**Računalstvo III1 18.05.**

**Jednodimenzionalni niz**

Jednodimenzionalni niz je najjednostavniji oblik niza. U njemu se elementi dohvaćaju s pomoću samo jednog indeksa.

Pojam jednodimenzionalnog polja ekvivalentan je pojmu vektora u matematici

Elementi niza nalaze se u linearnom slijedu.

Niz se, kao i svaka druga varijabla u programu, mora najaviti (deklarirati).

Ovo je opći oblik deklaracije jednodimenzionalnog niza:

**tip\_elemenata\_niza** ime (duljina\_niza);

gdje je:

* **tip\_elementa\_niza**  jedan od osnovnih tipova podataka u C-u (**int, float, double, char)**, a odnosi se na tip elementa koji će se upisivati u niz
* **ime –** identifikator
* **duljina\_niza** najveći mogući broj elemenata koji se mogu upisati u dani niz.

Kako to izgleda u memoriji?

Deklaracijom jednodimenzionalnog niza u memoriji se za svaki element rezervira posebna memorijska lokacija. Rezerviranje lokacije obično idu jedna iza druge.

Na primjer, ako se deklarira realna varijabla a (**float** a), u memoriji se zauzme prostor za jedan realan broj. To lokacija pristupa se s pomoću imena varijable:

|  |  |
| --- | --- |
| ime: | **a** |
| vrijednost: | 3.14 |

Program prevoditelj je već prilikom deklaracije osigurao mjesto za jednu realnu varijablu (4 bajta).

Kada se deklarira niz od 9 realnih elemenata(float a [9]), program prevoditelj osigurava u memoriji neprekinuti prostor za 9 realnih podataka:

indeksi

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|  |  |  |  |  |  |  |  |  |  |

Ime niza

Ako se niz deklarira kao globalna varijabla (prije funkcije **main()**) onda je početna vrijednost svih elemenata niza jednaka 0. No ako se deklarira kao lokalna varijabla, onda elementi niza nisu inicijalizirani (nisu postavljeni na određenu početnu vrijednost). To znači da su elementi tako deklariranog niza popunjeni slučajnim vrijednostima (ovisno o tome što se nalazilo na tome mjestu u memoriji prije navedene deklaracije).

Ako je potrebno niz se može inicijalizirati prilikom deklaracije ili u programu.

Primjer:

**inicijalizacija niza prilikom deklaracije:**

**int** b[5]={10,20,30,40,50}

indeksi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| b | 0 | 1 | 2 | 3 | 4 |
|  | 10 | 20 | 30 | 40 | 50 |

ime

vrijednosti elemenata niza

Kod ovakve inicijalizacije potrebno je pripaziti da broj elemenata unutar vitičastih zagrada ne bude veći od duljine polja. Može biti manji ili jednak.

Ako prilikom inicijalizacije skip vrijednosti ima manje elemenata nego što je deklarirano, vrijednosti preostalih elemenata su 0 (to se odnosi na inicijalizaciju niza kao globalne varijable).

Na primjer:

Int b[7] = {10, 20, 30, 40, 50}

dat će ovaj niz u memoriji:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| b[0] | b[1] | b[2] | b[3] | b[4] | b[5] | b[6] |
| 10 | 20 | 30 | 40 | 50 | 0 | 0 |

Da bi se izbjegle moguće pogreške, duljina niza se može i izostaviti. Program prevoditelj tada će sam odrediti duljinu niza, prema broju elemenata navedenih unutar vitičastih zagrada.

Navedeni niz može se deklarirati i incijalizirati i na ovaj način:

Int b[] = {10, 20, 30, 40, 50};

Zauzeti memorijski prostor u ovom slučaju izgleda ovako:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| b[0] | b[1] | b[2] | b[3] | b[4] |
| 10 | 20 | 30 | 40 | 50 |

Za dohvaćanje vrijednosti bilo kojeg elementa niza potrebno je znati njegovo mjesto u nizu, odnosno njegov indeks. S obzirom na to da indeksi u C-u (za razliku do Pascala) kreću od nule, treba voditi računa o tome da je element na i-tom mjestu u nizu, zapravo, element i – 1. Ako treba dohvatiti treći član niza b, to je element b[2]. Vrijednost elementa b[2] u navedenom primjeru je 30.

Naredbom

printf(„\n %d“,b[3]);

na zaslonu će se ispisati broj 40, jer to je vrijednost koja se nalazi na 4. mjestu u nizu ( i ima indeks 3).

**Zadatak 1.**

Napišate program koji traži unošenje vrijednosti u niz od 20 elemenata, a zatim ih ispisuje na zaslon monitora.

#include <stdio.h>

int i,n,a[20];

void main()

{

 printf("upisite broj elemenata u polju: ");

 scanf("%d",&n);

 if (n<=20 && n>0)

 {

 printf("\n upisi elemente polja:\n");

 for (i=0; i<n; i++)

 {

 printf("a[%d] ",i);

 scanf("%d",&a[i]);

 }

 printf("\n ucitani elementi su: \n");

 for (i=0; i<n; i++)

 printf("\n %d. el. je %d",i, a[i]);

 }

 else

 printf("\n upisali ste krivi broj");

}

Za zadaću:

**Zadatak 1.**
Napravite program koji će na zaslon ispisati prvih 20 prirodnih brojeva, ali u obrnutom redoslijedu (od 20 do 1)