

# Nizovi

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

Ako niz nazovemo  $x$ , tada se element čiji je indeks  $i$  označavamo sa  $x_i$  (matematički) ili  $x[i]$  (u C++). Npr. u gornjem primjeru je  $x_0=2$  ili  $x[0]=2$ , a  $x_5=12$  ( $x[5]=12$ ). Niz ima 7 elemenata, a u C++ indeksi uvijek počinju od 0.

Indeks	0	1	2	3	4	5	6
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:  $x[i]=2^{i+1}$ ,  $i=0,1,2,3,4,5,6$

Primjer: Neka je  $x_i=2^{i+1}-3$ ,  $i=0,1,2,3,4$ .

Indeks	0	1	2	3	4
Elementi	1	5	13	29	61

$x_i=2^{i+1}-3$  je opšti član niza,  $i=0,1,2,3,4$ .

## Nizovi u programiranju

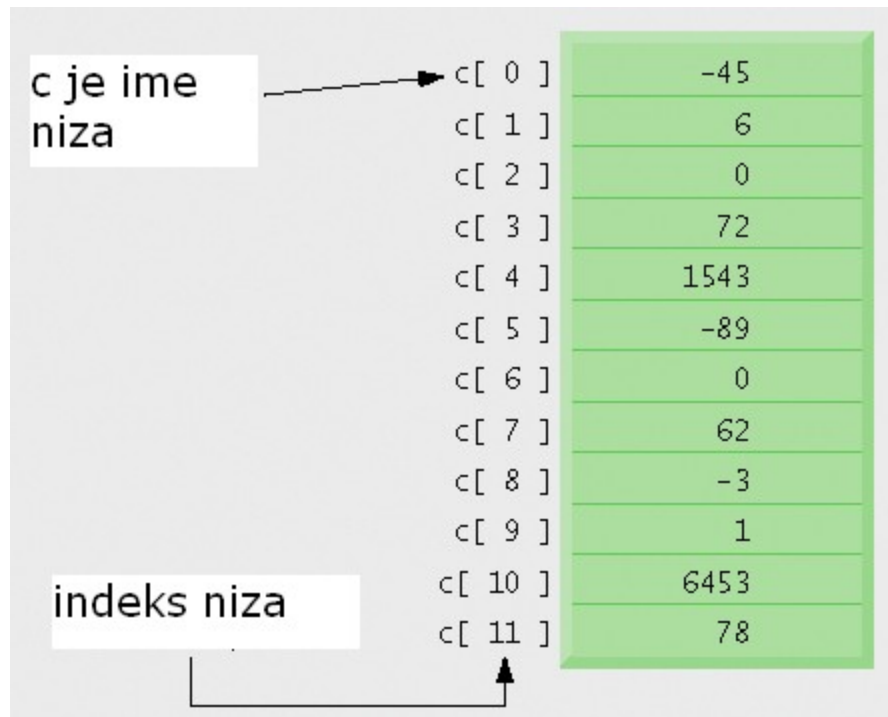
Nizovi sadrže samo elemente jednog tipa (npr. cijele brojeve, realne brojeve, karaktere, nizove karaktera, stringove, automobile, predmete, ocjene...). Elementi niza obično se označavaju srednjim (uglastim) zagradama. Uzastopni elementi niza zauzimaju uzastopne memorijske lokacije.

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

Primjer: niz u jezicima C, C++, Java. Obratite pažnju da indeksi počinju od 0.



## Deklarisanje niza

```
int c[12];      // 12 cijelih brojeva
double x[6];   // 6 realnih brojeva
char dd[100];  // 100 karaktera
```

## Pristup elementima niza

Elementu niza pristupa se navođenjem imena niza i odgovarajućeg indeksa

Primjer:

```
int a[4], ind = 3;
a[0] = 3;
a[1] = 5;
a[2] = 7;
a[3] = -4;
```

```
cout << a[2]<< endl; // stampa se 7
cout << a[ind] << endl; // stampa se -4
cout << a[15] << endl; // nije jasno sto se stampa
a[3] = a[0] + 2 * a[1];
cout << a[3] << endl; // stampa se 13
```

**Zadatak 1** – Unosimo 6 ocjena, štampamo ih i odredimo prosječnu ocjenu.

```
#include <iostream>
#include <iomanip>

using namespace std;

int main()
{
    int ocjene[6]; // 6 cijelih brojeva
    ocjene[3] = 4;
    ocjene[0] = 5;
    ocjene[1] = (ocjene[3] + ocjene[0])/2;
    ocjene[3] = ocjene[3] - 1; // ocjene[3]--; ili ocjene[3] -= 1;
    ocjene[2] = ocjene[4] = 5;
    ocjene[5] = 4;

    // stampanje niza
    int ind = 0;
    while(ind<6)
    {
        cout << ocjene[ind] << endl;
        ind++;
    }

    // ucitavanje elemenata sa tastature
    ind = 0;
    while(ind < 6)
    {
        cout << "Unesite ocjenu:";
        cin >> ocjene[ind];
        ind++;
    }

    // prosjecna ocjena
    ind = 0;
    int suma = 0;
    while(ind < 6)
    {
        suma = suma + ocjene[ind];
        ind++;
    }
    float prosjek = suma/6.0;
    // podesavanje da stampa samo 2 decimale
    cout << fixed << setprecision(2);
    cout << "prosjek je " << prosjek << endl;
    return 0;
}
```

**Zadatak 2** – Napisati program koji učitava prirodan broj  $n$  ( $1 < n < 100$ ) i niz od  $n$  realnih brojeva i štampa koliko ima parova uzastopnih elemenata niza koji se razlikuju najviše za 1.5.

**Prvo rješenje:**

```
#include <iostream>
#include <iomanip>
#include <iostream>
```

```

using namespace std;

int main()
{
    int n; // broj elemenata niza
    int ind = 0;
    float d[100]; // rezervisemo 100 elemenata
    // učitavanje niza
    cin >> n; // stvarni broj elemenata u nizu, mora biti manji od 100
    while(ind < n)
    {
        cin >> d[ind];
        ind++;
    }
    int brojParova = 0;
    ind = 0;
    while (ind < n-1) // petlja ide do n-1 jer uporedjujemo d[ind] i d[ind+1]
    {
        // ako je d[ind] i d[ind+1] razlikuju najviše za 1.5
        // povećaj brojParova za 1
        if (abs(d[ind]-d[ind+1])<=1.5) {
            brojParova++;
        }
        ind++;
    }
    cout << brojParova << endl;
    return 0;
}

```

## Drugo rješenje:

```

#include <iostream>
#include <iomanip>
#include <iostream>

using namespace std;

int main()
{
    int n; // broj elemenata niza
    int ind = 0;

    // učitavanje niza
    cin >> n; // broj elemenata u nizu
    float d[n];
    while(ind < n)
    {
        cin >> d[ind];
        ind++;
    }
    int brojParova = 0;
    ind = 0;
    while (ind < n-1) // petlja ide do n-1 jer uporedjujemo d[ind] i d[ind+1]
    {
        // ako je d[ind] i d[ind+1] razlikuju najviše za 1.5
        // povećaj brojParova za 1
        if (abs(d[ind]-d[ind+1])<=1.5) {
            brojParova++;
        }
    }
}

```

```
    }  
    ind++;  
}  
cout << brojParova << endl;  
return 0;  
}
```