

# Lab 2 - CMPS 1043, Computer Science I

## Introduction to File Input/Output (I/O)

### Projects and Solutions (C++)

(Revised from <http://msdn.microsoft.com/en-us/library/bb384842.aspx>)

**\* Keep this information to reference when completing your projects. \***

In Visual Studio, you organize your work in projects and solutions. A solution can contain more than one project. For more information, see [Introduction to Solutions, Projects, and Items](#).

#### To create a new project

1. From the **File** menu, point to **New**, and then click **Project...**
2. In the **Project Types** area, click **Visual C++**. Then, in the **Visual Studio installed templates** pane, click **Win32 Console Application**.
3. Type a name for the project. In this example, we'll use **HelloFiles>YourName**.

When you create a new project, Visual Studio puts the project in a solution. Accept the default name for the solution, which is the same name as the project.

You can accept the default location, type a different location, or browse to a directory where you want to save the project.

Press **OK** to start the **Win32 Application Wizard**.

4. On the **Overview** page of the **Win32 Application Wizard** dialog box, click **Next**.
5. On the **Application Settings** page under **Application type**, select **Console Application**. Under **Additional options** select the **Empty Project** setting and click **Finish**.

You now have a project without source code files.

#### To add a new source file (program)

1. From the **Project** menu, click **Add New Item**.

Alternatively, to use Solution Explorer to add a new file to the project, right-click the **Source Files** folder in Solution Explorer and point to **Add**. Then click **New Item**.

In the **Visual C++** area, select **Code**. Then click **C++ File (.cpp)**.

2. Type the Name of the file to be added (for example, type **HelloFiles>YourName**) and click **Add**. The source file **HelloFiles>YourName.cpp** will be added to the project.
3. In the **HelloFiles.cpp** editing window, type the code below for the **HelloFiles>YourName** program. In this example, **HelloFiles>YourName.cpp** will be the main function. Note: Punctuation, Spacing and

Indentation are important!! Blank lines are used for readability, but do not affect the execution of the program.

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

int main( )
{
    int birth_Date;
    ifstream infile;
    ofstream outfile;

    infile.open("input.txt");
    outfile.open("output.txt");
    infile >> birth_Date;

    // You may put other statements here

    outfile << " your name here \n";
    outfile << birth_Date;
    infile.close();
    outfile.close();
    return 0 ;
}
```

4. To compile the program, on the **Build** menu, click **Build Solution**.

You should see output from the build in the **Output** window indicating that the project compiled without errors. If not, correct the errors indicated in the **Output** window.

#### To add a text file for input or output

1. From the **Project** menu, click **Add New Item**.

In the **Visual C++** area, select **Utility**. Then click **Text File (.txt)**.

2. Type the Name of the file to be added (for example, type **input** if the file is to contain data to be read by the program or type **output** if the file will have execution results written to it) and click **Add**. The empty text file (**input.txt** or **output.txt**) will be added to the project. (Observe that they have been listed in the folders list on the left of the screen.)

NOTE: For this project you will add both an **input** and an **output** file.

#### To execute the program

**CTRL + F5** will compile and execute your program. Because all output was to a file, you will **NOT** see the Display Window with results on the screen. Observe the Output Window at the bottom of the screen. If no error messages appear, your program probably executed properly.

## To View Your Results

Click on output.txt either in the left-most window or its tab, if displayed. Your results will be displayed in the main window.

## Printing your Program Code

With your program code displayed on the screen, click the **Windows icon** in the upper left corner of the screen, select **Print**. If you are using a color printer, your printout will be in color as on the screen. If not, it will be B/W. Color is NOT necessary for work turned in to your instructors. Lab printers are not color.

## Printing your Input and Output files

With your file code displayed on the screen, click the **Windows icon** in the upper left corner of the screen, select **Print**. Do this for each file that you want to print.

## Documentation

Good programming practice insists that all program files must contain documentation. That is, comments from the programmer providing basic information about the program. Documentation lines begin with the **//** symbols. All programs should have at least the following documentation at the beginning of EVERY program. Your instructor may require additional documentation be included.

```
// Your Last Name, Your First Name
// Computer Science 1 - Instructor name
// Project # - Project Name
// Date
// Additional information as required by your instructor
```

In addition, the body of your program should also be documented. This includes adding comments for all components of your program. For more examples of appropriate documentation, see the sample programs in your textbook. Your lab assignment is fully documented. You are to include all the comments as you complete the assignment.

**\*Important Notes:** Sometimes you will want to type a program in Word or Notepad then copy into the Visual Studio programming window. If you do this, you MUST check it very carefully because there WILL BE errors in the copy. For example, quotes (“ ”) must be deleted and retyped after the copy; they will not be the correct version of quotes for C++. Word automatically capitalized some words. This will cause errors in C++ so you will have to retype some words, removing the capital letter at the first of the word or line. There are probably others. \*

**\*Learn to recognize the colors in the Visual Studio editor;** it will help you spot errors. For example, comments are always in green or should be. Also, literals (words in legal quotes) will be or should be in red. Commands are in black; reserved words in blue.

## LAB 2 – Assignment

Modify the program that you began in class to include the additional code below, including all the documentation (comments). Create an input file containing your year of birth on the first line and the values 2.5 and 3.7 on the second line (as shown in the sample output below). Execute the program, correcting errors as necessary. Print off the program code, the input file and the output file. Staple them together in this order and turn in no later than the first class meeting next week (week of February 7).

```
// Type your name here
// Computer Science I – Instructor Name (Halverson or Donovan)
// Project 2 – Lab 2
// January ??, 2011
// This is a sample program for the purpose of learning how to enter, compile,
// and execute (run) a program with input and output files using the
// Microsoft C++ 8.0 programming environment.
// It is in the format that each program submitted for grading should follow.
//
#include <iostream>
#include <fstream>
using namespace std;
int main( )
{
    // Declarations
    int birth_Date;           // declare integer variable for your year of birth
    float A, B, Sum, Product; // declare real variables for arithmetic
    ifstream infile;         // declare an input file
    ofstream outfile;        // declare an output file

    infile.open("input.txt"); // open "input.txt" for reading
    outfile.open("output.txt"); // open "output.txt" for writing

    // Program Code for reading your year of birth
    infile >> birth_Date;    // read a value for date of birth from input.txt

    // Program Code for performing some arithmetic operations
    infile >> A >> B;         // read values for A & B from input.txt
    Sum = A + B;             // Add A & B, storing result in Sum
    Product = A * B;        // Multiply A & B, storing result in Product

    // Program Code to output results to the output file
    outfile << " your name here \n"; // write your name to "out.txt"
    outfile << birth_Date << endl;   // write birth date to "out.txt"
    outfile << " A and B have the values " << A << " " << B << endl;
    outfile << "The sum is " << Sum << endl;
    outfile << "The Product is " << Product << endl;

    // Commands to close files
    infile.close();          // always close each file so that the I/O buffer will be
    outfile.close();         // written (otherwise, you may not get all of your
                             // results written to the output file)

    return 0 ;
}
```

Sample input & output files

\*\*\*\*\*

Input File

1985

2.5 3.7

\*\*\*\*\*

Output File

Yournamehere

1985

A and B have the values 2 3

The sum is 6.2

The product is 9.25