

Zadatak 1: Treba unijeti realni broj pa provjeriti da li je ≥ 0 . Ako jeste, računa se kvadratni korijen tog broja, a ako nije ispisuje se poruka: „**Broj je manji od 0, ponovi upis**”. i ponavlja se unos broja.

```
#include<iostream>
#include<cmath>
using namespace std;
int main(){
    double a;
    upis:cout<<"Unesite realan broj: ";
    cin>>a;
    if(a>=0)
        cout<<"Kvadratni korijen realnog broja | "<<a<<" je: "<<sqrt(a)<<endl;
    else{
        cout<<"Broj je manji od 0, ponovi upis. "<<endl;
        goto upis;
    }
    return 0;
}
```

Biblioteku **cmath** smo morali koristiti radi kvadratnog korijena. Osim toga koristili smo i naredbu **goto** za slučaj da se unese negativan broj kako bi program od korisnika tražio ponovno unošenje broj. To radimo iz jednostavnog razloga jer kada računamo kvadratni korijen potkorijena veličina mora biti pozitivna ili nula.

Zadatak 2: Napisati program za određivanje funkcije: $y = \begin{cases} x - 1 & \text{za } x < 3 \\ x + 3 & \text{za } x \geq 3 \end{cases}$. Ova funkcija je kombinacija dvije linearne funkcije. Korisnik treba da unese vrijednost x , od vrijednosti x -a će zavisiti vrijednost funkcije y .

Rješenje je dosta jednostavno:

```
1  #include<iostream>
2  #include<cmath>
3  using namespace std;
4  int main(){
5      double x,y;
6      cout<<"Unesite varijablu x: ";
7      cin>>x;
8      if(x<3)
9          y=x-1;
10     else
11         y=x+3;
12     cout<<"Vrijednost izraza y za x="<<x<<" je "<<y<<endl;
13     return 0;
14 }
```

Pogledajmo testne primjere:

```
C:\Users\kadic\Desktop\struktura_grananja_zadaci\zadatak2.exe
Unesite varijablu x: 5
Vrijednost izraza y za x=5 je 8
Process returned 0 (0x0)   execution time : 3.135 s
Press any key to continue.
```

Kada je korisnik unio 5 izvršava se samo dio if else strukture i to naj u liniji 13 koda tj u bloku **else** jer 5 nije manje od 3.

```
C:\Users\kadic\Desktop\struktura_grananja_zadaci\zadatak2.exe
Unesite varijablu x: 0
Vrijednost izraza y za x=0 je -1
Process returned 0 (0x0)   execution time : 3.273 s
Press any key to continue.
```

Pošto je unesena 0 u ovom slučaju je izvršen blok unutar **if** u redu pod brojem 9.

Zadatak 3: Treba unijeti cijeli broj pa provjeriti da li je <0. Ako jeste, broj treba postati apsolutnu vrijednost unesenog broja . Potom, treba provjeriti da li je broj paran. Ako je paran treba izračunati i ispisati njegovu treću potenciju, a ako nije treba ispisati: Broj je neparan.

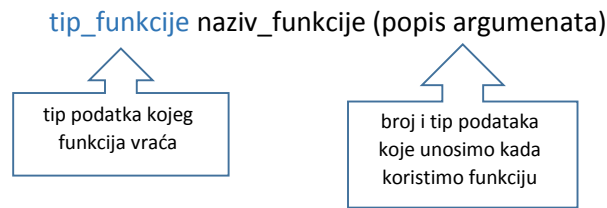
(Napomena: obratite pažnju kako se koriste funkcije **abs** za računanje apsolutne vrijednosti i **pow** za računanje stepena)

```
1  #include<iostream>
2  #include<cmath>
3  using namespace std;
4  int main(){
5      int x;
6      cout<<"Unesite cijeli broj: ";
7      cin>>x;
8      //ovdje koristimo samo jednostruku strukturu grananja
9      if(x<0){
10         x=abs(x);
11     }
12     //ovdje dvostruku strukturu grananja
13     if(x%2==0){
14         cout<<"Trecu potencija broja je "<<pow(x,3)<<endl;
15     }
16     else{
17         cout<<"Broj je neparan."<<endl;
18     }
19     return 0;
20 }
```

Primjetite da kada računamo izraz x^3 u cpp-u koristimo izraz `pow(x, 3)`. Funkcija **pow** prima kao parametre dva realna parametra: bazu i eksponent stepena a kao rezultat vraća realan broj jednak vrijednosti tog stepena. Funkcija **abs** prima jedan realan parametar i kao rezultat vraća apsolutnu vrijednost ponovo u obliku realnog broja.

Recimo nešto o korištenju gotovih funkcija:

Deklaracija funkcije



Vezano za tip funkcije koji vraća trena istaći da on može biti bilo koji tip koji smo do sada radili isto tako može biti da funkcija ne vraća nikakvu vrijednost (umjesto tipa bi pisali **void**). Isto tako u popisu argumenata može da ne bude niti jedan argumenat, da bude jedan, dva, (funkcija **main** koju često koristimo nema argumenata tada samo pišemo zagrade).

Korisniku na raspolaganju stoji mnoštvo funkcija razvrstanih u odgovarajuće biblioteke. Većina matematičkih funkcija deklarirana je u biblioteci **cmath**. Biblioteka **cmath** se u program uključuje preprocesorskom naredbom **#include**.

U nastavku imate nekoliko osnovnih funkcija biblioteke **cmath** kao i njihov način korištenja i funkciju koju obavljaju.

Matematičke funkcije

<i>oblik funkcije</i>	<i>zadatak koji obavlja funkcija</i>
<i>float sqrt (float x)</i>	Kvadratni korijen argumenta x.
<i>float abs (float x)</i>	Apsolutna vrijednost broja x.
<i>float floor (float x)</i>	Zaokružuje argument x na najbliži manji cijeli broj.
<i>float ceil (float x)</i>	Zaokružuje argument x na najbliži veći cijeli broj.
<i>float pow (float baza, float eksponent)</i>	Računa stepen: baza ^{eksponent} .
<i>float fmod (float djeljenik, float djelitelj)</i>	Računa ostatak dijeljenja dva realna broja.
<i>atan(x)</i>	Arkus tangens, računa ugao u radijanima.
<i>sin (x)</i>	Računa sinus <i>ugla</i> zadanog u radijanima.

DZ1: Napraviti program koji ispituje koji je od dva unesena broja veći.

(Napomena: Ako je prvi broj \geq od drugog broja treba to ispisati. Isto tako ako je $<$ od drugog opet to treba ispisati)

Testni primjer:

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz1.exe
Unesite dva broja: 4
Prvi broj je manji od drugog.
Process returned 0 (0x0) execution time : 4.115 s
Press any key to continue.
```

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz1.exe
Unesite dva broja: 9
9
Prvi broj je veci ili jednak od drugog.
Process returned 0 (0x0)   execution time : 2.793 s
Press any key to continue.
```

DZ2: Napisati program za određivanje y po formuli: $y = \begin{cases} 2x - 1 & \text{za } x < 0 \\ 4x + 9 & \text{za } x \geq 0 \end{cases}$.

(Napomena: Pogledati Zadatak 2)

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz2.exe
Unesite vrijednost x: -3
-3
Za uneseno x=-3 vrijednost y je -7
Process returned 0 (0x0)   execution time : 2.126 s
Press any key to continue.
```

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz2.exe
Unesite vrijednost x: 5
5
Za uneseno x=5 vrijednost y je 29
Process returned 0 (0x0)   execution time : 1.906 s
Press any key to continue.
```

DZ3: Učitati realan broj, ako je broj pozitivan ili 0 ispisati njegov korijen, a ako je negativan, njegov kvadrat.

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz3.exe
Unesite realan broj: -5
-5
Broj -5 je negativan i njegov kvadrat je 25
Process returned 0 (0x0)   execution time : 3.590 s
Press any key to continue.
```

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz3.exe
Unesite realan broj: 256
256
Broj 256 je pozitivan i njegov korijen je 16
Process returned 0 (0x0)   execution time : 4.761 s
Press any key to continue.
```

DZ4: Učitati dva realna broja (a, b) a zatim izračunati i ispisati z prema izrazu: $z = \begin{cases} 2 * a + b & \text{za } a \geq b \\ a - 2 * b & \text{za } a < b \end{cases}$

DZ5: Učitati dva realna broja (a, b) a zatim izračunati i ispisati y prema izrazu:

$$z = \begin{cases} a + b, & \text{za } a \text{ neparno i } b \text{ neparno} \\ a * b, & \text{za sve ostale slučajeve} \end{cases}$$

Zadatak 4: Napisat program u koji unesemo broj godina a on nam vrati kao rezultat dali smo beba, dijete, mlad covjek ili star covjek.

(**Napomena:** ako neko ima između 0 i 3 godine njega smatramo bebom, između 3 i 18 godina su djeca, čovjek je mlad između 18 i 40 godina, od 40 do 100 godina je star čovjek, ako neko ima manje od 0 ili više od 100 godina prepuštam vama da prepravite program ☺. Koristimo višestruko grananje)

```
1  #include<iostream>
2  using namespace std;
3  int main(){
4      int godina;
5      cout<<"Unesite koliko imate godina: ";
6      cin>>godina;
7      if(godina>=0&&godina<3)
8          cout<<"Vi ste beba!"<<endl;
9      else if(godina>=3&&godina<18)
10         cout<<"Vi ste dijete!"<<endl;
11     else if(godina>=18&&godina<40)
12         cout<<"Vi ste mlad covjek!"<<endl;
13     else if(godina>=40&&godina<100)
14         cout<<"Vi ste ostarili!"<<endl;
15     else
16         cout<<"Ili se niste jos rodili ili ste Japanac !"<<endl;
17
18     return 0;
19 }
```

Na ovom primjeru smo koristili logički operator && „čita se i“ dakle mi podrazumjevamo da je beba starija od 0 godina a da ima manje od tri godine, isl. Primjetite da nakon zadnjeg **else** nema **if**. Šta to znači? To znači ako unesemo bilo šta osim godina od 1 do 99 program će ispisati Ili ste japanac ili se niste rodili. Obratite pažnju i na korištenje >= i <= operatora. Tako sam na primjer želio da osobe sa 18 godina pripadaju „mladima“ a ne „djeci“. Dakle logičko "i" odnosno && znači da oba uslova trebaju biti zadovoljena da bi godine pripadale tom intervalu.

DZ6: Napisat program koji omogućava korisniku da vidi plate zaposlenika te firme. Recimo neka su zaposlenici oblika: portir, kuhar, službenik, član uprave i direktor. U zavisnosti da li smo unijeli 1, 2, 3, 4 ili 5 program treba da nam vrati kolika je kome od uposlenika plata. Ako smo unijeli 3 program treba da nam ispiše kolika je plata kod službenika, isl. Sami odredite visinu plate radnika.

(Napomena: Za poređenje koristite operator ==. Dakle ako korisnik unese varijablu *unos* mi moramo u uslovu strukture grananja ispitivat na nacin `if(unos==3)`. Mozete probati riješiti zadatak koristeći **switch-case**)

Testni primjeri:

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz6.exe
Unesete odgovarajuci broj od 1 do 5 redom za plate uposlenika:
portir, kuhar, sluzbenik, clan uprave i direktor:
5
Direktor ima platu od: 1500 KM.
Process returned 0 (0x0) execution time : 6.109 s
Press any key to continue.
```

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz6.exe
Unesete odgovarajuci broj od 1 do 5 redom za plate uposlenika:
portir, kuhar, sluzbenik, clan uprave i direktor:
1
Portir ima platu od: 630 KM.
Process returned 0 (0x0)   execution time : 1.878 s
Press any key to continue.
```

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz6.exe
Unesete odgovarajuci broj od 1 do 5 redom za plate uposlenika:
portir, kuhar, sluzbenik, clan uprave i direktor:
4
Clan uprave ima platu od: 1200 KM.
Process returned 0 (0x0)   execution time : 2.522 s
Press any key to continue.
```

DZ7: Pretpostavimo da imamo samo 5 predmeta u školi. Napisat program koji traži da unesemo zaključne ocjene iz tih predmeta a zatim u ovisnosti od ocjena ispisuje naš uspjeh (odličan, vrlodobar,...). Ako je bilo koji od predmeta jedinica program treba da ispiše da smo pali (koristeći operator „ili“).

Testni primjeri:

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz7.exe
Unesite zakljucne ocjene iz pet predmeta:
N
N
N
N
N
Vas uspjeh je dobar!
Process returned 0 (0x0)   execution time : 7.785 s
Press any key to continue.
```

```
C:\Users\kadici\Desktop\struktura_grananja_zadaci\dz7.exe
Unesite zakljucne ocjene iz pet predmeta:
1
N
N
N
N
Pali ste!
Process returned 0 (0x0)   execution time : 10.024 s
Press any key to continue.
```