

Razgranata struktura
predavač: Nadežda Jakšić

Programiranje programski jezik C++

Grananje u programu

- ispituje se uslov (logički izraz)
- u zavisnosti od toga da li je uslov ispunjen ili ne, prelazi se na neku granu programa
- uslov je logički izraz i rezultat ispitivanja je nula, ili vrednost različita od nule
- vrednost nula se tumači kao FALSE (uslov nije ispunjen), a bilo koja nenulta vrednost kao TRUE (uslov je ispunjen)
- grananje se opisuje se konstrukcijama:

if

if else

ugnježdeni if

switch

If

```
if (uslov)
{
    naredba ili blok naredbi;
}
```

```
int main()
{
    int a;
    cout<<"Unesi broj:"<<endl;
    cin>>a;
    if (a>0) cout<<"Prvi manji broj od unetog broja je "<<a-1;
    if (a<0) cout<<"Prvi veci broj od unetog broja je "<<a+1;
    return 0;
}
```

If else

```
if (uslov)
{
    naredba ili blok naredbi;
}
else
{
    naredba ili blok naredbi;
}
```

```
//da li je prvi broj veći od drugog
int main() {
    int p, d;
    cin >> p >> d;
    if (p > d)
        cout << "DA" << endl;
    else
        cout << "NE" << endl;
    return 0;}

```

ternarni operator

uslov?naredba1:naredba2; ako je uslov ispunjen izvršiće se naredba1, a ako uslov nije ispunjen izvršiće se naredba2

```
cout << (p > d ? "DA" : "NE") << endl;
```

Ugnježdeni if

```
if (uslov1)
{
naredba ili blok naredbi;
}
else
{
    if (uslov2)
        {
            naredba ili blok naredbi;
        }
    else
        {
            naredba ili blok naredbi;
        }
}
```

često se zapisuje ovako

```
if (uslov1)
    cout << "nesto" << endl;
else if (uslov2)
    cout << "nesto" << endl;
else
    cout << "nesto" << endl;
```

Switch case

- za razliku od višestrukog uslovnog grananja kod koga su uslovi logički izrazi, u naredbi **switch-case** uslov je **celobrojni izraz** ili **celobrojna promenljiva**
- vrednost celobrojnog izraza ili vrednost celobrojne promenljive je ceo broj
- vrednost uslova se upoređuje sa nizom datih celobrojnih konstanti
- ako je vrednost uslova jednaka nekoj od zadatih konstanti, izvršava se blok naredbi koji je pridružen toj konstanti
- po završetku izvršenog bloka, naredba **break** prekida izvršavanje **switch-case** bloka naredbi i nastavlja sa izvršavanjem prve naredbe iza **switch-case** bloka
- ako vrednost uslova nije jednaka nijednoj od ponuđenih konstanti, izvršava se blok naredbi pridružen naredbi **default**
- ako se izostavi naredba **default** program izvršava prvu naredbu iza **switch-case** bloka
- ceo **switch-case** blok zapisuje se između vitičastih zagrada

Dan u nedelji

```
int main()
{
    int broj;
    cout<<"Unesi broj od 1-7"<<endl;
    cin>>broj;
    switch (broj)
    {
        case 1: cout << "Ponedeljak" << endl; break;
        case 2: cout << "Utorak" << endl; break;
        case 3: cout << "Sreda"<< endl; break;
        case 4: cout << "Cetvrtak" << endl; break;
        case 5: cout << "Petak"<< endl; break;
        case 6: cout << "Subota" << endl; break;
        case 7: cout << "Nedelja" << endl; break;
        default: cout << "Nedelja ima 7 dana!" << endl;
    }
    return 0;
}
```

Matematičke funkcije

opisane u biblioteci `cmath`, prihvataju i vraćaju tip podatka `double`

```
#include <iostream>
#include <cmath>
using namespace std;
#define PI 3.14159265
int main(){
double n = 0.0;
cout << "Enter a number: ";
cin >> n;
cout << endl << "abs - returns the absolute value: " << abs(n) << endl;
cout << "acos - returns the angle whose cosine is the specified number:
" << acos(n) * 180 / PI << " degrees."<< endl;
cout << "asin - returns the angle whose sine is the specified number:
" << asin(n) * 180 / PI << " degrees." <<endl;
cout << "atan - returns the angle whose tanges is the specified number:
" << atan(n) * 180 / PI << " degrees." <<endl;
cout << "atan2 - returns the angle whose tangent is the quotient of two
numbers. " << endl << "\tIn this case, other number is -5.0: " << atan2(n,-
0.5) * 180 / PI << " degrees." << endl;
cout << "cos - returns the cosine of the specified angle: " << cos(n) << endl;
```


Matematičke funkcije

```
cout << "sin - returns the sine of the specified angle: " << sin(n) << endl;
cout << "tan - returns the tangent of the specified angle: " << tan(n) << endl;
cout << "cosh - returns the hyperbolic cosine of the specified angle:
    " << cosh(n) << endl;
cout << "sinh - returns the hyperbolic sine of the specified angle:
    " << sinh(n) << endl;
cout << "tanh - returns the hyperbolic tangent of the specified angle:
    " << tanh(n) << endl;
cout << "exp - returns e raised to the specified power: " << exp(n) << endl;
cout << "floor - returns the largest integer less than or " << endl << "equal to
    the specified double-precision floating-point number: " << floor(n) << endl;
cout << "log - returns the natural (base e) logarithm of a specified number:
    " << log(n) << endl;
cout << "sqrt - returns the square root of a specified number:
    " << log(n) << endl;
cout << "pow - returns a specified number raised to the specified
    power." << endl << "\tIn this case, specified power is 5 :
    " << pow(n,5) << endl << endl;
return 0;
}
```

Stepeni, radijani

- ugao u stepenima (0,90,180,360)
- ugao u radijanima (0, $\pi/2$, π , $3\pi/2$, 2π)

- stepeni u radijane

180 stepeni = π radijana

1 stepen = $\pi/180$ radijana

x stepeni = $x \cdot (\pi/180)$ radijana

- radijani u stepene

π radijana = 180 stepeni

1 radijan = $180/\pi$ stepeni

y radijana = $y \cdot (180/\pi)$ stepeni

izrazite -120° u radijanima

$$-120^\circ = -120 \cdot \frac{\pi}{180} \text{ rad} = -\frac{2\pi}{3} \text{ rad}$$

izraziti $11\pi/3$ u stepenima

$$\frac{11\pi}{3} \text{ rad} = \frac{11\pi}{3} \cdot \left(\frac{180}{\pi}\right)^\circ = 11 \cdot 60^\circ = 660^\circ$$