Welcome to the C++ Course!

What is C++?

Created in 1985 by Bjarne Stroustrup, C++ is a general-purpose programming language which expands on the programming language C.
Initially designed for creating software infrastructure for desktop applications, video games, servers etc.

Why learn C++?

C++ is a MUST for students and working professionals to become great Software Engineers. Key advantages of learning C++:

- C++ is very close to the hardware, so you get a chance to work at a low level which gives you a lot of control in terms of memory management, better performance and finally a robust software development.
- C++ is one of the evergreen programming languages and is loved by millions of software developers.
- C++ is the most widely used programming language in application and system programming. So you can choose your area of interest in software development.

What we will look at to begin learning about C++:

Header Files

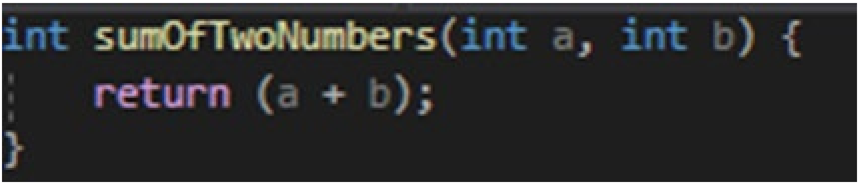
When coding, in some cases for larger projects it can become tiresome to individually declare every function that you want to implement, this can lead to syntax errors occurring.

So that your program runs more efficiently and potentially saves memory space, and provides increased organisation, header files can be used to store the declaration of variables, structures and classes to be implemented in the source code file.

Declared by .h or .hpp as a file type. In whichever IDE (Integrated Development Environment) you use at the start of the source code file, to use the header file, it's initialised via "#include <filename.h>" syntax. This tells the compiler to file the relative file path of the header file and apply the contents when declared in the specified file.

Header File Declaration Example

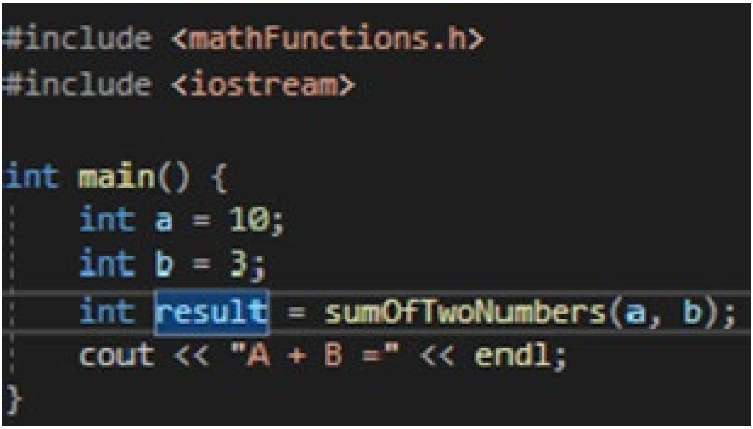
In the figures below the header file is declared at the start of the file, this can include other header files and any library that the developer intends to use such as iostream (Standard Input Output)



```
int sumOfTwoNumbers(int a, int b) {
    return (a + b);
}
```

Figure 1: mathFunctions.h

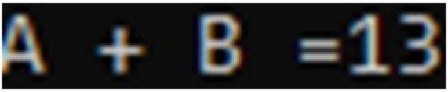
Now in the main function (Figure 2) in the source code file, when the function is called because it was declared at the start and 2 integers (a and b) were passed the compiler recognises that the header file is being used and applies the function to the integer variables.



```
#include <mathFunctions.h>
#include <iostream>

int main() {
    int a = 10;
    int b = 3;
    int result = sumOfTwoNumbers(a, b);
    cout << "A + B =" << endl;
}
```

Figure 2: mathFunctions.h used in source code file

Figure 1 shows the contents of the header file, this is a function that takes 2 different integers as and adds them together before returning the value of the sum.
This is shown in the console output:



```
A + B =13
```

Figure 3: Console Output

C++ Libraries

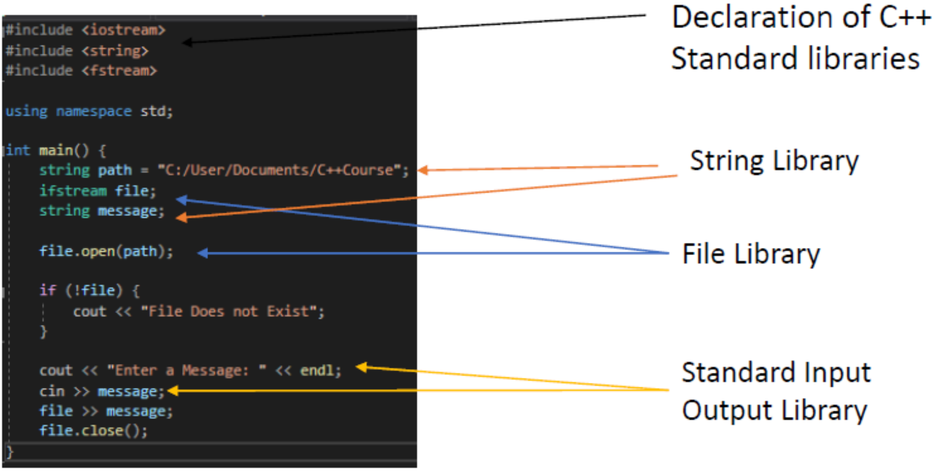
What are they?

Libraries are groups of codes that are intended to be used for their reusability in many programs. They are declared as header files that determine the functionality and which library to implement into the code.

How are they used?

The most commonly used library in C++ is the C++ standard library this is a collection of multiple features such as classes functions, constants etc, different parts of this library vary in size so it’s broken down into multiple parts which means multiple declarations for if a developer wants to use a certain part of the library.

Example of usage



```
#include <iostream>
#include <string>
#include <fstream>

using namespace std;

int main() {
    string path = "C:/User/Documents/C++Course";
    ifstream file;
    string message;

    file.open(path);

    if (!file) {
        cout << "File Does not Exist";
    }

    cout << "Enter a Message: " << endl;
    cin >> message;
    file >> message;
    file.close();
}
```

Declaration of C++ Standard libraries
String Library
File Library
Standard Input Output Library

C++ User Input and Output

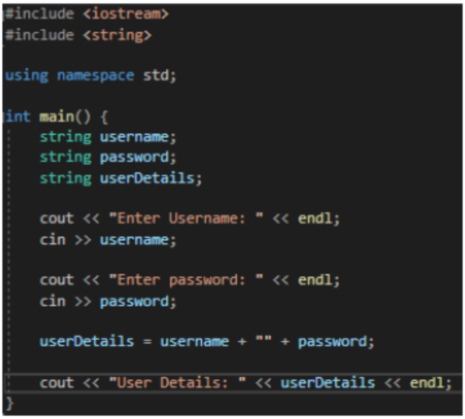
What is User input/output?

User input and output are declared for usage by the iostream header aka standard input-output library. It takes input from the user assigns it to a variable and when used outputs to the console or another form of display such as an application executable.

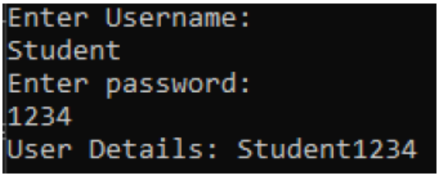
How is it used?

Output is defined by ‘Cout’ followed by ‘<< ’ (2 left angular braces) displays the following element(s) in the console. Input is defined by ‘Cin’ followed by ‘>>’ (2 right angular braces), takes the next line entered by the user and stores it in a user-defined variable (i.e. array, string, int, vector)

Example of its usage



```cpp
#include <iostream>
#include <string>

using namespace std;

int main() {
    string username;
    string password;
    string userDetails;

    cout << "Enter Username: " << endl;
    cin >> username;

    cout << "Enter password: " << endl;
    cin >> password;

    userDetails = username + "" + password;

    cout << "User Details: " << userDetails << endl;
}
```



```
Enter Username:
Student
Enter password:
1234
User Details: Student1234
```

The program takes the user input, assigns it to a string then re assigns it to a string which then can be used later when identifying the user details of a certain user

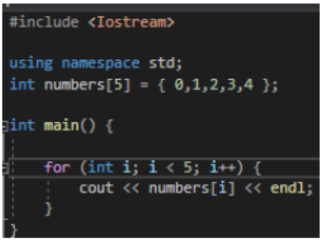
C++ Conditionals and Loops

What are they?

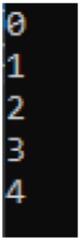
They consist of if statements, while, do-while and for loops. These statements/loops take a true or false condition which the user can define and then execute code based on the set condition until that condition is met.

How are they used?

For Loop:



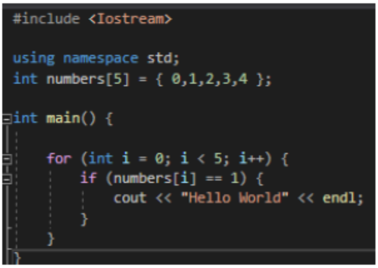
```cpp
#include <Iostream>

using namespace std;
int numbers[5] = { 0,1,2,3,4 };

int main() {

    for (int i; i < 5; i++) {
        cout << numbers[i] << endl;
    }
}
```



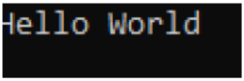
```
0
1
2
3
4
```

Takes a starting point to iterate from, limit to iterate until and either pre or post incrementation/deincrementation. Helpful to print out every element of an array or characters of a string.

If Statements:



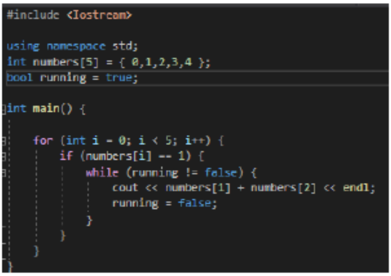
```
#include <Iostream>

using namespace std;
int numbers[5] = { 0,1,2,3,4 };

int main() {

    for (int i = 0; i < 5; i++) {
        if (numbers[i] == 1) {
            cout << "Hello World" << endl;
        }
    }
}
```



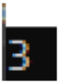
```
Hello World
```

The statement returns true if the number in the array called ‘numbers’ is equal to one. This then executes the below code, displaying a message to the console

While Loop:



```
#include <Iostream>

using namespace std;
int numbers[5] = { 0,1,2,3,4 };
bool running = true;

int main() {

    for (int i = 0; i < 5; i++) {
        if (numbers[i] == 1) {
            while (running != false) {
                cout << numbers[1] + numbers[2] << endl;
                running = false;
            }
        }
    }
}
```



```
3
```

This loop executes under the condition that that the Boolean variable running is not false, once it has passed a sum of the numbers index(position) 1 and 2 (2nd and 3rd value) in the array. The loop exists once running is false