Maths Functions

This lesson introduces the cmath library and the functionality that it provides in C++ such as square root, power, and trigonometry

We'll cover the following

C++ math library

The **C++** math library is actually **C's** math library. It is easy to use and is accessed by including `cmath`.

```
#include <cmath>
```

Now that we have the **C math** library let's use some neat functions.

Square Root

The function `sqrt` is used to compute the **square root** of a number. It takes only **one** *argument*, the number, whose **square root** needs to be calculated.

```
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    float myFloat = 7.4f; //the f (requires decimal) tells the compiler to treat this real number a
    //and not as a 64 bit double. this is more of a force of habit than a requirement
    cout << "The square root of " << myFloat << " is " << sqrt(myFloat) << endl;


    return 0;
}
```

Powers

The function `pow` is used to calculate the **power** of a number in C++. It takes as its *first argument* the number itself and the value to which it needs to be raised is the *second argument*.

```
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    float myFloat = 5.7f;
    // we use the pow function to compute the powers of the integer values given
    cout << myFloat << " in the power of 2 is " << pow(myFloat, 2) << endl;
    cout << myFloat << " in the power of 3 is " << pow(myFloat, 3) << endl;
    cout << myFloat << " in the power of 0.5 is " << pow(myFloat, 0.5) << endl;


    return 0;
}
```

Trigonometry

To perform *trigonometric* operations, the `cmath` library provides the functions `sin` , `cos` and `tan` . Each of the *three* takes only **one** *argument*, the number, on which these operations need to be applied.

> **Note:** Trigonometric functions in `cmath` use *RADIANS*.

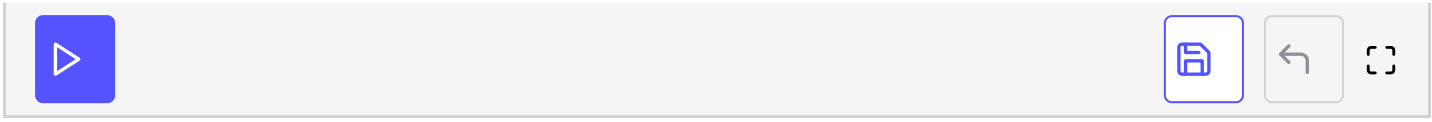
```
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    float myFloat = 5.7f;

    // here the sin, cos and tan functions are being used to compute the trignometric values
    cout << "sin(" << myFloat << ") = " << sin(myFloat)<<endl;
    cout << "cos(" << myFloat << ") = " << cos(myFloat) << endl;
    cout << "tan(" << myFloat << ") = " << tan(myFloat) << endl;
```

```
    return 0;
}
```



Well, now that you're familiar with the sorts of operations that can be performed in C++ it's time to learn about new concepts such as loops in C++. Read on to next chapter to find out more.