

# Polja

*polja, znakovni nizovi i dvodimenzionalna polja*

Dejan Ljubobratović, mag. educ. math. et inf.

dejan.ljubobratovic@uniri.hr

Fakultet informatike i digitalnih tehnologija, Sveučilište u Rijeci



Sveučilište u Rijeci  
**Fakultet informatike  
i digitalnih tehnologija**

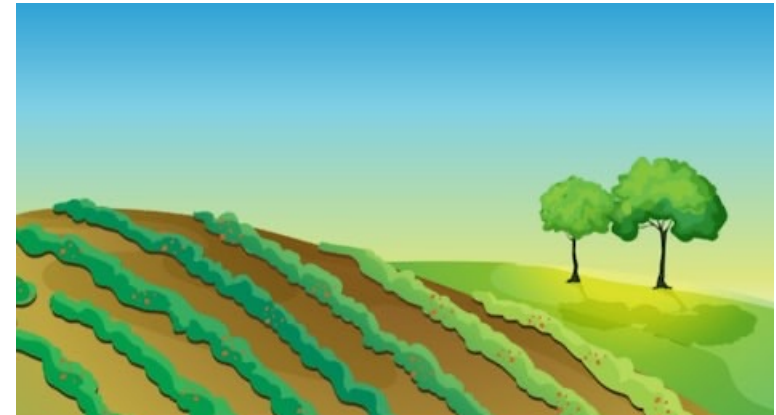


# Polja

$x[]$

# Jednodimenzionalno polje

Kod jednodimenzionalnih polja članovi polja (podaci) su poredani u niz (jedan iza drugoga), a indeks svakog od članova odgovara njegovoj udaljenosti od prvog člana.



## Jednodimenzionalno polje

$x[0]$	$x[1]$	$x[2]$	$\dots$	$\dots$	$x[N-1]$
--------	--------	--------	---------	---------	----------

Svaki od članova niza se označava cjelobrojnim indeksom  $i$  to tako da **prvi član** niza ima **indeks 0**, a **posljednji član** indeks **za jedan manji od duljine polja** (u gornjem primjeru duljina polja je  $N$ ).

# Način zadavanja indeksa članova niza

Kod izdvajanja (dohvaćanja) članova niza, indeks člana niza može biti zadan kao:

- cjelobrojna konstanta
- cjelobrojna varijabla
- cjelobrojni izraz



## Način zadavanja indeksa članova niza

$x[3]$	cjelobrojna konstanta
$x[a]$	cjelobrojna varijabla
$x[a*3-1]$	cjelobrojni izraz

- Polje treba deklarirati kako bi se u memoriji osigurao potreban prostor za podatke u polju.
- Svi članovi polja moraju biti istog tipa

## Primjeri deklaracije polja

```
tip_clanova_niza ime_niza[duljina_niza];
```

Opći oblik deklaracije jednodimenzionalnog polja

```
int x[6];
```

Polje X je jednodimenzionalno polje koje ima šest članova koji su po tipu cijeli brojevi.

```
int a[100];  
float x[20];  
char tekst[50];
```

Članovi polja mogu biti bilo kojeg tipa (int, float ili bilo kojeg drugog).

```
int A[6];  
int A;
```

Kao što se pri deklaraciji ne smiju ponavljati ista imena varijabli, tako se ni ime polja ne smije podudarati s imenom drugog polja ili neke od varijabli. Primjerice, ovakva je deklaracija pogrešna.

- Članovima deklariranog polja mogu se pridružiti vrijednosti
- Vrijednosti se navode unutar para vitičastih zagrada, odvojene zarezom

## Pridruživanje vrijednosti članovima polja

```
int X[6]={2,34,1,67,99,7}
```

Polje X ima 6 članova i svi su po tipu cijeli brojevi.

```
int X[ ]={2,34,1,67,99,7}
```

Vrijedit će i ako se ne navede duljina polja: (prevoditelj sam "zaključuje" da je u listi šest članova i nakon toga se duljina polja više ne može mijenjati).

Član polja	X[0]	X[1]	X[2]	X[3]	X[4]	X[5]
Vrijednost	2	34	1	67	99	7

# Zadatak 1

Treba unijeti članove jednodimenzionalnog polja

```
int A[6]= {7, 16, 2, 13, 93, 3};
```

pa ih ispisati.

```
#include <iostream>
using namespace std;
int main ()
{
    int i;
    int A[6]= {7, 16, 2, 13, 93, 3};
    for(i=0; i<=5; i++)
        cout<<"A["<<i<<"]="<<A[i]<<"    ";
    cout<<endl;
    return 0;
}
```

A[0]=7 A[1]=16 A[2]=2 A[3]=13 A[4]=93 A[5]=3  
Process returned 0 (0x0)  
Press any key to continue. execution time : 0.420 s





Sveučilište u Rijeci  
**Fakultet informatike  
i digitalnih tehnologija**

uniri



# Znakovni niz

*char x[]*

- Ako je podatak znak, njegova oznaka tipa je char.
- Podatak tipa char je predstavljen jednim znakom unutar jednostrukih navodnika ili ASCII vrijednošću tog znaka (dekadska protuvrijednost)

## Primjer pridruživanja vrijednosti varijablama tipa char

```
char slovo1 = 'A';
```

Pridružen je znak označen jednostrukim navodnicima

```
char slovo2 = 65;
```

Pridružen je dekadski ekvivalent ASCII znaka

- Ako je podatak tekst (niz znakova) rabe se **znakovni nizovi** (engl. **character strings**)
- Znakovni niz je **jednodimenzionalno polje** čiji su članovi znakovi
- Sadržaj se u znakovni niz unosi između para dvostrukih navodnika  
`char X[] = "Znakovni niz"`  
a u memoriju se sprema:

'Z'	'n'	'a'	'k'	'o'	'v'	'n'	'i'	' '	'n'	'i'	'z'	'\0'
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	------

- Svaki znakovni niz završava zaključnim znakom `\0` (engl. null character). Taj znak ne treba posebno navoditi, ali treba znati da on zauzima jedno mjesto u memoriji.

# Zadatak 2

Treba prebrojiti koliko puta se pojavljuje slovo 'o' u znakovnom nizu:  
"Koliko slova o ima u ovoj rečenici?"

```
#include <iostream>
using namespace std;
int main ()
{
    int i, brojo;
    brojo=0;
    char X[]="Koliko slova o ima u ovoj recenici?";
    for (i=0; X[i]!='\0'; i++)
        if(X[i]=='o')
            brojo=brojo+1;
    cout<<endl<<"U znakovnom nizu: 'Koliko slova o ima ";
    cout<<"u ovoj recenici?' Ima "<<brojo<<" slova o.\n";
    return 0;
}
```

U znakovnom nizu: 'Koliko slova o ima u ovoj recenici?' Ima 6 slova o.  
Process returned 0 (0x0) execution time : 0.192 s  
Press any key to continue.



## dvodimenzionalno polje

$x[][]$

# 2 dimenzionalno polje

- Članovi dvodimenzionalnog polja mogu se izdvojiti (dohvatiti) pomoću dva indeksa
- Zamislimo li polje podataka kao tablicu onda prvi indeks određuje redak, a drugi stupac
- Prvi član polja ima indeks [0][0] a posljednji indeks [(broj redaka-1)][(broj stupaca-1)].

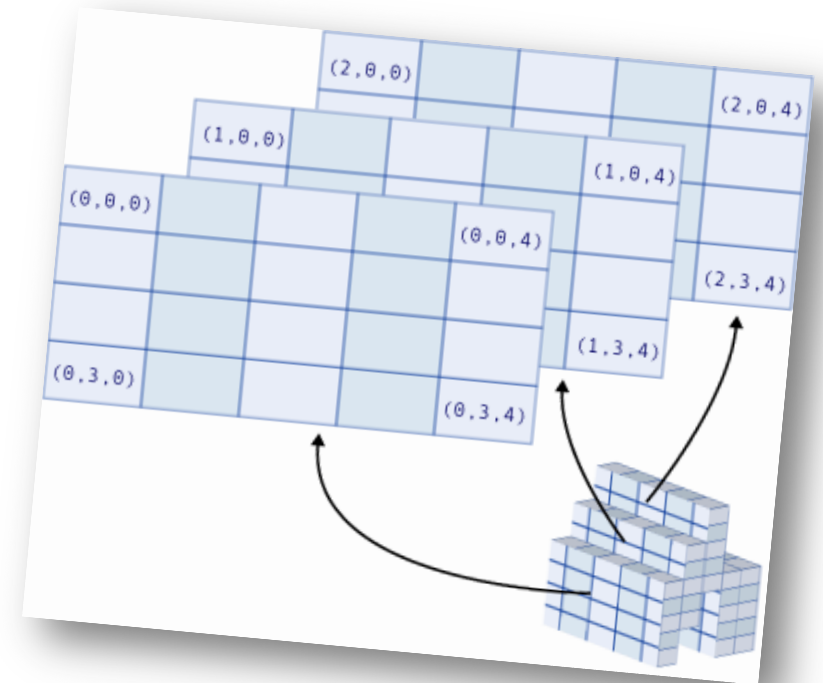
Za X[2][3] indksi su:		
X[0][0]	X[0][1]	X[0][2]
X[1][0]	X[1][1]	X[1][2]

# Deklaracija 2 dimenzionalnog polja

## Deklaracija dvodimenzionalnog polja:

```
int x[2][3];
```

X je dvodimenzionalno polje koje se može prikazati kao tablica s dva retka i tri stupca u kojima su pohranjeni podaci po tipu cijeli brojevi.



# Pridruživanje vrijednosti

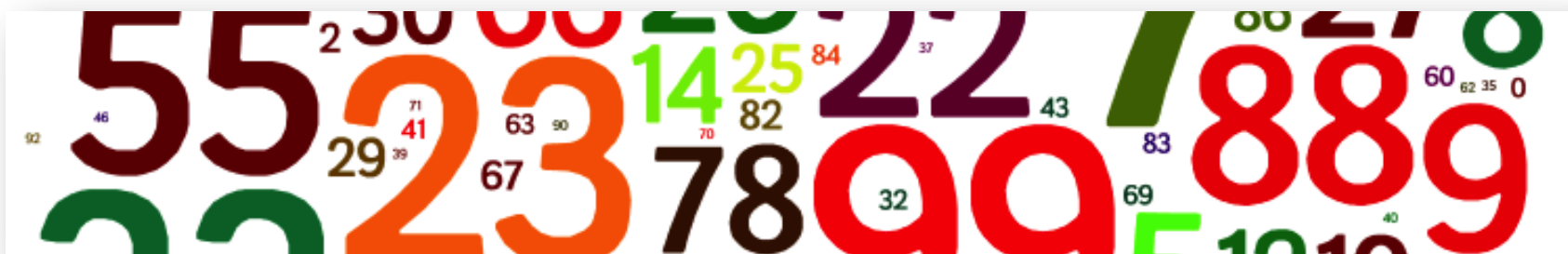
## Pridruživanje vrijednosti članovima polja

```
int X[2][3]={ {..., ..., ...}, {..., ..., ...}};
```

Vrijednosti članova pojedinih redaka omeđuju se parom otvorena-zatvorena vitičasta zagrada

```
int X[2][3]={ {1, 2, 3}, {4, 5, 6}};
```

Članovima polja mogu se pridružiti vrijednosti i prilikom deklaracije.



# Zadatak 3

Treba unijeti članove dvodimenzionalnog polja pa ih ispisati u obliku tablice.

```
int X[2][3]={{1,2,3}, {4,5,6}};
```

```
x[0][0]= 1  x[0][1]= 2  x[0][2]= 3
x[1][0]= 4  x[1][1]= 5  x[1][2]= 6
```

```
#include <iostream>
using namespace std;
int main ()
{
    int i,j;
    int X[2][3]= {{1,2,3}, {4,5,6}};
    for(i=0; i<2; i++)
    {
        for(j=0; j<3; j++)
            cout<<"X["<<i<<"]["<<j<<"]="<<X[i][j]<<" ";
        cout<<endl;
    }
    return 0;
}
```