

Hello, C++

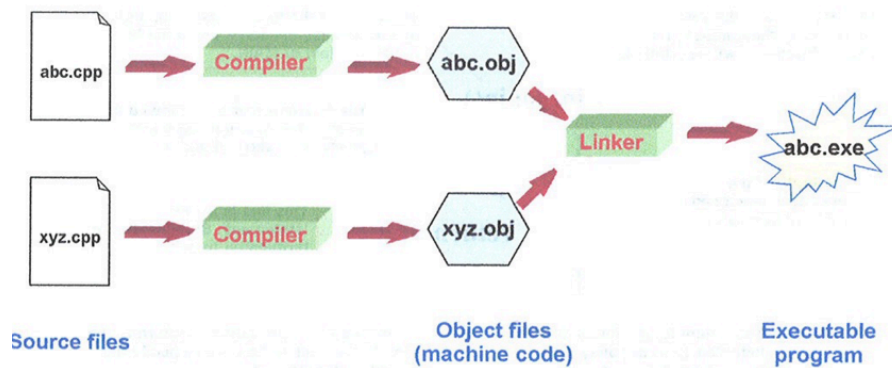
What is C++

- A general purpose programming language
- Originally based on C
- Supports multiple programming paradigms
 - procedural
 - object oriented
 - generic
 - functional
- Developed and implemented by Bjarne Stroustrup



Writing, Building, Running a C++ Program

- C++ Programs normally consist of a number of "modules", which are individually compiled and then linked to form an executable program



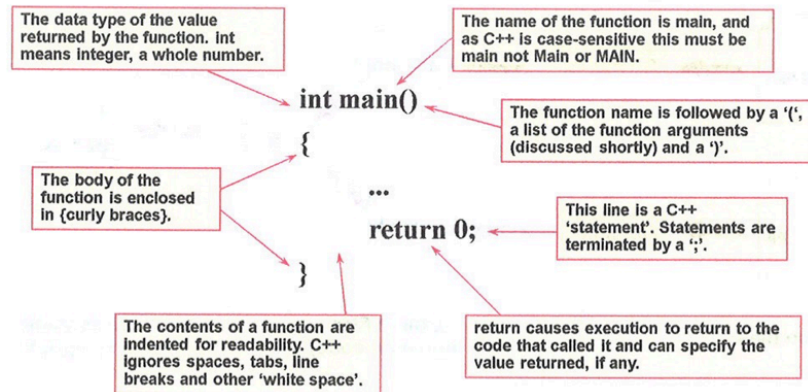
Inside main

- When a C++ program is run, the operating system uses the function `main()` as an entry point

```
#include <iostream>
using namespace std;
int main()
{
    cout << "Hello C++\n";
    return 0;
}
```

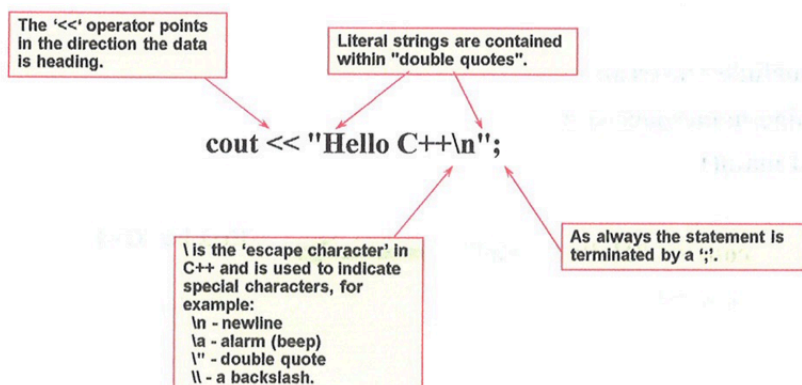
Inside main()

- The main() function is called by the system runtime
 - Should return – to indicate successful completion



Console Output

- Standard output is represented by the C++ object `cout`
 - The `<<` operator is used to send data to `cout`



Housekeeping

- Some additional code is required to allow compilation to succeed

```
#include <iostream>
using namespace std;
```

cout is a member of the iostream library. To use it we need to include iostream, the file which tells the compiler about the contents of this library. #include is a 'preprocessor directive' (the # implies this), as it indicates that the contents of this file are included before the true compilation takes place.

Without the using statement each name must be qualified, e.g. std::cout
A single name from a namespace can also be declared, for example:
using std::cout;

Parts of a C++ program can be contained within 'namespaces'. These allow the names of functions and objects to be unique relative to their namespace, without conflicting with names used by other programmers in their own namespaces. The iostream objects are written within a namespace called 'std', so the using statement allows their names to be seen.

Variables

- Named memory locations
 - Associated with a type, must be declared with their type

```
cout << "Hello C++\n";
int number;
cout << "Enter the number to be multiplied by 2: ";
cin >> number;
cout << number << " times 2 is " << number * 2;
```

cin is the console input object. Notice the direction of the >>.

This defines a single variable called number - don't forget the ';'.

You can define several variables of one type in a single statement such as
int number1, number2, x, y, z;

Several items can be passed to cout by chaining them as shown.

```
Hello C++
Enter the number to be multiplied by 2: 5
5 times 2 is 10
```

Basic Integer Types

Name	Defined size	Windows size	Range of values
int	system dependent	32 bits	+/- 2Gi
short	>= 16 bits	16 bits	+/- 32Ki
long	>= 32 bits	32 bits	+/- 2Gi
long long	>= 64 bits	64 bits	+/- approx 10^{19}
char	ASCII/ANSI symbol	8 bits	-128 to +127

- Use unsigned prefix to cause range of values to start at 0
 - No negative numbers
- long long standardised in C++11
 - Available in other earlier versions though

About Types

- C++ allows size and capacity of types to be discovered
 - Aids portability
- Sizeof operator yields the size in bytes of a given type or variable
 - E.g. sizeof(int)
 - Returns value of type size_t, typically a synonym for unsigned int
- Min and Max values for types can be discovered in file limits.h
 - Discussed later

Other Fundamental Data Types

- `bool` represents true/false
 - Supported by keywords `true` and `false`
 - Equivalent to 1 and 0
- Floating point types
 - `float`
 - `double`
 - `long double`
- `wchar_t` is 16 bit char type
 - For Unicode

Declaring and Initialising Variables

- Variables must be declared before use
 - May be declared at any point
- Initialisation is allowed at declaration
 - Otherwise initial value is determined by other factors (see later)

e means exponent, so this number is 1.7×10^7 or 17 000 000.

```
int a = 0;
double b = 1.7e7;
float f = 2.71828f;
```

Numbers beginning with 0 are considered octal or base 8. Numbers beginning 0x are considered hexadecimal or base 16.

```
short c = 027;
long d = 0x14E;
unsigned long e = 4ul;
```

Suffixes on numbers can be used to explicitly specify their type. Here `f` means float, `ul` means unsigned long. These are seldom necessary.

Strings

- Strings may be represented in different ways
 - Character arrays: "C Style Strings"
 - Standard library string class

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string s;
    cout << "What is your name: ";
    cin >> s;
    cout << "Hello " << s << "\n";
    return 0;
}
```

To use string the file which defines it must be included. Such files are known as 'header files'.

cin >> can be used to read in a string, but input is terminated by any white space character such as a space or enter.

What is your name: Arthur
Hello Arthur