

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 1. važi 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: $x_i=2^i$, $i=1,2,3,4,5,6,7$

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. element niza x sa indeksom 15 (tj. x_{15}) označava se sa $x[15]$. Uzastopni elementi niza zauzimaju uzastopne memorijske lokacije. Primjer: Niz cijelih brojeva u jeziku C++, jedan cio broj zauzima 4 bajta

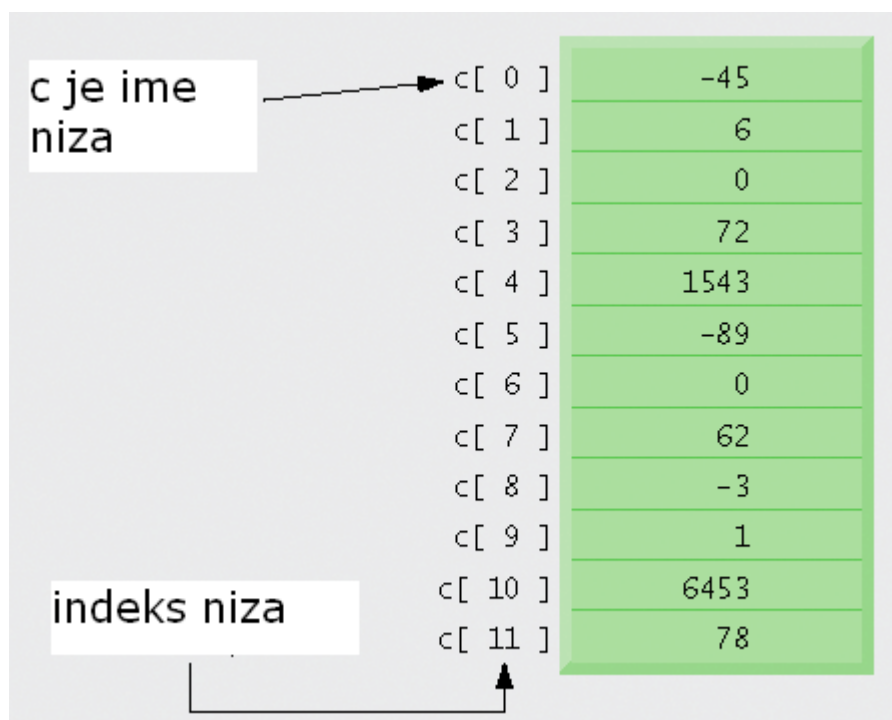
| | | | | | |
|---------|------|------|------|------|------|
| Indeks | 0 | 1 | 3 | 4 | 5 |
| Element | -22 | 12 | 123 | 0 | -13 |
| Adresa | 3000 | 3004 | 3008 | 3012 | 3016 |

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

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



Deklarisanje niza

Postoji više načina da se u jeziku C/C++ deklariraju niz:

- Prvi način: statički niz

```
int c[12]; // niz od 12 cijelih brojeva
char a[5]; // niz od 5 karaktera
double b[22]; // niz od 22 realna broja tipa double
```
- Drugi način: statički niz sa inicijalizacijom

```
int c[5] = {-22, 12, 123, 0, -13}; // niz od 5 cijelih brojeva
```

Na slici je prikazana memorija za niz c:

| | | | | | |
|--------------|-----|----|-----|---|-----|
| Indeks | 0 | 1 | 3 | 4 | 5 |
| Element c[i] | -22 | 12 | 123 | 0 | -13 |

```
char a[6] = {'a', 'b', '+', '1', '#', 'A'};
```

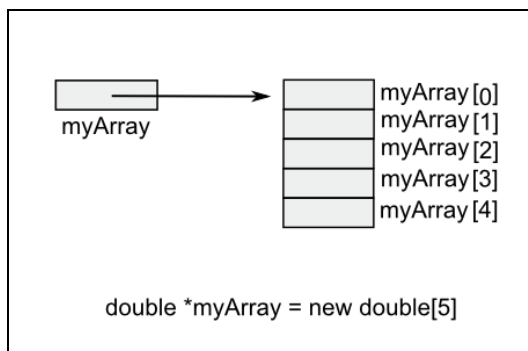
Na slici je prikazana memorija za niz a:

| | | | | | | |
|--------------|-----|-----|-----|-----|-----|-----|
| Indeks | 0 | 1 | 2 | 3 | 4 | 5 |
| Element a[i] | 'a' | 'b' | '+' | '1' | '#' | 'A' |

- Treći način: dinamički niz

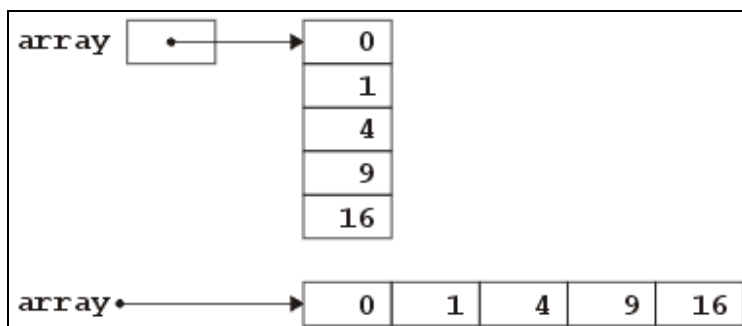
```
double *myArray = new double[5];
```

Na slici je prikazana memorija za niz myArray:



```
int *array = new int[5]; // niz od 5 cijelih brojeva
int i = 0;
while (i<5) {array[i] = i*i; i = i+1;}
```

Na slici je prikazana memorija za niz array:



```
char *z = new char[5]; // niz od 5 karaktera
```

Zadaci sa časa

```
#include <iostream>

using namespace std;
/*
    Program koji ucitava prirodan broj n manji od 2000000000 i
    stampa zbir njegovih cifara
*/
int main()
{
    int n, zbir= 0;
    cin >> n;
    // posljednja cifra broja n je n%10
    // tu cifru dodajemo na zbir i zatim je otkinemo.
    // Za otkidanje koristimo n=n/10.
    // Ponavljamo dok ima cifara.
    while(n>0)
    {
        zbir = zbir + n%10;
        n = n/10;
    }
    cout<< zbir;
    return 0;
}
```

```
#include <iostream>

using namespace std;
/*
    Program koji ucitava prirodan broj n manji od 2000000000 i
    stampa njegovu najveću cifru
*/
int main()
{
    int n, najveca = 0;
    cin >> n;
    while(n>0)
    {
        if (n%10 > najveca)
        {
            najveca = n%10;
        }
        n = n/10;
    }
    cout<< najveca << endl;
    return 0;
}
```

```

#include <iostream>

using namespace std;
/*
    Prvi program sa nizovima.
*/
int main()
{
    int x[6]; // niz od 6 cijelih brojeva,
    // Elementi niza zauzimaju uzatsopne memorijske lokacije
    // indeksi niza pocinju od 0.
    //Najveci indeks je n-1, gdje je n broj elemenata niza

    // dodjeljivanje vrijednosti elementima niza
    x[0] = 20;
    x[1] = x[0] + 10;
    x[2] = x[0]+2*x[1];
    x[3] = 25-x[1];
    x[4] = 10+x[2]/2;;
    x[5] = 22-x[3];

    // stampanje elemenata niza
    cout << x[0] << " " << x[1] << " " << x[2] << " " << x[3] << " " <<
x[4] << " " << x[5] << endl;

    // stampanje niza pomocu petlje
    int i = 0; // indkes prvog elementa niza
    int n = 6; // broj lemenata niza
    while(i<n)
    {
        cout << x[i] << " "; // stampamo i-ti element
        i = i + 1;           // prelazimo na sljedeci
    }

    return 0;
}

```

```
#include <iostream>

using namespace std;
/*
    Napisati program koji kreira i stampa sljedeci niz od 100 brojeva:
    1,4,7,10,...
*/
int main()
{
    // Primjetite da je svaki element niza za 3 veci od prethodnog elementa
    int x[100];

    x[0] = 1;
    int i = 1, n = 100;
    while (i<n)
    {
        x[i] = x[i-1] + 3; // i-ti element (tj. x[i]) je za 3 veci od
prethodnog (tj. x[i-1]
        i = i + 1;
    }

    // Stapanje niza, moglo je i da se odradi u prethodnoj petlji
    i = 0;
    while(i<n)
    {
        cout << x[i] << " ";
        i = i + 1;
    }

    return 0;
}
```

```

#include <iostream>

using namespace std;

/*
Napisati program koji kreira i stampa Fibonacijev niz od n brojeva :
1,1,2,3,5,8,...
Svaki element, osim prva dva, je zbir prethodna dva elementa.
*/

int main()
{
    int n; // broj elemenata niza'
    cin >> n;

    int * x = new int [n]; // dinamicki niz sa n elemenata

    x[0] = 1;
    if (n>1)
    {
        x[1] = 1;
    }

    int i = 2;
    while (i<n)
    {
        x[i] = x[i-1] + x[i-2]; // i-ti element je zbir prethodna dva
elementa
        i = i + 1;
    }

    // Stapanje niza, moglo je i da se odradi u prethodnoj petlji
    i = 0;
    while(i<n)
    {
        cout << x[i] << " ";
        i = i + 1;
    }

    return 0;
}

```

```

#include <iostream>

using namespace std;
/*
Napisati program koji ucitava broj n, unosi n elemenata niza x i unosi broj
el
i stampa indeks prvog pojavljivanja broja el u nizu x,
ili stampa -1 ako el nije u nizu x.
Npr. neka je niz [1,2,3,4,5,6,7,8,4,5,6].
Ako je el = 4, program stampa 3. Ako je el=10, program stampa -1.
*/

int main()
{
    int n; // broj elemenata niza
    cout << "Unesite broj elemenata niza:" << endl;
    cin >> n;

    int * x = new int [n]; // dinamicki niz sa n elemenata

    cout << "Unesite elemente niza:" << endl;
    int i = 0;
    while (i<n)
    {
        cin >> x[i];
        i = i + 1;
    }

    int el = 0;
    cout << "Trazi se:" << endl;
    cin >> el;

    // Trazenje broja el u nizu x
    i = 0;
    while(i<n)
    {
        if (el == x[i])
        {
            break;
        }
        i = i + 1;
    }
    if (i<n)
    {
        cout << i<< endl;
    }
    else
    {
        cout << -1 << endl;
    }
    return 0;
}

```