Introduction to C++

Dr. Jing, Yaping

CS270 – Data Structures Using C++

Why Programming

C++ Pioneer

- C++ was developed by Bjarne Stroustrup at Bell Laboratories

C++ Pioneer

- C++ was developed by Bjarne Stroustrup at Bell Laboratories

  - Originally called “C with classes”

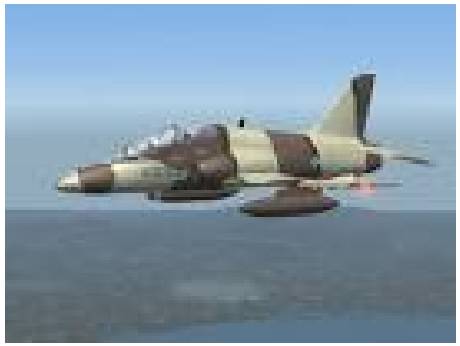
C++ Pioneer

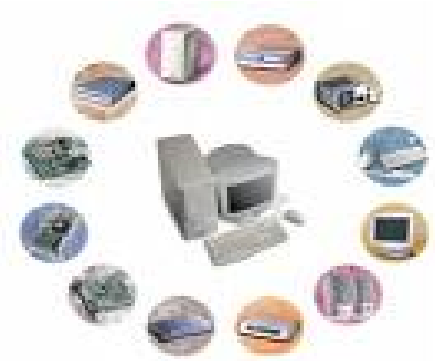
- C++ was developed by Bjarne Stroustrup at Bell Laboratories

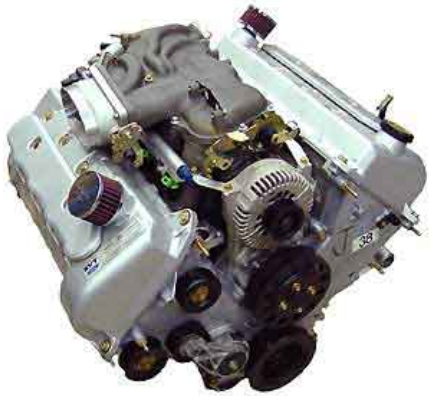
  - Originally called “C with classes”
  - The increment operator (++) of the C++ Indicates that C++ is an enhanced version of C

Why C++

- C++ is the most widely used language in engineering areas







Why C++ (Cont')

- C++ is ISO standard, which is universally accepted
- C++ is available on nearly all kinds of computers
- Programming concepts you learn in C++ can be applied in other languages, including Java, C#, etc.

After the Course, you'll be able to

- Write small C++ programs

After the Course, you'll be able to

- Write small C++ programs
- Read much larger programs

After the Course, you'll be able to

- Write small C++ programs
- Read much larger programs
- Learn the basics of other languages by yourself

After the Course, you'll be able to

- Write small C++ programs
- Read much larger programs
- Learn the basics of other languages by yourself
- Take advanced programming language course

After the Course, you'll not yet be

- An expert programmer or C++ language expert

Our First C++ Program

- Open a text editor such as Sublime

Our First C++ Program

Type in the following source code.

```
// directive for input and output
#include <iostream>

int main ()
{
  std::cout<<"Hello World!" <<std::endl;

  return 0;
}
```

// Save the source file as: HelloWorld.cpp

How to Compile

- Open a Console Terminal
- Type:
  g++ -o out HelloWorld.cpp
- Type:
  ls -la to see if the new file ”out” is produced

How to Execute/Run C++ Binary File

- Type:
  ./out where the symbol ./ means the current directory
- Check Results!

How to learn C++

The best way to learn a programming language is to try writing programs and test them on a computer!

Supporting Tools for Writing and Testing a C++ Program

- a text editor such as Sublime Text, or pico, vi, emacs in UNIX, etc.
- a compiler such as g++
- a linker
- a debugger that help diagnose problems

Steps for Writing and Testing C++ Program

- Edit
- Compile (if error, fix the problem)
- Execute (if error, fix the problem)