

Динамички низ

Пример: Креирати динамички низ чије ће елементе унети корисник. После уноса потребног броја елемената, приказати садржај динамичког низа.

```
using System;
using System.Collections;
namespace Proba
{
    class Program
    {
        static void Main()
        {
            var niz = new ArrayList();
            Console.WriteLine("Uneti element dinamickog niza:");
            string odgovor = "da";
            while (odgovor == "da")
            {
                var x = Console.ReadLine();
                niz.Add(x);
                Console.WriteLine("Da li treba uneti jos elemenata u
niz?");
                odgovor = Console.ReadLine();
            }
            Console.WriteLine("Elementi dinamickog niza su:");
            foreach (var x in niz)
            {
                Console.Write(x + " ");
            }
            Console.WriteLine();
        }
    }
}
```

Пример: У предефинисани динамички низ dniz = {true, 0, 100, false, null, 11.23, "mica"} унети два цела броја. Из низа избацити све булове вредности и null вредности. Приказати преостале елементе низа.

```
using System;
using System.Collections;
namespace Proba
{
    class Program
    {
        static void Main()
        {
            var dniz = new ArrayList() { true, 0, 100, false, null,
11.23, "mica"};
            dniz.Add(1);
        }
    }
}
```

```

        dniz.Add(2);
        for(int i = 0; i < dniz.Count; ++i)
        {
            if ((dniz[i] == null) || (dniz[i].GetType() ==
typeof(bool)))
            {
                dniz.RemoveAt(i);
            }
        }
        foreach (var i in dniz) Console.Write(i + " ");
        Console.WriteLine();
    }
}

```

Пример: У динамичкој листи {null, 1, 3, null} заменити свако појављивање null са низом {0, 0}.

```

using System;
using System.Collections;
namespace Proba
{
    class Program
    {
        static void Main()
        {
            var dniz = new ArrayList() {null, 1, 3, null};
            var niz = new ArrayList() {0, 0};
            for(int i = 0; i < dniz.Count; i++)
            {
                if (dniz[i] == null)
                {
                    dniz.Remove(null);
                    dniz.InsertRange(i, niz);
                }
            }
            foreach (var i in dniz)
            {
                Console.Write(i + " ");
            }
            Console.WriteLine();
        }
    }
}

```

Задаци за самосталан рад:

1. Из предефинисаног динамичког низа {4, 3.56, "miki", -19, "ja", 0} избацити све нумеричке вредности.

```

using System;
using System.Collections;
namespace Proba
{
    class Program
    {
        static void Main()
        {
            var dniz = new ArrayList() {4, 3.56, "miki", -19, "ja",
0};
            Console.WriteLine("Elementi dinamickog niza pre prover
su:");
            foreach (var x in dniz)
            {
                Console.Write(x + " ");
            }
            Console.WriteLine();
            for (int i = 0; i < dniz.Count; ++i)
            {
                if ((dniz[i].GetType() == typeof(double)) ||
(dniz[i].GetType() == typeof(int)))
                {
                    dniz.RemoveAt(i);
                    --i;
                }
            }
            Console.WriteLine("Elementi dinamickog niza posle prover
su:");
            foreach (var x in dniz)
            {
                Console.Write(x + " ");
            }
            Console.WriteLine();
        }
    }
}

```

2. Предефинисан је динамички низ {4, 3.56, "miki", -19, "ja", 0.54}. Креирати статичке низове celi, decimalni, tekstovi у којима ће се наћи елементи типа података који се налазе у динамичком низу.

```
using System;
using System.Collections;
namespace Proba
{
    class Program
    {
        static void Main()
        {
            var dniz = new ArrayList() {4, 3.56, "miki", -19, "ja",
0.54};
            Console.WriteLine("Elementi dinamickog niza su:");
            foreach (var x in dniz)
            {
                Console.Write(x + " ");
            }
            Console.WriteLine();
            int brojCelih = 0, a = 0, brojRealnih = 0, b = 0,
            brojStringova = 0, c = 0;
            for (int i = 0; i < dniz.Count; ++i)
            {
                if (dniz[i].GetType() == typeof(int))
                {
                    brojCelih++;
                }
                else if (dniz[i].GetType() == typeof(string))
                {
                    brojStringova++;
                }
                else
                {
                    brojRealnih++;
                }
            }
            int[] nizCeli = new int[brojCelih];
            double[] nizRealnih = new double[brojRealnih];
            string[] nizStringova = new string[brojStringova];
            for (int i = 0; i < dniz.Count; ++i)
            {
                if (dniz[i].GetType() == typeof(int))
                {
                    nizCeli[a] = (int)(dniz[i]);
                    a++;
                }
            }
        }
    }
}
```

```

        else if (dniz[i].GetType() == typeof(string))
        {
            nizStringova[b] = (string)(dniz[i]);
            b++;
        }
        else
        {
            nizRealnih[c] = (double)(dniz[i]);
            c++;
        }
    }
    Console.WriteLine("Celobrojni niz:");
    foreach (int i in nizCeli)
    {
        Console.Write(i + " ");
    }
    Console.WriteLine();
    Console.WriteLine("Realni niz:");
    foreach (double i in nizRealnih)
    {
        Console.Write(i + " ");
    }
    Console.WriteLine();
    Console.WriteLine("String niz:");
    foreach (string i in nizStringova)
    {
        Console.Write(i + " ");
    }
    Console.WriteLine();
}
}
}

```