

Polja

© S. Šutalo i D. Grundler, 2009.

Skupovi podataka istog tipa

- U programima ponekad treba rukovati s više podataka koji čine jednu cjelinu, a istog su tipa.
- Ako bi se za svaki podatak deklarirala zasebna varijabla, rukovanje tim podacima bilo bi vrlo nespretno, osobito kada nad takvim podacima treba provoditi iste naredbe.

(c) S. Šutalo i D. Grundler, 2009.

2

Skupovi podataka istog tipa

- Skupove podataka istog tipa može se pohraniti pod zajedničkim imenom, a redni broj podatka u skupu označiti brojem (indeksom).
- To je moguće pomoću strukture podataka koja se naziva polje (niz, engl. *array*).

(c) S. Šutalo i D. Grundler, 2009.

3

Polje podataka

- Polje je konačni skup podataka istog tipa koji čine cjelinu.
- Sastoji se od članova polja (podataka).
- Položaj člana unutar polja označen je cjelobrojnim brojevanim indeksom (jednim ili s više, ovisno o vrsti polja).

(c) S. Šutalo i D. Grundler, 2009.

4

Jednodimenzionalno polje

- Polja mogu biti jedno ili više dimenzionalna.
- Najjednostavnija su jednodimenzionalna polja kod kojih su članovi (podaci) poredani u niz (jedan iza drugog).

(c) S. Šutalo i D. Grundler, 2009.

5

Jednodimenzionalno polje

- Svaki od članova niza označava se cjelobrojnim indeksom koji odgovara njegovoj udaljenosti od prvog člana i to tako da:
 - prvi član niza ima indeks 0,
 - posljednji član ima indeks za jedan manji od duljine polja.

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]
------	------	------	------	------	------

(c) S. Šutalo i D. Grundler, 2009.

6

Deklaracija polja

- Deklaracija jednodimenzionalnog polja je oblika:

tip_polja naziv_polja [dimenzija_polja]

↓

tip podataka koji se
pohranjuju u polje

↓

broj članova polja

(c) S.Šutalo i D.Grundler, 2009.

7

Deklaracija polja

Primjer:

- Jednodimenzionalno polje se deklarira, npr.:

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

- A je jednodimenzionalno polje od 6 članova koji su po tipu cjelobrojne vrijednosti.

(c) S.Šutalo i D.Grundler, 2009.

8

Deklaracija polja

- Duljina deklariranog polja se ne može mijenjati tijekom izvođenja programa.
- Podaci, članovi polja mogu biti bilo kojeg tipa (int, float i dr.), npr.:

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

(c) S.Šutalo i D.Grundler, 2009.

9

Deklaracija polja

- Pri deklaraciji se ime polja ne smije podudarati sa imenom neke od varijabli.

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

neispravna deklaracija

- U slučaju neispravne deklaracije prevoditelj će javiti pogrešku:

```
cpp(9) : error C2040: 'A' : 'int' differs in levels of indirection from 'int [6]'

1 error(s), 0 warning(s)
```

(c) S.Šutalo i D.Grundler, 2009.

10

Pridruživanje vrijednosti

- Članovima polja vrijednosti se pridružuju tako da se navedu nakon operatora pridruživanja unutar para vitičastih zagrada, odvojene zarezom.

(c) S.Šutalo i D.Grundler, 2009.

11

Pridruživanje vrijednosti

Primjer:

- Prethodno deklariranom jednodimenzionalnom polju A od 6 članova treba pridružiti sljedeće vrijednosti:

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]
2	34	1	67	99	7

(c) S.Šutalo i D.Grundler, 2009.

12

Pridruživanje vrijednosti

Primjer:

- Članovima polja se vrijednosti pridružuju:

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

- ili bez navedene duljine polja:

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

(prevoditelj sam "zaključuje" da polje ima šest članova i nakon toga se duljina polja više ne može mijenjati).

(c) S.Šutalo i D.Grundler, 2009.

13

Pridruživanje vrijednosti

- Članovi polja razlikuju po indeksu koji se mijenja od 0 do n-1 (ako je n ukupan broj članova niza)
- Za upis ili ispis članova polja pogodno je rabiti petlju.

(c) S.Šutalo i D.Grundler, 2009.

14

Primjer 89



- Treba unijeti članove jednodimenzionalnog polja iz prethodnog primjera, pa ih ispisati.
- Ispis treba izgledati ovako:

```
A[0]=2 A[1]=34 A[2]=1 A[3]=67 A[4]=99 A[5]=7
```

(c) S.Šutalo i D.Grundler, 2009.

15

Primjer 89

```
#include <iostream>
using namespace std;
int main ()
{
    int i;
    int A[6]={2,34,1,67,99,7};
    for(i=0;i<6;i++)
    {
        cout<<"A["<<i<<"]="<<A[i]<<" ";
    }
    cout<<endl;
    return 0;
}
```

deklaracija polja A
uz pridruživanje
vrijednosti
članovima polja

ispis vrijednosti
i-tog člana polja

(c) S.Šutalo i D.Grundler, 2009.

16

```
#include <iostream>
using namespace std;
int main ()
{
    int i;
    int A[6]={2, 34, 1, 67, 99, 7};
    for(i=0;i<6;i++)
    {
        cout<<"A["<<i<<"]="<<A[i]<<" ";
    }
    cout<<endl;
    return 0;
}
```

Primjer 89

- Provjera:

```
G:\
A[0]=2 A[1]=34 A[2]=1 A[3]=67 A[4]=99 A[5]=7
```

(c) S.Šutalo i D.Grundler, 2009.

18

Duljina polja (pri deklaraciji)

- Duljina polja koja se navodi pri deklaraciji mora biti veća ili jednaka broju članova polja.
- Ako je deklarirana duljina polja veća od broja pridruženih podataka, preostalim se članovima pridružuje vrijednost 0.

(c) S.Šutalo i D.Grundler, 2009.

19

Primjer 90



- Članovima polja A[6] iz prethodnog primjera, pridružiti samo četiri podatka
(`int A[6]={2,34,1,67};`).

(Rezultat: Preostalim se članovima pridružuje 0.)

A[0]=2 A[1]=34 A[2]=1 A[3]=67 A[4]=0 A[5]=0

(c) S.Šutalo i D.Grundler, 2009.

20

Primjer 91



- Treba unijeti N članova (N je manji ili jednak 10) jednodimenzionalnog polja, pa ih ispisati i zbrojiti.
- Ispis neka bude oblika:

Upisi broj članova polja:
Upisi X[...] član polja:
....
....

Ako je N=..., zbroj članova polja iznosi

(c) S.Šutalo i D.Grundler, 2009.

21

Primjer 91

- Kako N nije unaprijed poznat, treba ga unijeti pa provjeriti da li je $N \leq 10$ i $N > 0$.
- Provjerava se, npr. *do-while* petljom.
- Prva *for* petlja će poslužiti za upis i zbrajanje članova polja, a druga za ispis.

(c) S.Šutalo i D.Grundler, 2009.

22

Primjer 91

```
#include <iostream>
using namespace std;
int main ()
{
    cout<<"Program ispisuje i zbraja N članova polja X. ";
    cout<<"N mora biti <=10."<<endl;
    int N,i,zbroj;
    int X[10];
    zbroj=0;
    do
    {
        cout<<"Upisi broj članova polja:";
        cin>>N;
    }
    while((N<=0)|| (N>10));
```

N mora biti
veći od 0 i
manji ili
jednak 10

(c) S.Šutalo i D.Grundler, 2009.

23

Primjer 91

```
for (i=0;i<N;i++)
{
    cout<<endl<<"Upisi X["<<i<<"] član polja:";
    cin>>X[i];
    zbroj=zbroj+X[i];
}

cout<<endl<<"Članovi polja X su:"<<endl;
for (i=0;i<N;i++)
{
    cout<<X[i]<<endl;
}

cout<<endl<<"Ako je N="<<N<<" zbroj članova polja iznosi "
<<zbroj<<endl;

return 0;
```

upis i
zbrajanje
vrijednosti
članova
polja

ispis članova
polja

(c) S.Šutalo i D.Grundler, 2009.

24

```
#include <iostream>
using namespace std;
int main ()
{
    int N,i,zbroj;
    int X[10];
    zbroj=0;
    do
    {
        cout<<"Upisi broj clanova polja:";
        cin>>N;
    }
    while (N<=0) || (N>10);
}
```

```
for (i=0;i<N;i++)
{
    cout<<endl<<"Upisi X["<<i<<" ] clan polja:";
    cin>>X[i];
}
for (i=0;i<N;i++)
{
    cout<<X[i]<<endl;
    zbroj=zbroj+X[i];
}
cout<<endl<<"Ako je N="<<N<<" zbroj "<<N
<<" clanova polja iznosi "<<zbroj<<endl;
return 0;
}
```

Primjer 91

```
Program ispisuje i zbraja N clanova polja X. N mora biti <=10.
Upisi broj clanova polja:4
Upisi X[0] clan polja:23
Upisi X[1] clan polja:13
Upisi X[2] clan polja:5
Upisi X[3] clan polja:26
Clanovi polja X su:
23
13
5
26
Ako je N=4 zbroj 4 clanova polja iznosi 67
```

(c) S.Šutalo i D.Grundler, 2009.

27

Dvodimenzionalno polje

- Dvodimenzionalno polje se može predložiti tablicom sa zadanim brojem redaka i stupaca.
- Položaj člana unutar dvodimenzionalnog polja označen je sa dva cjelobrojna indeksa.
- Prvi indeks određuje redak, a drugi stupac.

(c) S.Šutalo i D.Grundler, 2009.

28

Dvodimenzionalno polje

- Prvi član dvodimenzionalnog polja (prvi redak, prvi stupac) označen je indeksom:
[0,0]
- Posljednji (posljednji redak, posljednji stupac) indeksom:
[(broj redaka-1),(broj stupaca-1)].

(c) S.Šutalo i D.Grundler, 2009.

29

Deklaracija dvodim. polja

- Dvodimenzionalno polje se deklarira, npr.:
int X [2] [3];
- Za dvodimenzionalno polje X[2] [3], indeksi su:
X[0,0], X[0,1], X[0,2]
X[1,0], X[1,1], X[1,2]

(c) S.Šutalo i D.Grundler, 2009.

30

Dvodimenzionalno polje

- Vrijednosti članovima dvodimenzionalnog polja pridružuju se tako da se vrijednosti članova svakog retka navedu unutar para vitičastih zagrada.

```
{ { ..., ..., ... },  
  { ..., ..., ... }  
  ...  
};
```

(c) S.Šutalo i D.Grundler, 2009.

31

Pridruživanje vrijednosti

- Prethodno deklariranom polju (int X [2] [3]) treba pridružiti sljedeće vrijednosti podataka:

	1. stupac	2. stupac	3. stupac
1.redak	1	2	3
2.redak	4	5	6

(c) S.Šutalo i D.Grundler, 2009.

32

Pridruživanje vrijednosti

- Članovima dvodimenzionalnog polja X [2] [3] se vrijednosti podataka pridružuju ovako:

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

(c) S.Šutalo i D.Grundler, 2009.

33

Primjer 92

- Potrebno je unijeti članove dvodimenzionalnog polja, pa ih ispisati u obliku tablice.
- Ispis treba izgledati ovako:

12	25	38
44	51	16

(c) S.Šutalo i D.Grundler, 2009.

34

Primjer 92

```
#include <iostream>  
using namespace std;  
int main ()  
{  
    int i, j;  
    int X[2][3]={{12,25,38},  
                {44,51,16}};  
    for(i=0; i<2; i++)  
    {  
        for(j=0; j<3; j++)  
        {  
            cout<<X[i][j]<<" ";  
        }  
        cout<<endl;  
    }  
    return 0;  
}
```

deklaracija polja
X i pridruživanje
vrijednosti
članovima polja

ispis vrijednosti
[i][j]
člana polja

(c) S.Šutalo i D.Grundler, 2009.

35

```
#include <iostream>  
using namespace std;  
int main ()  
{  
    int i, j;  
    int X[2][3]={{12,25,38},  
                {44,51,16}};  
    for(i=0; i<2; i++)  
    {  
        for(j=0; j<3; j++)  
        {  
            cout<<X[i][j]<<" ";  
        }  
        cout<<endl;  
    }  
    return 0;  
}
```

Primjer 92

- Ako se prethodni primjer dopuni, uz vrijednost pojedinog člana polja bit će prikazani i njegovi indeksi.

```
cout<<"X["<<i<<"["<<j<<"]= "<<X[i][j]<<" ";
```

```
C:\
X[0][0]= 12  X[0][1]= 25  X[0][2]= 38
X[1][0]= 44  X[1][1]= 51  X[1][2]= 16
```

(c) S.Šutalo i D.Grundler, 2009.

37

Duljina polja (pri deklaraciji)

- Ako se kod pridruživanja vrijednosti izostave vitičaste zagrade koje odjeljuju vrijednosti po redovima, članovi se popunjavaju po redu (red po red).
- Ako je deklarirana duljina polja veća od broja pridruženih podataka, preostali se članovima pridružuje vrijednost 0.

(c) S.Šutalo i D.Grundler, 2009.

38

Primjer 93



- Članovima polja X[2][3], iz prethodnog primjera, pridružiti samo četiri podatka:

```
int X[2][3]={12, 25, 38, 44};
```

(Rezultat: Preostali se članovima pridružuje 0.)

```
X[0][0]= 12  X[0][1]= 25  X[0][2]= 38
X[1][0]= 44  X[1][1]= 0   X[1][2]= 0
```

(c) S.Šutalo i D.Grundler, 2009.

39

Znak

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

```
char slovoA='A';
char SLOVOA=65;
```

(c) S.Šutalo i D.Grundler, 2009.

40

Znakovni niz

- Više znakova čini znakovni niz (engl. *character string*).
- Znakovni niz je jednodimenzionalno polje čiji su članovi znakovi (*char*).

(c) S.Šutalo i D.Grundler, 2009.

41

Znakovni niz

- Znakovnom nizu vrijednosti se pridružuju tako da se navedu nakon operatora pridruživanja, između para dvostrukih navodnika. Npr.:

```
char X [] = "Ovo je znakovni niz"
```

(c) S.Šutalo i D.Grundler, 2009.

42

Znakovni niz

- Znakovni niz se u memoriju sprema ovako:

'O'	'v'	'o'	' '	'j'	'e'	' '	'z'	'n'	'a'	'k'	'o'	'v'	'n'	'i'	' '	'n'	'i'	'z'	'\0'
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	------

(c) S.Šutalo i D.Grundler, 2009.

43

Znakovni niz

- 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.

(c) S.Šutalo i D.Grundler, 2009.

44

Primjer 94



- Za svaki član znakovnog niza: "Ovo je znakovni niz" treba zapisati indeks i njegovu vrijednost.
- Ispis treba izgledati ovako:

```
X[0]=O
X[1]=v
X[2]=o
X[3]=
X[4]=j
X[5]=e
X[6]=
X[7]=z
X[8]=n
X[9]=a
X[10]=k
X[11]=o
X[12]=v
X[13]=n
X[14]=i
X[15]=
X[16]=n
X[17]=i
X[18]=z
X[19]=
```

(c) S.Šutalo i D.Grundler, 2009.

45

Primjer 94

```
#include <iostream>
using namespace std;
int main ()
{
    int i;
    char X[]="Ovo je znakovni niz";
    for(i=0;i<20;i++)
    {
        cout<<"X["<<i<<"]="<<X[i]<<" "<<endl;
    }
    return 0;
}
```

ispis indeksa i vrijednosti
i-tog člana polja

(c) S.Šutalo i D.Grundler, 2009.

46

```
#include <iostream>
using namespace std;
int main ()
{
    int i;
    char X[]="Ovo je znakovni niz";
    for(i=0;i<20;i++)
    {
        cout<<"X["<<i<<"]="<<X[i]<<" "<<endl;
    }
    return 0;
}
```

Primjer 94

- Provjera:

```
C:\
X[0]=O
X[1]=v
X[2]=o
X[3]=
X[4]=j
X[5]=e
X[6]=
X[7]=z
X[8]=n
X[9]=a
X[10]=k
X[11]=o
X[12]=v
X[13]=n
X[14]=i
X[15]=
X[16]=n
X[17]=i
X[18]=z
X[19]=
```

'\0'

(c) S.Šutalo i D.Grundler, 2009.

48

Znakovni niz

- Ako se znakovni niz proteže kroz više redaka, završetak svakog retka se označava lijevom kosom crtom (\ engl. *backslash*).

```
#include<iostream>
using namespace std;
int main()
{
    cout<<"Ovo je primjer u kome se želi pokazati \n
    kako znakovni niz može zauzeti više redaka \n
    Potrebno je završetak prvog dijela niza označiti lijevom kosom crtom."<<endl;
    return 0;
}
```

(c) S.Šutalo i D.Grundler, 2009.

49

Znakovni niz

- Kosa crta (\) označava da će se ispis protezati do kraja zaslona a zatim preći u novi redak.

Ovo je primjer u kome se želi pokazati kako znakovni niz može zauzeti više redaka. Potrebno je završetak prvog dijela niza označiti lijevom kosom crtom.

(c) S.Šutalo i D.Grundler, 2009.

50

Primjer 95

- Treba prebrojiti koliko se puta u znakovnom nizu pojavljuje slovo o :

Koliko slova o ima u ovoj recenici?

U znakovnom nizu:

Koliko slova o ima u ovoj
recenici?
ima ... slova o.

(c) S.Šutalo i D.Grundler, 2009.

51

Primjer 95

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

- Da li je sadržaj i-tog člana polja jednak znaku o?
- Ako je, brojač se uveća za 1.

(c) S.Šutalo i D.Grundler, 2009.

52

```
#include <iostream>
using namespace std;
int main ()
{
    cout<<"Program prebrojava pojavljivanje \
slova o u recenici"<<endl;
    int i,brojo;
    broj=0;
    char X[]="Koliko slova o ima u ovoj recenici?";
```

```
    for (i=0;X[i]!='\0';i++)
    {
        if(X[i]=='o')
        {
            broj=broj+1;
        }
    }
    cout<<endl<<"U znakovnom nizu:"<<endl
    <<"Koliko slova o ima u ovoj recenici?"
    <<endl<<"ima "<<broj<<" slova o."<<endl;
    return 0;
}
```

Primjer 95

- Provjera pokazuje koliko u znakovnom nizu ima slova o.

```
CA
Program prebrojava pojavljivanje slova o u recenici
U znakovnom nizu:
Koliko slova o ima u ovoj recenici?
ima 6 slova o.
```

