**MATEMATIČKE FUNKCIJE**

PROGRAM kojim korisnik unosi jedan broj, a program računa i ispisuje kvadrat tog broja

 int a;

 cout <<"UNESI BROJ: ";

 cin >> a;

 cout << a\*a<<endl;

 system ("PAUSE"); return 0; }

**pow**

#include<iostream>

**#include <math.h> // ili #include<cmath>**

using namespace std;

int main ( )

{

int a,b,c;

cout<<“Unesi broj“<<endl;

cin>>a>>c;

b=**pow**(a,c);

cout<<“Kvadrat je: “<<b<<endl;

system ("PAUSE");

return 0; }

MATEMATIČKE FUNKCIJE

**pow - potencija**

**sqrt – kvadratni korijen**

**ceil – zaokruživanje na cijeli veći**

**floor - zaokruživanje na cijeli manji**

**trunc – izrezivanje decimala**

**div –**

div 12:5=2

 2

mod

**mod -**

1. Unesi jedan decimalan broj, a program neka izračuna i ispiše najbliži manji cijeli broj (floor).
2. Unesi jedan decimalan broj, a program neka izračuna i ispiše taj isti broj bez decimala (trunc).
3. Unesi jedan cijeli broj, a program neka izračuna i ispiše kvadrat tog broja, zatim kub tog broja, taj broj na četvrtu potenciju a zatim korijen tog broja.
4. Izračunaj i ispiši opseg kružnice (2rPI).
5. Unesi jedan realni broj. Izračunaj i ispiši kvadrat unesenog broja, zaokruži realni broj na najbliži cijeli veći broj, zaokruži na najbliži manji broj, rastavi broj na njegov cijeli i decimalni dio.

## Unarni aritmetički operatori

Unarni operatori djeluju samo na jednu varijablu. Postoje sljedeći unarni operatori:

* za promjenu predznaka,
* za uvećavanje (inkrementiranje),
* za umanjivanje (dekrementiranje).

**– a** Unarni minus (mijenja predznak broja)

**a++** Operator za uvećavanje (uvećava broj za 1 nakon što se varijabla dobavi iz memorije)

**a – –** Operator za umanjivanje (umanjuje broj za 1 nakon što se varijabla dobavi iz memorije)

**++a** Operator za uvećavanje (uvećava broj za 1 prije nego se varijabla dobavi iz memorije)

**– – a** Operator za umanjivanje (umanjuje broj za 1 prije nego se varijabla dobavi iz memorije)

Operatori obnavljajućeg pridruživanja omogućavaju kraći zapis aritmetičkih izraza. Sastoje se od znaka jednakosti i odgovarajućeg aritmetičkog operatora. Npr. izraz a=a+8, može se zapisati kao a+=8.

Neki od operatora obnavljajućeg pridruživanja:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| + = | – = | \* = | / = | % = |
| a = a + .... | a = a – .... | a = a \* .... | a = a / .... | a = a % .... |

Pri uporabi unarnih operatora važno je paziti na položaj operatora (da li se nalazi prije ili poslije varijable) jer se njegovo djelovanje u jednom i drugom slučaju razlikuje. Ako je operator ispred varijable (npr. ++a) tada se u izrazu računa s uvećanom/umanjenom vrijednošću varijable. U primjeru

a=1;

b=++a +5;

po izvršenju naredbi, sadržaj varijable **b**=7. (Prvo se poveća vrijednost varijable **a** za 1 a zatim se računa vrijednost izraza.) Ako je operator iza varijable (npr. a++) tada se vrijednost varijable uveća/umanji tek nakon izračunavanja izraza. U primjeru

a=1;

b=a++ +5;

po izvršenju naredbi, sadržaj varijable **b=6**. (Prvo se računa vrijednost izraza, a nakon toga se povećava vrijednost varijable **a** za 1.)

**Zadatak 10 (operatori obnavljajućeg pridruživanja):** Potrebno je unijeti realni broj i pridružiti ga varijabli a. Sadržaj varijable a prvo treba uvećati za 5, pa umanjiti za 8, na kraju pomnožiti sa 3. Koristiti operatore obnavljajućeg pridruživanja**.** Ispis neka bude oblika:

Upisi zeljeni broj:

Sadrzaj varijable a se uvecava za 5. Sada a iznosi: ....

Od trenutnog sadrzaja varijable a se oduzima 8. Sada a iznosi: ....

Trenutni sadrzaj varijable a se mnozi sa 3. Sada a iznosi: ....

#include<iostream.h>

int main()

{

float a;

 cout<<"Upisi zeljeni broj:";

 cin>>a;

 cout<<"Sadrzaj varijable a se uvecava za 5. Sada a \

 iznosi:"<<(a+=5)<<endl;

 cout<<"Od trenutnog sadrzaja varijable a se oduzima 8.\

 Sada a iznosi:"<<(a-=8)<<endl;

 cout<<"Trenutni sadrzaj varijable a se mnozi sa 3.\

 Sada a iznosi:"<<(a\*=3)<<endl;

 System(pause);

 return 0;

}