C++ Primer Plus

Setting Out to C++

- C++ case sensitive
- Use #include<iostream.h>
  - Old compiler
  - Omit using namespace std;
- printf(), scanf()
  - C input and output functions
  - stdio.h
- Constuct C++ programs from functions
- Preprocessor directive #include
- Statement separators
  - Fortran - end of the line
  - Pascal - semicolon
  - C++ - semicolon as terminator
    - Semicolon is part of the statement, rather than separator
- Int main() = int main(void)
  - C++ - takes no arguments
  - C - remaining silent about whether there are arguments
- Void main
  - Omit return statement
  - Not part of C++ standard
- main()
  - Every program must have
  - exceptions

- Dynamic link library module = DDL

  • Code that other Windows programs can use, not stand alone program

- Comments

  • // single line

  • /* … */ multiple line

- #include<iostream> and using name std;

  • Input and output facilities

  • Preprocessor add content of iostream file

  • i = input, o = output

- C++ and C uses a preprocessor

  • Program that processes a source file before the main compilation takes place

- Header filenames

  • = include files

  • iostream, math

  • h extension

    - reserved for old C header files

    - C++ files have no extension

    - Prefixing the filename with c in C++

      • C++ version of math.h is cmath

- Using namespace std

  • Directive

  • Which version you mean

    - XXX::hello() or YYY::hello()

  • Classes, functions and variables that are a standard component of C++ compilers are place in a namespace std

  • Std::cout, std::endl

- • Using std::cout, using std::endl

  - Make available just those names you need

- Character string = characters enclosed in double quotations

  - • cout << MyString;

- Operator overloading

  - • Insertion operator and left-shift operator   <<

  - • Address operator and bitwise AND   &

  - • Multiplication and dereferencing a pointer   *

  - • Symbol can have more than one meaning

- New line

  - • endl - cout << endl;

  - • \n - cout << "\n";

- Tokens and white space

  - • Indivisible elements

- Statements

  - • C++ program is a collection of functions

  - • Function is a collection of statements

  - • Declaration statement creates a variable

  - • Assignation statement provides a value for that variable

- BASIC - you don't need to declare variables

- C++ style - declare variable just before it is used, not in the beginning

- You can use = serially

- cout

  - • Better than printf()

  - • Recognizes types

  - • Extensible - can display new data types you develop

- Get input

  • cin >> number;

- Concatenate cout

  • cout << "My number is " << number << endl;

- Class

  • OOP

  • Data type that user defines

- Class libraries

  • Not build in to the C++ language

  • Come with language

- Functions

  • Argument, parameter = passed to function

  • Return value

  • Function prototype

    - What types are involved (info sent to the function and the info sent back)

  • Function definition

    - Code for function

- C++ library functions

  • stored in library files

  • Compilers automatically search the math library

  • Standard C library - 140 predefined functions

- Number = main();

  • Your operating system is calling your program

  • OS can use return value (exit value)

    - Convention - 0 …program ran successfully, nonzero … problem

- Keywords

- Can't use as variable name (except main), function and object names