



## Table of Contents

<a href="#">Introduction.....</a>	<a href="#">1</a>
<a href="#">Math operators.....</a>	<a href="#">1</a>
<a href="#">Square root and exponents.....</a>	<a href="#">2</a>
<a href="#">Math that your program just can't even.....</a>	<a href="#">3</a>

## Introduction

We can do math operations with programming languages. This is especially useful for tedious or large calculations. C++, C#, and Java has the standard math operators built in, like +, -, \*, and /, and they also has extended code libraries for more advanced math, like using square roots or powers.

## Math operators

Addition	Subtraction	Multiplication	Division	Modulus
+	-	*	/	%

You can also use parenthesis ( ) in programming similar to how you would in algebra – to explicitly specify the order of operations of your math calculations.

Remember that when declaring a variable, the variable's name goes on the LHS (left-hand side) and the computation goes on the RHS (right-hand side).

```
int perimeter = 2 * width + 2 * length;  
int area = width * height;  
double mph = miles / hour;
```

The modulus operator returns the **remainder** in a division. This can be useful for keeping numbers within a given range.



## Square root and exponents

C++, Java, and C# each have their own math libraries which contain additional math operations that might be useful in certain applications.

For C++, you need to `#include <cmath>` in order to use these functions.

### Square Root

Returns the square root of a value. For example, the square root of 4 is 2.

C++

```
double root = sqrt( 144 );
```

C#

```
double root = Math.Sqrt( 144 );
```

Java

```
double root = Math.sqrt( 144 );
```

### Exponents

Raise some number to some power (  $x^n$  ). For example,  $2^3=8$

C++

```
double power = pow( base, exponent );  
double fiveSquared = pow( 5, 2 );
```

C#

```
double power = Math.pow( base, exponent );  
double fiveSquared = Math.pow( 5, 2 );
```

Java

```
double power = Math.Pow( base, exponent );  
double fiveSquared = Math.Pow( 5, 2 );
```

There are also functions for **trigonometric functions** and other math concepts. This means you can write a program to help you with your trig homework later on. ;)



## Math that your program just can't even

C++, C#, and Java cannot handle really out-of-the-ordinary math operations, like we can in some forms of math. There may be code libraries out there written by people to handle some of these (such as special logic for imaginary numbers).

- If a number goes above the maximum value, it will wrap around to the minimum value.
- If a number is divided by zero, the program will crash!
- If a number is an imaginary number, it will have the value “NaN” or “nan”, which means “Not a number”.
- If a number is just too dang big, it will show up as “Infinity” in Java, and “inf” in C++.