

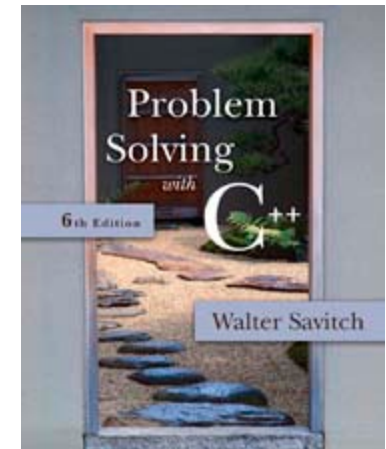
APS105: Lecture

Wael Aboelsaadat

wael@cs.toronto.edu

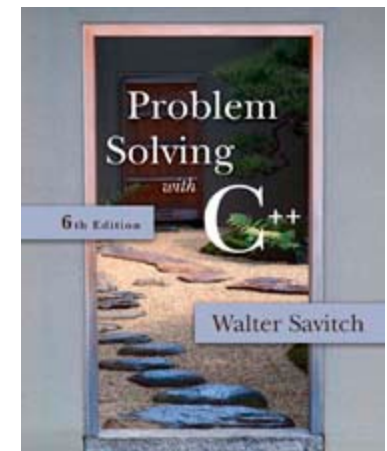
<http://ccnet3.utoronto.ca/20079/aps105h1f/>

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



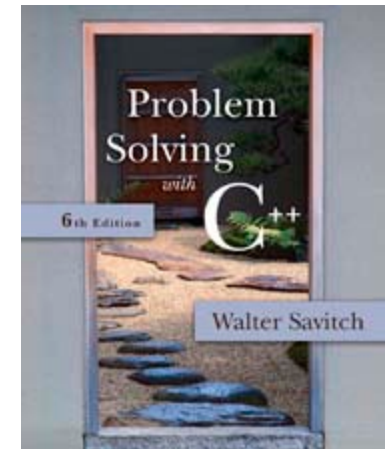
Chapter 2

C++ Basics

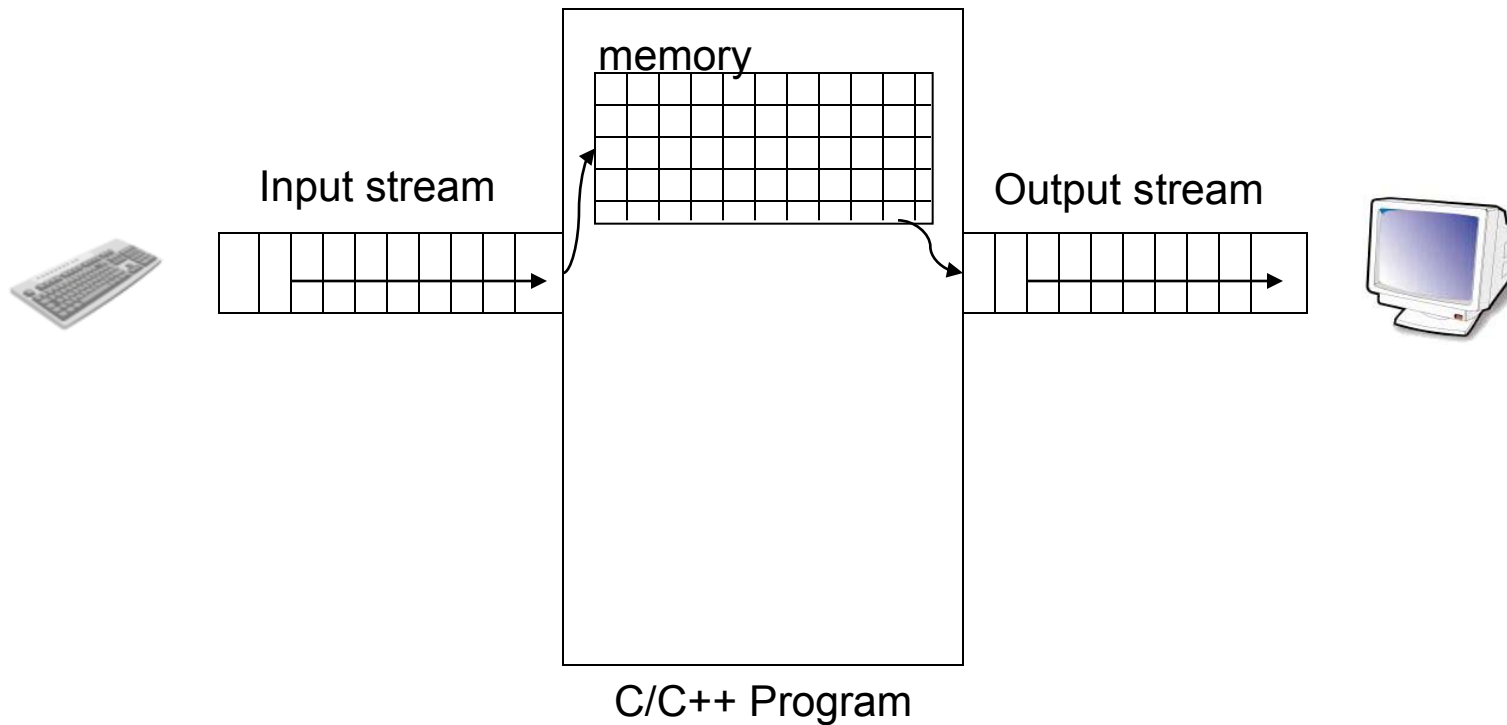


2.2

Input and Output



Input and Output



Output using cout

- cout is an output stream sending data to the monitor
- The insertion operator "<<" inserts data into cout
- Example:

```
cout << number_of_bars << " candy bars\n";
```

 - This line sends two items to the monitor
 - The value of number_of_bars
 - The quoted string of characters " candy bars\n"
 - Notice the space before the 'c' in candy
 - The '\n' causes a new line to be started following the 's' in bars
 - A new insertion operator is used for each item of output

Examples Using cout

- This produces the same result as the previous sample

```
cout << number_of_bars ;  
cout << " candy bars\n";
```

- Here arithmetic is performed in the cout statement

```
cout << "Total cost is $" << (price + tax);
```
- Quoted strings are enclosed in double quotes ("Walter")
 - Don't use two single quotes ('')
- A blank space can also be inserted with

```
cout << " " ;
```

if there are no strings in which a space is desired as
in " candy bars\n"

Include Directives

- Include Directives add library files to our programs
 - To make the definitions of the cin and cout available to the program:

```
#include <iostream>
```

- Using Directives include a collection of defined names
 - To make the names cin and cout available to our program:

```
using namespace std;
```

Escape Sequences

- Escape sequences tell the compiler to treat characters in a special way
- '\' is the escape character
 - To create a newline in output use
 - `\n` – `cout << "\n";`
 - or the newer alternative
 - `cout << endl;`
 - Other escape sequences:
 - `\t` -- a tab
 - `\\` -- a backslash character
 - `\"` -- a quote character

Formatting Real Numbers

- Real numbers (type double) produce a variety of outputs

```
double price = 78.5;  
cout << "The price is $" << price << endl;
```

- The output could be any of these:
 - The price is \$78.5
 - The price is \$78.500000
 - The price is \$7.850000e01
- The most unlikely output is:
 - The price is \$78.50

Showing Decimal Places

- cout includes tools to specify the output of type double
- To specify fixed point notation
 - `setf(ios::fixed)`
- To specify that the decimal point will always be shown
 - `setf(ios::showpoint)`
- To specify that two decimal places will always be shown
 - `precision(2)`
- Example:

```
cout.setf(ios::fixed);
cout.setf(ios::showpoint);
cout.precision(2);
cout    << "The price is "
        << price << endl;
```

A C++ Program (part 1 of 2)

```
#include <iostream>
using namespace std;
int main()
{
    int number_of_bars;
    double one_weight, total_weight;

    cout << "Enter the number of candy bars in a package\n";
    cout << "and the weight in ounces of one candy bar.\n";
    cout << "Then press return.\n";
    cin >> number_of_bars;
    cin >> one_weight;

    total_weight = one_weight * number_of_bars;

    cout << number_of_bars << " candy bars\n";
    cout << one_weight << " ounces each\n";
    cout << "Total weight is " << total_weight << " ounces.\n";

    cout << "Try another brand.\n";
    cout << "Enter the number of candy bars in a package\n";
    cout << "and the weight in ounces of one candy bar.\n";
    cout << "Then press return.\n";
    cin >> number_of_bars;
    cin >> one_weight;

    total_weight = one_weight * number_of_bars;

    cout << number_of_bars << " candy bars\n";
    cout << one_weight << " ounces each\n";
    cout << "Total weight is " << total_weight << " ounces.\n";

    cout << "Perhaps an apple would be healthier.\n";

    return 0;
}
```

Display 2.1 (1/2)



Display 2.1

(2 / 2)



A C++ Program (*part 2 of 2*)

Sample Dialogue

Enter the number of candy bars in a package
and the weight in ounces of one candy bar.
Then press return.

11 2.1

11 candy bars

2.1 ounces each

Total weight is 23.1 ounces.

Try another brand.

Enter the number of candy bars in a package
and the weight in ounces of one candy bar.
Then press return.

12 1.8

12 candy bars

1.8 ounces each

Total weight is 21.6 ounces.

Perhaps an apple would be healthier.

Display 2.2



DISPLAY 2.2 Some Number Types

Type Name	Memory Used	Size Range	Precision
<i>short</i> (also called <i>short int</i>)	2 bytes	-32,767 to 32,767	(not applicable)
<i>int</i>	4 bytes	-2,147,483,647 to 2,147,483,647	(not applicable)
<i>long</i> (also called <i>long int</i>)	4 bytes	-2,147,483,647 to 2,147,483,647	(not applicable)
<i>float</i>	4 bytes	approximately 10^{-38} to 10^{38}	7 digits
<i>double</i>	8 bytes	approximately 10^{-308} to 10^{308}	15 digits
<i>long double</i>	10 bytes	approximately 10^{-4932} to 10^{4932}	19 digits

These are only sample values to give you a general idea of how the types differ. The values for any of these entries may be different on your system. *Precision* refers to the number of meaningful digits, including digits in front of the decimal point. The ranges for the types *float*, *double*, and *long double* are the ranges for positive numbers. Negative numbers have a similar range, but with a negative sign in front of each number.