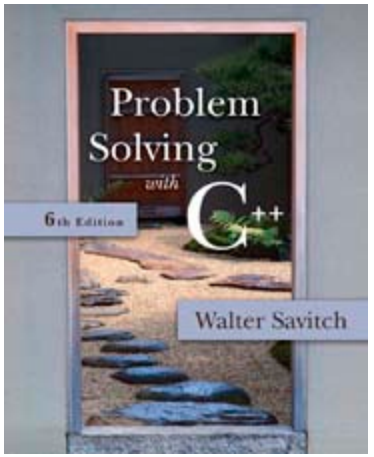
APS105: Lecture 4

Wael Aboelsaadat

wael@cs.toronto.edu

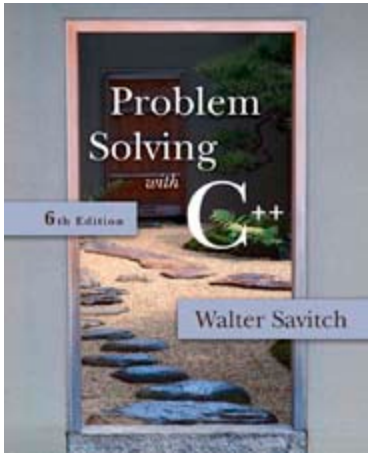
http://ccnet3.utoronto.ca/20079/aps105h1f/

Acknowledgement: These slides are a modified version of the text book slides as supplied by Addison Wesley



Chapter 1

Introduction to Computers and C++ Programming



A Sample C++ Program

- A simple C++ program begins this way

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

int main()
{
```

- And ends this way

```
        return 0;
}
```

Display 1.8

Explanation of code (1/5)

- Variable declaration line

  int number_of_pods, peas_per_pod, total_peas;

  - Identifies names of three variables to name numbers
  - int means that the variables represent integers

Explanation of code (2/5)

- Program statement

  cout << “Press return after entering a number.\n”;

  - cout (see-out) used for output to the monitor
  - “<<“ inserts “Press…a number.\n” in the data bound for the monitor
  - Think of cout as a name for the monitor
    - “<<“ points to where the data is to end up
  - ‘\n’ causes a new line to be started on the monitor

Explanation of code (3/5)

- Program statement

  cin >> number_of_pods;

  - cin (see-in) used for input from the keyboard
  - “>>” extracts data from the keyboard
  - Think of cin as a name for the keyboard
    - “>>” points from the keyboard to a variable where the data is stored

Explanation of code (4/5)

- Program statement

```
total_peas = number_of_pods * peas_per_pod;
```

  - Performs a computation
  - ‘*’ is used for multiplication
  - ‘=‘ causes total_peas to get a new value based on the calculation shown on the right of the equal sign

Explanation of code (5/5)

- Program statement

```
cout << number_of_pods;
```

  - Sends the value of variable number_of_pods to the monitor

Program Layout (1/3)

- Compiler accepts almost any pattern of line breaks and indentation
- Programmers format programs so they are easy to read
  - Place opening brace ‘{‘ and closing brace ‘}’ on a line by themselves
  - Indent statements
  - Use only one statement per line

Program Layout (2/3)

- Variables are declared before they are used
  - Typically variables are declared at the beginning of the program
  - Statements (not always lines) end with a semi-colon

- Include Directives

  #include <iostream>

  - Tells compiler where to find information about items used in the program
  - iostream is a library containing definitions of cin and cout

Program Layout (3/3)

- using namespace std;
- Tells the compiler to use names in iostream in a “standard” way

- To begin the main function of the program
- int main()
  {
- To end the main function
  return 0;
  }
  - Main function ends with a return statement

Running a C++ Program

- C++ source code is written with a text editor
- The compiler on your system converts source code to object code.
- The linker combines all the object code into an executable program.

Run a Program

- Obtain code in Display 1.10

Display 1.10

- Compile the code
- Fix any errors the compiler indicates and re-compile the code
- Run the program
- Now you know how to run a program on your system

Section 1.3 Conclusion

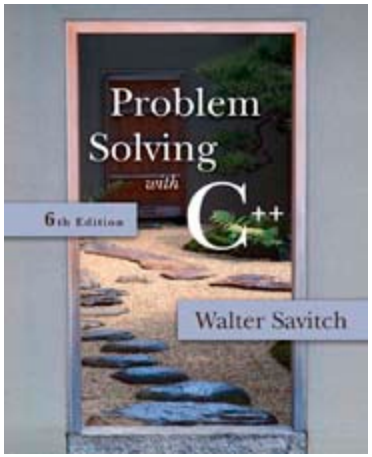
- Can you…
  - Describe the output of this line?

    cout << “C++ is easy to understand.”;
  - Explain what this line does?

    cin >> peas_per_pod;
  - Explain this? #include <iostream>

1.4

Testing and Debugging



Testing and Debugging

- Bug
  - A mistake in a program
- Debugging
  - Eliminating mistakes in programs
  - Term used when a moth caused a failed relay on the Harvard Mark 1 computer. Grace Hopper and other programmers taped the moth in logbook stating:

    “First actual case of a bug being found.”

Program Errors

- Syntax errors
  - Violation of the grammar rules of the language
  - Discovered by the compiler
    - Error messages may not always show correct location of errors
- Run-time errors
  - Error conditions detected by the computer at run-time
- Logic errors
  - Errors in the program's algorithm
  - Most difficult to diagnose
  - Computer does not recognize an error

Section 1-4 Conclusion

- Can you…
  - Describe the three kinds of program errors?
  - Tell what kind of errors the compiler catches?
  - What kind of error is produced if you forget a punctuation symbol such as a semi-colon?
  - Tell what type of error is produced when a program runs but produces incorrect results?

Writing & Compiling C++ on Windows

- TextEdit

  http://www.textedit.org/download/TESetup.exe

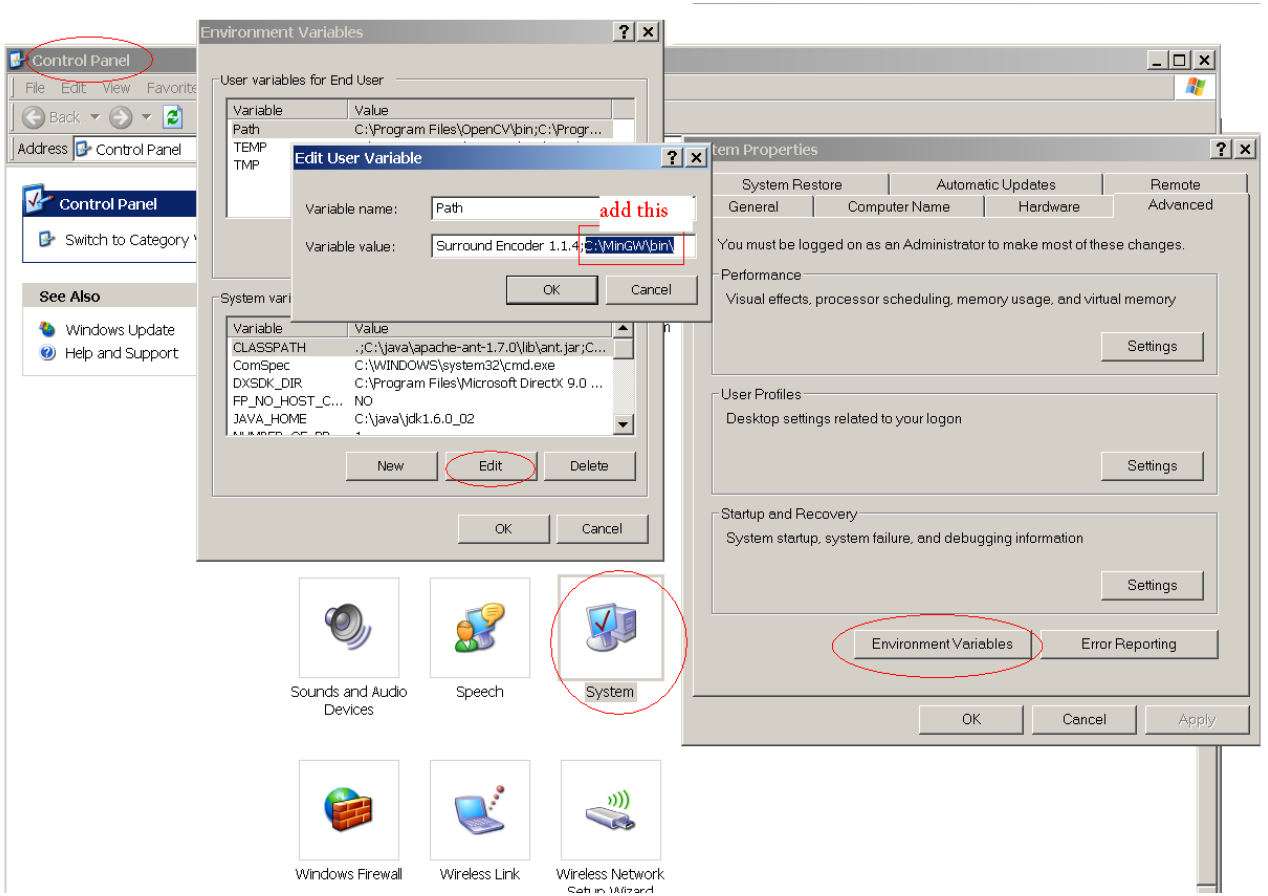
- Mingw

  http://downloads.sourceforge.net/mingw/MinGW-5.1.3.exe?modtime=1168811236&big_mirror=1

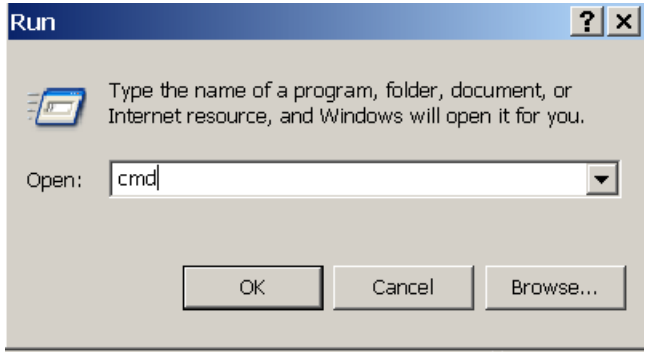
- Set the Path to the location where you installed mingw

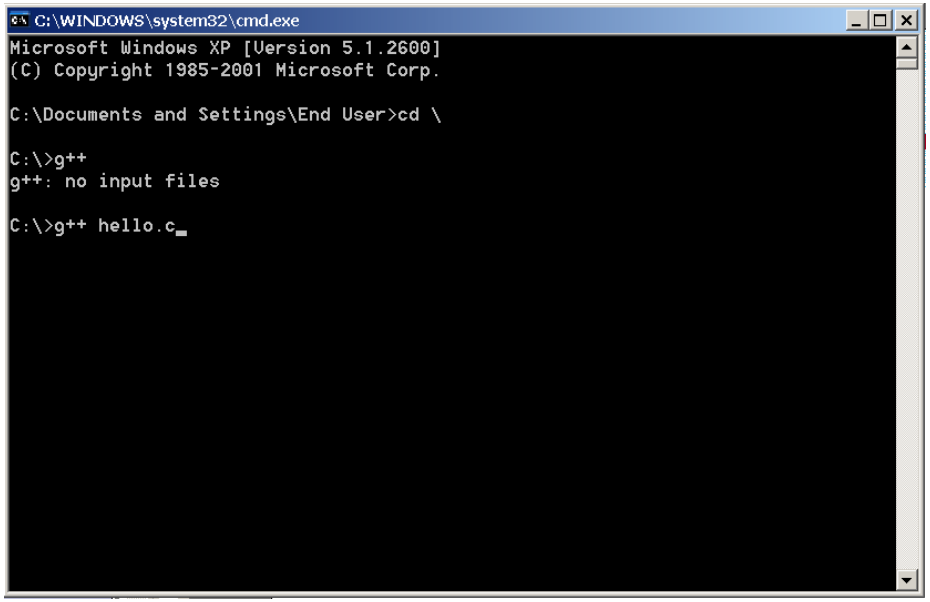
- Use command line to compile & run

Writing & Compiling C++ on Windows

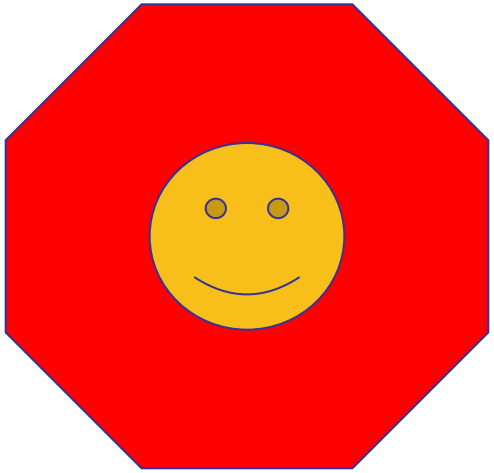


Writing & Compiling C++ on Windows





Chapter 1 -- End



Display 1.8

A Sample C++ Program

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

int main( )
{
    int number_of_pods, peas_per_pod, total_peas;

    cout << "Press return after entering a number.\n";
    cout << "Enter the number of pods:\n";
    cin >> number_of_pods;
    cout << "Enter the number of peas in a pod:\n";
    cin >> peas_per_pod;

    total_peas = number_of_pods * peas_per_pod;

    cout << "If you have ";
    cout << number_of_pods;
    cout << " pea pods\n";
    cout << "and ";
    cout << peas_per_pod;
    cout << " peas in each pod, then\n";
    cout << "you have ";
    cout << total_peas;
    cout << " peas in all the pods.\n";

    return 0;
}
```

Sample Dialogue

```
Press return after entering a number.
Enter the number of pods:
10
Enter the number of peas in a pod:
9
If you have 10 pea pods
and 9 peas in each pod, then
you have 90 peas in all the pods.
```

Display 1.9

Layout of a Simple C++ Program

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

int main( )
{
    Variable_Declarations

    Statement_1
    Statement_2
      ...
    Statement_Last

    return 0;
}
```

Display 1.10

Testing Your C++ Setup

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

int main( )
{
    cout << "Testing 1, 2, 3\n";
    return 0;
}
```

If you cannot compile and run this program, then see the programming tip entitled "Getting Your Program to Run." It suggests some things you might do to get your C++ programs to run on your particular computer setup.

Sample Dialogue

```
Testing 1, 2, 3
```