("T")` декларише стринг са именом `t` и преко њега се приступа сваком од елемената низа а то значи и методама и особинама података типа стринг.

Метода `StartsWith()` (покренута из `System.String` класе преко филтера `when`) прихвата један параметар и ако елемент низа почиње са тим параметром, извршава се блок кода у том case исказу.

Исказ `case string e when e.Length == 0` декларише стринг са именом `e` који позива `Length` особину `System.String` класе и ако је дужина елемента у низу једнака 0, извршава се блок кода у том case исказу.

Исказ `case null` проверава да ли је вредност елемента једнака `null`.

Исказ `case var` `x` користи `var` декларацију променљиве `x` за дохватање било ког другог типа података у низу.

Тренутно су у низу само стринг типови али је могуће да буди елементи и непознатог типа, па се употребом `GetType()` методе `System.Object` класе, добија којег је елемент типа.

Због употребе овог исказа, битан је редослед постављања исказа у switch структури, пошто да је последњи исказ постављен као први, он би реаговао на сваки елемент у низу пре других провера са case исказима.

Пример: Написати програм којим се употребом `foreach` петље и `switch` структуре извршава `match pattern` над целобројним низом.

```
using System;
namespace Proba
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int suma = 0, brojac = 0, x = 0;
            int?[] niz = new int?[7] { 5, x, 9, 10, null, 2, 99 };
            foreach (var broj in niz)
            {
                switch (broj)
                {
                    case 0:
                        Console.WriteLine($"Celi broj '{brojac}' ima
                            difoltnu vrednost 0");
                        brojac++;
                        break;
                    case int vrednost:
                        suma += vrednost;
                        Console.WriteLine($"Celi broj '{brojac}' ima
                            vrednost {vrednost}");
                        brojac++;
                }
            }
        }
    }
}
```

```

        break;
    case null:
        Console.WriteLine($"Celi broj '{brojac}' je
            null");
        brojac++;
        break;
    default:
        break;
    }
}
Console.WriteLine($"Suma {brojac} celih brojeva je
    {suma}");
}
}
}

```

Дaje:

Celi broj '0' ima vrednost 5  
 Celi broj '1' ima difoltnu vrednost 0  
 Celi broj '2' ima vrednost 9  
 Celi broj '3' ima vrednost 10  
 Celi broj '4' je null  
 Celi broj '5' ima vrednost 2  
 Celi broj '6' ima vrednost 99  
 Suma 7 celih brojeva je 125

У примеру се користи ? после декларације, чиме се упућује компајлер да низ може да садржи и null елементе, без тога компајлер би пријавио грешку.

Исказ **case 0** проверава да ли је вредност целобројног елемента низа једнака 0 (што је често случај са иницијланим целобројним променљивима).

И ако то није случај, пада се на следећу проверу **case int vrednost** што у коду врши сабирање било које вредности сем 0.

Исказ **case null** проверава да ли је неки елемент низа једнак null.

#### Задаци за самосталан рад:

1. Написати програм којим се од парних елемената низа niz1 = {0, -2, -20, 3, 10, 6, 1, 0} креира нов низ niz2.
2. Написати програм којим корисник уноси n стрингова у низ. Сваки тај стринг заменити са стрингом ("\*") ако је дужина стринга непаран број. Приказати измењен низ.