

NAŠTEVNI PODATKOVNI TIP

Programski jezik C sam po sebi pozna že nekaj osnovnih naštevnihih podatkovnih tipov.

int = {INT_MIN, ... -2,-1,0,1,2...INT_MAX};

char = {CHAR_MIN..CHAR_MAX};

Naštevni podatkovni tip - enum

Tip enum lahko nadomesti več stavkov deklaracije konstant. Za besedo enum, ki pove, da gre za naštevni tip, navedemo ime naštevnege tipa. Nato navedemo vse možne vrednosti, ki jih bo spremenljivka naštevnege tipa lahko vsebovala. Spremenljivko barva naštevnege tipa barvni deklariramo kot:

```
enum barvni { rdeca, zelena, modra } barva;
```

Prvi simbol bo imel vrednost 0 (rdeca = 0) in vsak naslednji simbol bo imel za eno večjo vrednosti.

Kot zaloga vrednosti ne moremo imeti posebnih znakov (torej ime rdeča,četrtek,... odpade).

Je pa mogoče prirediti določena števila, ki jih bodo imeli posamezni simboli:

```
enum obseg {suha = 65, vitka = 70, normalna = 80, debela = 95, zeloDebela = 135};
```

Prevajalniki so dovolj "natančni", da ugotovijo, in javijo napako, če priredimo spremenljivki tipa enum vrednost, ki ni v seznamu simbolov. Spodnji program prikazuje uporabo naštevnege tipa.

```
#include <stdio.h>
enum barvni {rdeca, zelena, modra, bela, crna, siva } barva;
int main(void)
{
    for (barva = rdeca; barva <= siva; barva++)
        printf("%d ",barva);
    printf("\n");
    return 0;
}
```

Program izpiše: 0 1 2 3 4 5

Prirejanje vrednosti:

```
#include <stdio.h>
enum barvni {rdeca, zelena, modra, bela, crna, siva };
int main(void)
{
    enum barvni barva;
    printf("Vnesi barvo: ");
    scanf("%d",&barva); // ne moremo vnesti rdeca, ampak vnesemo 0
    printf("%d\n",barva); // izpiše 0
    barva = modra;
    printf("%d\n",barva); // izpiše 2
    return 0;
}
```

Vrednosti ne moremo kar direktno prebrati (primer o barvah in da bi vnesli rdeca), ampak jih je potrebno prebrati preko številčnih vrednosti

Primer kako izpišemo posamezno barvo:

```
#include <stdio.h>
```

```
enum barvni {rdeca, zelena, modra, bela, crna, siva }; // naredimo naštevni tip
```

```
int main(void)
```

```
{
```

```
    enum barvni barva; // deklaracija spremenljivke naštevnega tipa
```

```
    printf("Vnesi barvo: ");
```

```
    scanf("%d",&barva); // ne moremo vnesti rdeca
```

```
    printf("%d\n",barva); // izpisemo stevilko
```

```
    // glede na izbrano barvo le to izpisemo
```

```
    switch (barva)
```

```
    {
```

```
        case rdeca: printf("Rdeca\n"); break;
```

```
        case zelena: printf("Zelena\n"); break;
```

```
        case modra: printf("Modra\n"); break;
```

```
        case bela: printf("Bela\n"); break;
```

```
        case crna: printf("Crna\n"); break;
```

```
        case siva: printf("Siva\n"); break;
```

```
    }
```

```
    return 0;
```

```
}
```

Izpisovanje na tak način je sicer nerodno, ampak žal v programskem jeziku C, za izpis naštevnega podatkovnega tipa, drugačnega ničina ni.

Naslednji programska sekvenca prikazuje prekoračitve intervala znotraj naštevnega tipa. Paziti je potrebno pri obeh mejnih primerih (prvi in zadnji podatek naštevnega tipa).

```
#include <stdio.h>
```

```
enum barvni {rdeca, zelena, modra, bela, crna, siva };
```

```
int main(void)
```

```
{
```

```
    enum barvni barva;
```

```
    barva = siva;
```

```
    barva++;
```

```
    // glede na izbrano barvo le to izpisemo
```

```
    switch (barva)
```

```
    {
```

```
        case rdeca: printf("Rdeca\n"); break;
```

```
        case zelena: printf("Zelena\n"); break;
```

```

case modra: printf("Modra\n"); break;
case bela: printf("Bela\n"); break;
case crna: printf("Crna\n"); break;
case siva: printf("Siva\n"); break;
default:
    printf("%d\n",barva);
}

return 0;
}

```

Ker ni nobene barve za sivo nam switch stavek ne najde podatka za izpis in tako se izpiše samo številka – v našem primeru 6.

Naštevne podatkovne tipov lahko tudi sestevamo/odštevamo/množimo/delimo – paziti je le potrebno na interval števil – da ne prekoračimo.

```

#include <stdio.h>
enum barvni {rdeca, zelena, modra, bela, crna, siva };
int main(void)
{
    enum barvni b1,b2;

    b1 = zelena;
    b2 = modra;
    printf("%d\n",b1+b2); // izpise 3
    b1 = b1 * b2;
    printf("%d\n",b1); // izpise 2

    return 0;
}

```

Primerjanje vrednosti:

```

#include <stdio.h>
enum teden {pon, tor, sre, cet, pet, sob, ned };
int main(void)
{
    enum teden danes;
    danes = pon;

    if (danes < pet)
        printf("Se se je potrebno ucit!\n");
    else
        printf("Malo pocitka\n");

    return 0;
}

```

Naloge

1. Napiši funkcijo, ki prebere višino človeka podano v centimetrih in razvrsti človeka v ustrezno kategorijo: zelo majhen, majhen, srednji,... Uporabite naštevni tip. Zelo veliki ljudje merijo več kot 200cm v višino, veliki več kot 180, srednji več kot 170...
2. Napišite funkcijo, ki bo izpisala vrsto vremena (toplo, vroče, mrzlo,...), če vpišemo dnevno temperaturo v stopinjah celzija. Vreme je vroče pri temperaturah nad 30 stopinj celzija...
3. Definirajte naštevni tip meseci (jan, feb, mar, apr,...) in ga uporabite za izpis števila dni posameznega meseca.