Format Review and File Input and Output

Larry Caretto
Computer Science 106

Computing in Engineering and Science

February 16, 2006

California State University Northridge

Outline

- Last Tuesday – Formatting output
  - Controlling appearance compared with default output of six significant figures
  - Manipulators for width and precision
- Today – File input and output
  - Uses same operators as screen output and keyboard input
  - Associate program variable name for file with operating system name

How do We Format Output?

- Use manipulators in output statements
- Requires use of #include<iomanip>
- Manipulators we will use
  - **fixed** forces fixed format output
  - **scientific** forces scientific format output
  - **setw(w)** assigns w spaces for output (right justified by default; see **left** and **right**)
  - **setprecision(p)** uses the value of p to set the number of significant figures

Persistence of Manipulators

- setw(w) is in effect for one output item only even in same output statement
  - cout << setw(10) << x << setw(10) << y;
- All other manipulators are in effect until changed (even over multiple output statements)
  - cout << fixed << x << " " << y;
  - cout << z; // fixed still in effect!

Use of setprecision(p)

- Effect depends on other options
  - For default output (not using fixed or scientific manipulators) setprecision(p) gives p significant figures
  - If fixed or scientific manipulators are used, setprecision(p) gives p decimal places
    - For scientific output p decimal places is p+1 significant figures
- Once used, setprecision(p) remains in effect until changed
- Setprecision(6) restores default

setw and setprecision

- setprecision controls the number of digits to be printed
- setw controls how many spaces are used for the output
- What happens if the setw manipulator does not give enough spaces
  - The entire output item is printed
  - There is no spaced between the output item and the previous output (if any)

setw and setprecision Question

- double w = -3.56878e5, x = 1.234; y = 784.525, z = 23456.789;
- Write the output statements to print w and x on the first line and y and z on the second line
- Print each number with 3 decimal places and allow at least two spaces between the numbers
- What is output without formats?
  - cout << w << x << endl << y << z;

setw and setprecision Question

- How do you print 3 decimal places?
- cout << fixed << setprecision(3);
- What is width in setw(width)?
- Have to consider number of spaces required for largest number
- How many spaces are required?
- First, we need 3 spaces numbers after the decimal and 1 for the decimal point for a total of 4 spaces

setw and setprecision Question

- How many spaces are required in front of the decimal place?
- Which number to be printed has the maximum size: w = -3.56878e5, x = 1.234; y = 784.525, z = 23456.789
- Maximum characters before decimal is seven for -3.56878e5 = -356878
- Don't forget that we want at least two spaces before this number

setw and setprecision Question

- Now, let's recap the number of spaces we need for the setw manipulator
- Add spaces required after decimal, before decimal, spacing between numbers and decimal point

  3 + 7 + 2 + 1 = 13 => setw(13)
- What is cout statement?
- cout << fixed << setprecision(3) << setw(13) << w << setw(13) << x << endl << setw(13) << y << setw(13) << z ;

Input and Output Files

- Want to be able to read from and write to files on disk storage
- Such files can be accessed by computer operating system
- Have operating system file name that has short and long form
  - Short: program.dat
  - Long: C:\temp\program.dat
- File name structure is different for other operating systems

Input and Output Files II

- Other programs can use output files prepared by C++ programs
- Can use other programs to prepare input files for C++ programs
- File input and output on C++ requires #include <fstream> for file library
- C++ commands use a program variable file name to refer to files
- Must associate program variable file name with operating system file name

C++ File Input and Output

- Three steps
- Use #include<fstream> directive to include file library in program
- Associate program variable file name with operating system file name
- Use same syntax as cout and cin
- Replace cout by and cin by program variable names for output and input files

C++ File Input

- Use #include<fstream>
- Use ifstream command to create program variable name for file and link to operating system file name
  - ifstream myInput("input.dat");
  - ifstream in("januaryData.txt");
  - ifstream inFile("exercise4.001");

  You choose both file names
- Use name to read from file
  - inFile >> x >> y >> z; (like cin >> x >> y >> z;)
  - myInput >> date >> time;

Input File Examples

```
12
20
32
55
43
19
27
88
```

```
12 20 32 55 43 19 27 88
x  w  u  v  a  b  c
```

```
12 20
32 55
43 19
27 88
```

- All three files are the same to C++
  - Same sequence of numbers
- ifstream inDat("today.dat")
- inDat >> x;
- inDat >> w >> u >> v;
- inDat >> a >> b >> c;
- What values do u and c have?

Output Files

- Like input, but use ofstream not ifstream
- ofstream command associates program and operating system names
  - ofstream outFile("results.out");
  - ofstream out("case.012");
- Use program name in place of cout to write to file
  - outFile << setw(10) << x;
  - out << "Results for Case " << case;

General File Syntax

- ifstream ***<InputFileVariableName>*** (***"<OperatingSystemInputFileName>"***);
- ofstream ***<OutputFileVariableName>*** (***"<OperatingSystemOutputFileName>"***);
- ***<OutputFileVariableName>*** << ***<OutputList>*** *// like cout*
- ***<InputFileVariableName>*** >> ***<InputList>*** *// like cin*
- You choose both names

Program File Example

```
// assumes #include<fstream> used
ifstream iFile("myInput.txt");
ofstream out("prog.1");
double x, y, z;
iFile >> x >> y;
z = sqrt( x * x + y * y );
out << "          x          y          z\n";
out << setw(12) << x << setw(12) << y <<
  setw(12) << z;
```

Operating system names

Program variable names

Accessing Files

- Access file from any windows program using operating system file name
- Can use Visual C++ (or other text editors) to create input files and view output files
- Input files must be in project folder
- To display available output files, select "All Files" from the pulldown menu next to files of type in the Open menu

California State University Northridge 19

File Questions

- Write statement to associate program name myInput with the operating system name, data.txt, as an input file

```
ifstream myInput( "data.txt" );
```

- Read the first three data items on the file data.txt into the variables x1, x2, and x3 in that order

```
myInput >> x1 >> x2 >> x3;
```

California State University Northridge 20

More File Questions

- Write statement to associate the operating system name, results.out, with a program variable name to be used as an output file
- Write the variables x and y to results.out, preceeded by the strings "x = " and "y = "

```
ofstream o( "results.out" );
o << "x = " << x << ", y = " << y;
```

California State University Northridge 21

Summary

- File input and output uses same syntax and commands as keyboard input and screen output
- Replace cin by program name for input file and cout by program name for output file
- Use ifstream and ofstream to associate operating system file names with program names for input and output

California State University Northridge 22