



Deklaracija strukture

Za razliku od polja, kod kojih svi elementi moraju biti istog tipa, strukture omogućavaju grupiranje elemenata različitih tipova u jednu cjelinu. Opći oblik najave strukture je:

```
struct ime_strukture  
{  
    najava članova strukture;  
};
```

gdje je:

- * **struct** ključna riječ koja označava početak najave strukture
- * *ime_strukture* identifikator strukture

Najava strukture mora završiti znakom ;

Struktura se može najaviti:

- 1) na početku programa (tj. izvan svih funkcija),
- 2) unutar neke funkcije,
- 3) unutar nekog bloka naredbi.

U prvom se slučaju struktura može pozvati iz bilo kojeg dijela programa, jer je globalna, a u ostala dva slučaja struktura vrijedi samo unutar funkcije ili bloka naredbi u kojem je najavljena.

Primjer: Najava strukture za kompleksne brojeve.

```
struct kompleksni  
{  
    float re;  
    float im;  
};
```

Definicija strukturnih varijabli uobičajeno se piše odmah iza najave strukture, npr.:

```
struct kompleksni
{
    float re;
    float im;
} a, b, z;
```

Sve tri strukturne varijable (a, b, z) se sastoje od dva člana: *re* i *im*.

Najava strukturnih varijabli može biti i ovako napisana:

```
struct kompleksni
{
    float re;
    float im;
};
struct kompleksni a, b, z;
```

S obzirom na to da ne postoji poseban format za ispis strukturnih varijabli, pri unosu ili ispisu se svaki član strukturne varijable tretira posebno, npr.:

```
printf ("n%.2f+%.2f i", z.re, z.im);
```

```
scanf ("%f %f",&z.re, &z.im);
```

Strukturne varijable se mogu, kao i sve ostale varijable, inicijalizirati:

- * prilikom najave:

```
struct kompleksni z={7.2, 3.1};
```
- * posebnim blokom naredbi, bilo gdje u programu:

```
z.re=7.2;  
z.im=3.1;
```

U deklaraciji strukture se mogu upotrebljavati polja osnovnih i složenijih tipova podataka, ali i druge strukture. Broj struktura unutar strukture nije ograničen.

Primjer:

Ako je datum struktura koja sadržava dan, mjesec i godinu.

```
struct datum
{
    int dan;
    int mjesec;
    int godina;
};
```

može se iskoristiti za datum rođenja unutar strukture koja sadržava osobne podatke učenika:

```
struct ucenik
{
    char ime[15];
    char prezime[15];
    int maticni;
    float prosjek;
    struct datum rodjendan;
} ucenik[40];
```

Ako treba inicijalizirati članove složene strukture kao što je ova, članove podstrukture treba pisati u posebnoj vitičastoj zagradi, npr.

```
struct ucenik ucenik[10]={{"Pero", "Peric", 234, 3.56, {25, 4, 1992}}};
```

Članovima podstrukture pristupa se na isti način kao i članovima strukture uporabom točke operatora, ali potrebno je spustiti se na razinu člana te strukture, npr.:

```
ucenik[10].rodjendan.godina=1992;
```

gdje je: ucenik[10] - strukturna varijabla,
rodjendan - prva razina,
godina - druga razina.

Zadaci za vježbu:

1. Napišite C program koji će omogućiti unošenje imena, prezimena, matičnog broja, prosjeka i datuma rođenja za sve učenike jednog razreda. Ispisati koji učenik ima najbolji prosjek. (Učenike upisati u polje od najviše 40 elemenata.)
2. Napišite C program u kojem se nalazi deklaracija strukture Student koja ima sljedeće podatke: ime, prezime, godinu studija, smjer, prosjek položenih ispita, matični broj studenta. Program treba s tipkovnice ponavljati učitavanje cijelog broja m sve dok je učitani broj manji od 2 ili veći od 100. Program zatim treba tražiti m puta unos podataka u strukturu Student i zapisati podatke o svakom studentu.
3. Napišite C program koji će omogućiti upis dva kompleksna broja te ispisati njihov zbroj, razliku i produkt.