

C++ Riyazi funksiyalar (C++ Math functions)

- C++ dilində bir çox hazır funksiyalar var ki, biz onlardan istifadə etməklə əsas riyazi əməliyyatları yerinə yetirə bilərik.

max və min funksiyaları

- **max(x, y)** funksiyası **x** və **y** dəyişənlərinin maksimumunu (ən böyüyünü) tapmaq üçün istifadə edilir.
- **min(x, y)** funksiyası **x** və **y** dəyişənlərinin minimumunu (ən kiçiyini) tapmaq üçün istifadə edilir.

```
#include <iostream>
using namespace std;
int main()
{
    int x = 2, y = 3;
    cout<<max(x,y)<<endl;
    cout<<min(x,y)<<endl;
}
```

3
2

```
#include <iostream>
using namespace std;
int main()
{
    double x = 3.5, y = 3.7;
    cout<<max(x,y)<<endl;
    cout<<min(x,y)<<endl;
}
```

3.7
3.5

- İkidən çox sayda dəyişənlər arasında da minimumu və ya maksimumu bu funksiyalardan istifadə etməklə tapmaq mümkündür.

```
#include <iostream>
using namespace std;
int main()
{
    int a = 2, b = 3, c = 4;
    cout<<max(a, max(b, c))<<endl;
    cout<<min(a, min(b, c))<<endl;
}
```

4
2

```
#include <iostream>
using namespace std;
int main()
{
    int a = 2, b = 3, c = 4, d = 5;
    cout<<max(max(a, b), max(c, d))<<endl;
    cout<<min(min(a, b), min(c, d))<<endl;
}
```

5
2

C++ **swap** funksiyası

- **swap(a, b)** funksiyası **a** və **b** dəyişənləri arasında dəyərlərinin dəyişdirilməsi üçün istifadə olunur, yəni ki, **a**-nın dəyəri **b**-yə, **b**-nin dəyəri isə **a**-ya verilir.

```
#include <iostream>
using namespace std;
int main()
{
    int a = 5, b = 10;
    swap(a, b);
    cout<<a<<" "<<b<<endl;
}
```

10 5

C++ **<cmath>** kitabxanası

- Elə funksiyalar var ki, biz onları istifadə edərkən **<cmath>** başlığını əlavə etməliyik.
- **pow(a, b)** – (power-qüvvət) funksiyası a^b -ni hesablamaq üçündür.
 - **sqrt(a)** – (square root – kvadrat kök) funksiyası $\sqrt{a}$ -ni hesablamaq üçündür.
 - **cbrt(a)** – (cube root – kub kök) funksiyası $\sqrt[3]{a}$ -ni hesablamaq üçündür.
 - **abs(a)** – (absolute – mütləq qiymət və ya modul) funksiyası **a**-nın qiymətini müsbətə çevirmək üçün istifadə edilir.
 - **round(a)** – (round – yuvarlaqlaşdırmaq) funksiyası **a**-nı ən yaxın tam ədədə yuvarlaqlaşdırmaq üçün istifadə edilir.
 - **ceil(a)** – (ceil – tavan) funksiyası **a**-nı yuxarıya doğru ən yaxın tam ədədə yuvarlaqlaşdırmaq üçün istifadə edilir.
 - **floor(a)** – (floor – döşəmə) funksiyası **a**-nı aşağıya doğru ən yaxın tam ədədə yuvarlaqlaşdırmaq üçün istifadə edilir.
 - **sin(a)** – (sin – sinus) funksiyası **a**-nın sinusunu hesablayır.
 - **cos(a)** – (cos – kosinus) funksiyası **a**-nın kosinusunu hesablayır.

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    int a = 2, b = 3;
    cout<<pow(a, b)<<endl;
    cout<<pow(4, 3)<<endl;
    cout<<pow(2, 60)<<endl;
    cout<<(long long)pow(2, 60)<<endl;
}
```

8
64
1.15292e+18
1152921504606846976

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    int a = 36;
    cout<<sqrt(a)<<endl;
    cout<<sqrt(25)<<endl;
    cout<<sqrt(30)<<endl;
}
```

6
5
5.47723

<pre>#include <iostream> #include <cmath> using namespace std; int main() { int a = 27; cout<<cbrr(a)<<endl; cout<<cbrr(8)<<endl; cout<<cbrr(100)<<endl; }</pre>	3 2 4.64159
--	-------------------

<pre>#include <iostream> #include <cmath> using namespace std; int main() { int a = -6; cout<<abs(a)<<endl; cout<<abs(-7.7)<<endl; cout<<abs(8)<<endl; }</pre>	6 7.7 8
--	---------------

<pre>#include <iostream> #include <cmath> using namespace std; int main() { double x = 3.6; cout<<round(x)<<endl; cout<<ceil(x)<<endl; cout<<floor(x)<<endl; }</pre>	4 4 3
--	-------------

<pre>#include <iostream> #include <cmath> using namespace std; int main() { cout<<ceil(3.0)<<endl; cout<<floor(3.0)<<endl; cout<<ceil(-2.6)<<endl; cout<<floor(-2.6)<<endl; }</pre>	3 3 -2 -3
---	--------------------

<pre>#include <iostream> #include <cmath> using namespace std; int main() { cout<<sin(1)<<endl; cout<<cos(1)<<endl; }</pre>	0.841471 0.540302
---	----------------------

C++ `cout.precision` və `fixed` əməlləri

- `double` tipli ədədləri çıxışa verərkən onluqdan sonra istənilən rəqəmə qədər yuvarlaqlaşdırmaq üçün `cout.precision` və `fixed` əməllərindən istifadə olunur.
- Qeyd: Onluqdan sonra maksimum 15 sayda rəqəmə qədər yuvarlaqlaşdırmaq mümkündür.

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    double x = 5.29182;
    cout.precision(3);
    cout<<fixed<<x<<endl;
}
```

5.292

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    double x = sqrt(30);
    cout.precision(10);
    cout<<fixed<<x<<endl;
}
```

5.4772255751

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    double a = 2, b = 3;
    cout.precision(5);
    cout<<fixed<<a/b<<endl;
}
```

0.66667

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    double a = 2.123456789;
    cout.precision(10);
    cout<<fixed<<a<<endl;
}
```

2.1234567890