

## Nizovi

Niz je uređena kolekcija objekata, gdje je svakom objektu pridružen nenegativan cio broj – indeks elementa niza

Primjeri:

1. niz parnih prirodnih brojeva manjih od 15: 2,4,6,8,10,12,14
2. niz prostih brojeva manjih od 20: 2,3,5,7,11,13,17,19
3. niz cifara u decimalnom razvoju razlomka 1/7: 0,1,4,2,8,5,7,1,4,2,8,5,7,1,4...

Ako niz nazovemo  $x$ , tada se element čiji je indeks  $i$  označavamo sa  $x_i$ . Npr. u gornjem primjeru je  $x_1=2$  a  $x_6=12$ . Niz ima 7 elemenata.

Indeks	1	2	3	4	5	6	7
Element	2	4	6	8	10	12	14

Ponekad je moguće napisati vezu između indeksa niza i elementa. Npr. u gornjem primjeru je to moguće:  $x_i=2i$ ,  $i=1,2,3,4,5,6,7$

Primjer: Neka je  $x_i=2^{i+1}-3$ ,  $i=1,2,3,4,5$ .

Indeks	1	2	3	4	5
Elementi	1	5	13	29	61

$x_i=2^{i+1}-3$  je opšti član niza,  $i=1,2,3,4,5$ .

Uobičajeno je da indeksi niza počinju od 1. Međutim, mogu počinjati i od nula

- $x_n=1+n^2$ ,  $n=0,1,2...$  predstavlja beskonačni niz 1, 2, 5, 10...
- Odredite  $x_{20}$

## Nizovi u programiranju

Nizovi sadrže samo elemente jednog tipa (npr. cijele brojeve, realne brojeve, karaktere, nizove karaktera, stringove, automobile, predmete, ocjene...). Elementi niza obično se označavaju srednjim (uglastim) zagradama. Npr. petnaesti element niza  $x$  (tj.  $x_{15}$ ) označava se sa  $x[15]$ . Uzastopni elementi niza zauzimaju uzastopne memorijske lokacije.

Primjer: Niz cijelih brojeva u Pascal-u, jedan cio broj zauzima 4 bajta

Indeks	1	2	3	4	5
Element	-22	12	123	0	-13
Adresa	3000	3004	3008	3012	3016

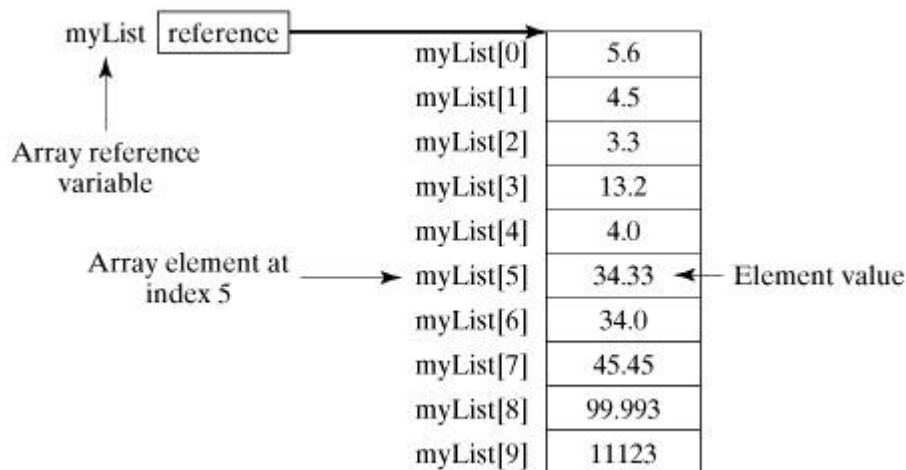
Primjer: niz realnih brojeva, jedan realan broj zauzima 8 bajtova

a	a[0]	1200
	a[1]	1208
	a[2]	1216
	a[3]	1224
	a[4]	1232
	a[5]	1240
	a[6]	1248
	a[7]	1256
Niz	Adrese	

Niz u jeziku Java je tzv. kontejnerski objekat koji može da čuva konačan fiksiran broj vrijednosti jednog tipa. Dužina niza (tj . broj elemenata niza) se određuje pri njegovom kreiranju i poslije toga je fiksirana. Elementi niza mogu biti proizvoljnog tipa. Dužini niza pristupamo pomoću promjenljive `length`. (npr. ako je niz `myList`, tada je broj elemenata `myList.length`). Ime niza je referenca na blok memorije koji sadrži elemente niza.

Primjer: niz u jeziku Java. Obratite pažnju da indeksi počinju od 0.

```
double [] myList = new int[10];
myList[0] = 5.6;
myList[1] = 4.5;
myList[2] = 3.3;
myList[3] = 13.2;
myList[4] = 4;
myList[5] = 0.33+3*10+4.0;
myList[6] = 34.0;
myList[7] = 45.45;
myList[8] = 99.993;
myList[9] = 11123;
```



### Primjeri deklarisanja niza (Java)

```
int [] xa; // niz cijelih brojeva
xa = new int[5]; // xa sadrzi 5 cijelih brojeva, svi su postavljeni na 0
// Napravimo niz ya koji sadrzi 10 realnih brojeva
double [] ya = new double [10]; // svi elementi su postavljeni na 0.0
int c[] = new int[ 12 ];
String b[] = new String[ 100 ];
```

### Načini deklarisanja i alokacija memorije

- Deklaracija i alokacija (rezervisanja) memorije odvojeno  
`int[] students;`  
`students = new int[100];`
- Jedna naredba  
`int[] students = new int[100];`
- Inicijalizacija niza vrijednostima  
`int[] gradStudents = {12,143, 123, 89, 32, 88};`
- Upotreba postojećeg niza  
`int[] gradStudents = new int[100];`  
`int[] students = gradStudents ;`

Kada je memorija alocirana, podrazumijevane vrijednosti za elemente niza su **0** za brojeve, **false** za tip boolean i **null** za reference.

Primjer:

```
int[] students = new int[5];
System.out.println(students[3]);
```

0

```
String[] students = new String[2];
System.out.println(students[1]);
```

null

Škola programiranja, 12.12.2015.

## Dužina niza (java)

Koristi se **length** za određivanje dužine niza

Primjer:

```
String[] staff = {"Red", "Sean", "Patrick", "Orr"};
int[] numbers = {4,2,1};
```

```
System.out.println(staff.length): // stampa se 4
System.out.println(numbers.length): // stampa se 3
```

## Pristup elementima niza

Elementu niza pristupa se navođenjem imena niza i odgovarajućeg indeksa

Primjer:

```
int[] a = {3, 5, 7, 9};
```

```
System.out.println(a[2]): // stampa se 7
System.out.println(a[a.length - 1]): // stampa se 9
System.out.println(a[15]): // java.lang.ArrayIndexOutOfBoundsException
```

Škola programiranja, 12.12.2015.

Primjeri upotrebe nizova:

**Primjer 1:**

```
public class Nizovi {

    public static void main(String[] args) {

        int [] ocjene = new int[7];

        // svi elementi niza su nule
        System.out.println("Stampa niza:");
        System.out.println(ocjene[0]);
        System.out.println(ocjene[1]);
        System.out.println(ocjene[2]);
        System.out.println(ocjene[3]);
        System.out.println(ocjene[4]);
        System.out.println(ocjene[5]);
        System.out.println(ocjene[6]);
        System.out.println();

        // dodjeljivanje vrijednosti elementima niza
        ocjene[0] = 5;
        ocjene[1] = 5;
        ocjene[2] = 4;
        ocjene[3] = 3;
        ocjene[4] = (ocjene[0]+ocjene[2])/2;

        // stampa pomocu ciklusa
        int ind = 0;
        while(ind < ocjene.length)
        {
            System.out.println(ocjene[ind]);
            ind = ind + 1; // ind++;
        }
        System.out.println("Kraj stampe niza ocjene");
        System.out.println();

        int n = 7;
        double [] b = new double[n];
        // stampa pomocu ciklusa, svi elementi u jednom redu
        ind = 0;
        while(ind < b.length)
        {
            System.out.print(b[ind] + " ");
            ind = ind + 1;
        }
        System.out.println();
        System.out.println("Kraj stampe niza b");
    }

}
```

## Primjer 2

```
import java.util.Scanner;

public class UcitavanjeNiza {

    public static void main(String[] args) {

        /* Niz se zadaje brojem elemenata - promjenljiva n.
        * U sljedećih n redova je po jedan broj - elementi niza a.
        * Stampati unesene brojeve.
        */

        Scanner in= new Scanner(System.in);
        int n = in.nextInt(); // učitavamo broj elemenata

        int [] a = new int[n]; // kreiramo niz
        int ind = 0;
        while (ind < a.length)
        {
            a[ind] = in.nextInt(); // učitavamo element sa indeksom ind
            ind = ind + 1; // prelazimo na sljedeći element
        }

        // stampanje niza
        ind = 0;
        while (ind < a.length)
        {
            System.out.println(a[ind]); // stampamo element
            ind = ind + 1; // prelazimo na sljedeći element
        }
    }
}
```

## Primjer 3

## Škola programiranja, 12.12.2015.

```
import java.util.Scanner;

public class MaksimumNiza {

    public static void main(String[] args) {
        /* Niz se zadaje brojem elemenata - promjenljiva n.
        * U sljedecih n redova je po jedan broj - elementi niza a.
        * Stampati najveći element niza.
        */

        Scanner in= new Scanner(System.in);
        int n = in.nextInt(); // učitavamo broj elemenata

        int [] a = new int[n]; // kreiramo niz
        int ind = 0;
        while (ind < a.length)
        {
            a[ind] = in.nextInt(); // učitavamo element sa indeksom ind
            ind = ind + 1; // prelazimo na sljedeći element
        }

        // najveći element
        ind = 0;
        int maks = a[0];
        while (ind < a.length)
        {
            if (a[ind] > maks)
            {
                maks = a[ind];
            }
            ind = ind + 1; // prelazimo na sljedeći element
        }
        System.out.println(maks);
    }
}
```

**Primjer 4:**

```
import java.util.Scanner;

public class Zbir {

    public static void main(String[] args) {
        /* Niz se zadaje brojem elemenata - promjenljiva n.
        * U sljedećih n redova je po jedan broj - elementi niza a.
        * Stampati zbir elemenata niza.
        */

        Scanner in= new Scanner(System.in);
        int n = in.nextInt(); // učitavamo broj elemenata

        int [] a = new int[n]; // kreiramo niz
        int ind = 0;
        while (ind < a.length)
        {
            a[ind] = in.nextInt(); // učitavamo element sa indeksom ind
            ind = ind + 1; // prelazimo na sljedeći element
        }

        // zbir elemenata
        ind = 0;
        int zbir = 0;
        while (ind < a.length)
        {
            zbir = zbir + a[ind]; // dodajemo element na poziciji ind
            ind = ind + 1; // prelazimo na sljedeći element
        }
        System.out.println(zbir);

    }

}
```