

Naredbe uslovnog grananja

Rešavanje većine problema često zahteva izvršavanje određenog dela programa u zavisnosti od ispunjenosti nekog uslova. Iz tog razloga neophodno je postojanje razgranatih struktura, koje se u programskom jeziku C ostvaruju korišćenjem naredbe uslovnog grananja **if** i naredbe višestrukog grananja **switch**.

Naredba uslovnog grananja – IF

Naredba uslovnog grananja **if** omogućava izvršavanje određenog dela programa u zavisnosti od toga da li je navedeni uslov ispunjen ili ne. Ukoliko je uslov ispunjen, tj. ukoliko je vrednost logičkog iskaza **tačno (true)**, onda se izvršava jedna naredba ili blok naredbi navedenih unutar komande **if**. Ukoliko je vrednost logičkog izraza **netačno (false)**, može se izvršiti neka druga naredba, odnosno blok naredbi.

Sintaksa

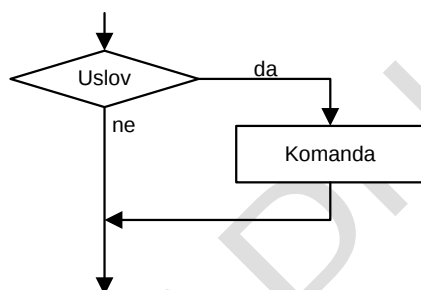
```
if (<logicki_izraz>) <naredba1> [ else <naredba2> ];
```

Korišćenjem ovakve **if – else** konstrukcije moguće je formirati više oblika uslovnog grananja, kao što su: jednostruko, dvostruko i višestruko grananje.

Jednostruko grananje

Kod jednostrukog grananja računar ispituje vrednost logičkog izraza i ukoliko je ova vrednost **tačno (true)**, izvršava se naredba, odnosno blok naredbi iza naredbe **if**. Ukoliko je vrednost izraza **netačno (false)**, ova naredba se ne izvršava.

Na slici je prikazan šematski prikaz ovakvog grananja:



Slika ### Šematski prikaz jednostrukog grananja

C kod koji bi odgovarao ovakvoj vrsti grananja bi bio:

```
if (logicki_izraz) naredba;

ili

if (logicki_izraz)
{
    naredba_1;
    naredba_2;
    ...
    naredba_n;
}
```

Primer

Napisati program koji na osnovu broja utrošenih kilovata električne energije P i cene po kilovatu C , izračunava ukupnu naknadu koju potrošač treba da plati T . Ukoliko je naknada veća od 1000 dinara, račun se uvećava za 10%, kako bi se stimulisala štednja električne energije.

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

```
main()
{
    const float LIMIT = 1000.0; {Limit iznad koga se uvecava naknada}
    const float PENALI = 10.0; {Procenat za koji se uvecava naknada}
    float P,C,T;

    printf("Unesite broj untrosenih kilovata P: ");
    scanf("%f", &P);
    printf("Unesite cenu po kilovatu C: ");
    scanf("%f", &C);

    T = P * C;

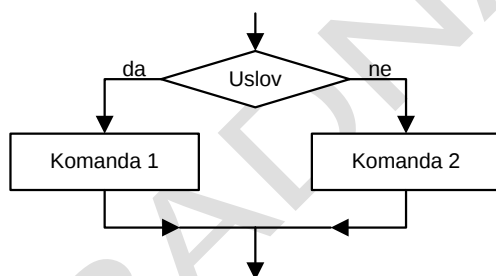
    if(T > LIMIT) T = T + T * PENALI / 100.0;

    printf("Vas racun za elektricnu energiju je %f dinara.\n", T);
}
```

Dvostruko grananje

U slučaju dvostrukog grananja, ukoliko je logički izraz tačan izvršava se naredba ili blok naredbi iza naredbe **if**, a ako logički izraz nije tačan izvršava se naredba, odnosno blok naredbi, koji se navode iza rezervisane reči **else**.

Šematski prikaz koji opisuje ovakvu vrstu grananja prikazan je na slici:



Slika ### Šematski prikaz dvostrukog grananja

C kod koji bi odgovarao dvostrukom grananju bi bio:

```
if (logički_izraz) naredba_1;
else naredba_2;

ili

if (logički_izraz)
{
    naredba_11;
    naredba_12;
    ...
    naredba_1n;
}
```

```
    }  
else  
{  
    naredba_21;  
    naredba_22;  
    ...  
    naredba_2m;  
}
```

Primer

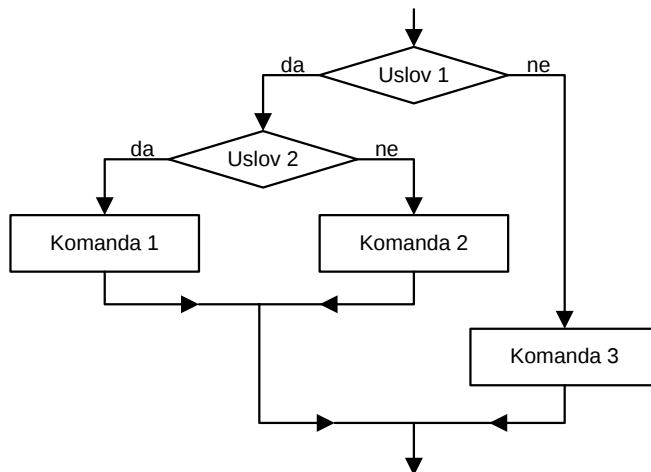
Napisati program koji za x dato na ulazu izračunava vrednost funkcija $f(x)$ i $g(x)$ koje imaju oblik:

$$f(x) = \begin{cases} -1, & \text{ako je } x < 0 \\ 2x + 3, & \text{ako je } x \geq 0 \end{cases} \quad \text{i} \quad g(x) = \begin{cases} \ln(-x), & \text{ako je } x < 0 \\ e^x, & \text{ako je } x \geq 0 \end{cases}$$

```
#include <stdio.h>  
#include <math.h>  
  
main()  
{  
    float x,f,g;  
  
    printf("Unesite x: ");  
    scanf("%f", &x);  
  
    if(x < 0)  
    {  
        f = -1.0;  
        g = log(-x);  
    }  
    else  
    {  
        f = 2.0*x+3.0;  
        g = exp(x);  
    }  
  
    printf("f(x)=%f\n", f);  
    printf("g(x)=%f\n", g);  
}
```

Višestruko grananje

Imajući u vidu da se unutar **if-else** konstrukcije mogu naći bilo koja naredba ili blok naredbi, zaključujemo da to mogu biti i druge **if** naredbe. Korišćenjem takvih konstrukcija možemo ostvariti višestruko grananje, kao što je to prikazano na slici:



Slika ### Šematski prikaz višestrukog grananja

U ovom slučaju program prvo ispituje da li je prvi uslov ispunjen. Ukoliko *Uslov 1* nije ispunjen, program izvršava *Komandu 3*. Međutim, ako je *Uslov 1* ispunjen, program ispituje *Uslov 2* i u zavisnosti od toga da li je on ispunjen ili nije, program izvršava *Komandu 1* ili *Komandu 2*. Na ovaj način dobili smo algoritamsku strukturu sa trostrukim grananjem, koja može izvršavati neku od tri zadate komande u zavisnosti od ispunjenosti pojedinih uslova. Na sličan način, kombinacijom **if** i **else** naredbi, možemo dobiti proizvoljan broj uslovnih grananja.

Primer

Napisati program koji na osnovu broja bodova osvojenih na ispitu, određuje ocenu prema sledećoj tabeli:

Broj bodova	Ocena
51-60	6
61-70	7
71-80	8
81-90	9
91-100	10

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

```
main()
```

```
{
```

```
    int bodovi,ocena;
```

```
    printf("Unesite broj bodova: ");
```

```
    scanf("%d", &bodovi);
```

```
    if(bodovi >= 91) ocena = 10;
```

```
    else
```

```
        if(bodovi >= 81) ocena = 9;
```

```
        else
```

```
            if(bodovi >= 71) ocena = 8;
```

```
            else
```

```
                if(bodovi >= 61) ocena = 7;
```

```
                else
```

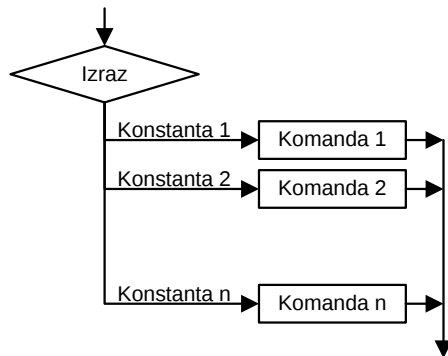
```
                    if(bodovi >= 51) ocena = 6;
```

```
                    else ocena = 5;
```

```
printf("Student je dobio ocenu %d\n", ocena);  
}
```

Naredba višestrukog grananja – SWITCH

Iako se višestruko grananje može ostvariti korišćenjem naredbe **if**, program pisan na takav način je često veoma nepregledan. Da bi se to izbeglo, u programskom jeziku C postoji naredba višestrukog grananja **switch**, koja omogućava grananje programa u zavisnosti od vrednosti nekog izraza, kao što je to prikazano na slici:



Slika ### Šematski prikaz višestrukog grananja pomoću naredbe **case**

Iza naredbe **switch** navodi se izraz koji može biti bilo kog nabrojivog tipa (char, bool, int, skup ili intervalni tip). Nakon izraza se nabrajaju moguće vrednosti izraza i odgovarajuće naredbe ili blokovi naredbi.

Sintaksa

```
switch (<izraz>)  
{  
    <case konstanta_1: naredba_1; break;>  
    [case konstanta_2: naredba_2; break;]  
    ...  
    [case konstanta_n: naredba_n; break;]  
    [default: naredba_def;]  
}  
end;
```

Napomenimo da posle naredbe ne mora da sledi komanda **break**. U tom slučaju naredba se izvršava ukoliko je navedeni izraz jednak bilo kojoj od uzastupnih konstanti.

Switch naredba funkcioniše tako što se prvo izračunava vrednost izraza, a zatim traži konstanta koja je jednaka vrednosti izraza. Ukoliko takva konstanta postoji, izvršava se naredba navedena iza konstante. Ako takve konstante nema, onda se nastavlja izvršavanje programa zanemarujući naredbu **switch**. Moguće je korišćenje rezervisane reči **default** umesto konstante iza koje se navodi naredba koja će se izvršiti u slučaju da vrednost izraza nije jednaka ni jednoj od navedenih konstanti.

Primer 1

Napisati program koji na osnovu ocene od 1 do 5, određuje opisnu ocenu koja može biti "nedovoljan", "dovoljan", "dobar", "vrlo dobar" ili "odličan".

```
#include <stdio.h>

main()
{
    int ocena;

    printf("Unesite ocenu: ");
    scanf("%d", &ocena);

    printf("Ucenik je ");

    switch(ocena)
    {
        case 1:
            printf("nedovoljan");
            break;
        case 2:
            printf("dovoljan");
            break;
        case 3:
            printf("dobar");
            break;
        case 4:
            printf("vrlo dobar");
            break;
        case 5:
            printf("odlican");
            break;
    }
}
```

Primer 2

Napisati program koji određuje broj dana u zadatom mesecu.

```
#include <stdio.h>

main()
{
    int mesec;
    char prestupna;

    printf("Unesite redni broj meseca: ");
    scanf("%d", &mesec);

    switch(mesec)
    {
        case 1:
        case 3:
        case 5:
        case 7:
        case 8:
        case 10:
        case 12:
            printf("Mesec ima 31 dan.\n");
            break;
        case 4:
        case 6:
        case 9:
    }
```

```
    case 11:
        printf("Mesec ima 30 dana.\n");
        break;
    case 2:
        printf("Da li je godina prestupna (D/N)? ");
        getchar(); // Ciscenje bafera !!!
        scanf("%c", &prestupna);
        if(prestupna == 'D' || prestupna == 'd')
            printf("Mesec ima 29 dana.\n");
        else
            printf("Mesec ima 28 dana.\n");
        break;
    default:
        printf("Pogresan broj meseca.\n");
        break;
}
}
```