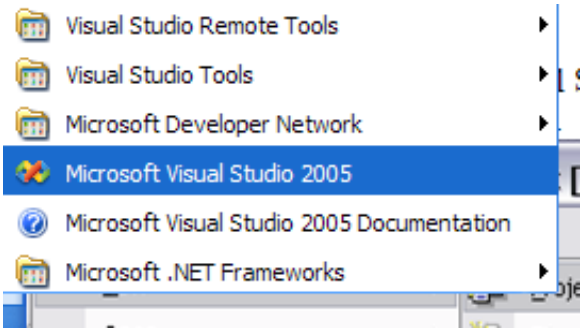
Getting Started with Visual Studio

Visual Studio is a sophisticated but easy to use integrated development environment (IDE) for C++ (and may other languages!) You will see that this environment recognizes C++ keywords and puts them into color. This guide assumes that you have access to a machine with Visual Studio 2005 or 2008 installed. You can download the express editions for free form the Microsoft website. This is a guide written for a laboratory class. Please ignore the irrelevant details.

Starting the Visual Studio Environment

1. Click on Start  in the task bar, and then move the mouse to **All Programs**.
2. Choose *Microsoft Visual Studio 2005 (2008)*
3. Again, choose *Microsoft Visual Studio 2005 (2008)*. See figure below.

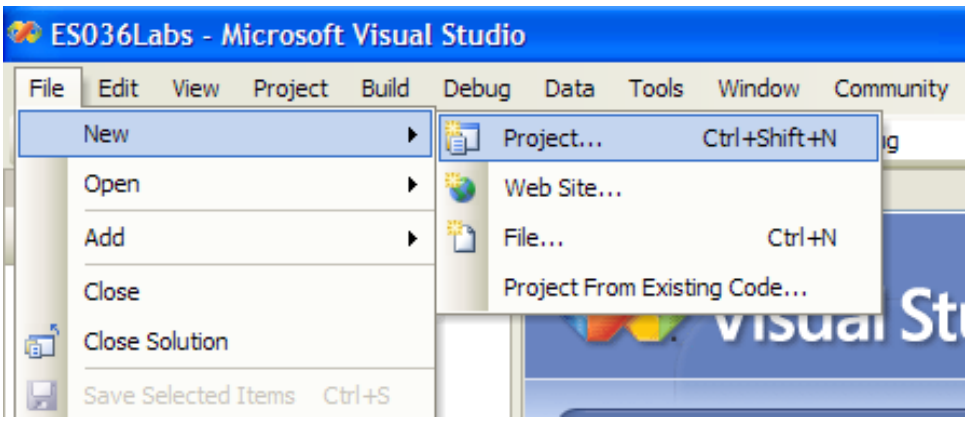


Creating a Project in Visual Studio (to compile and run programs)

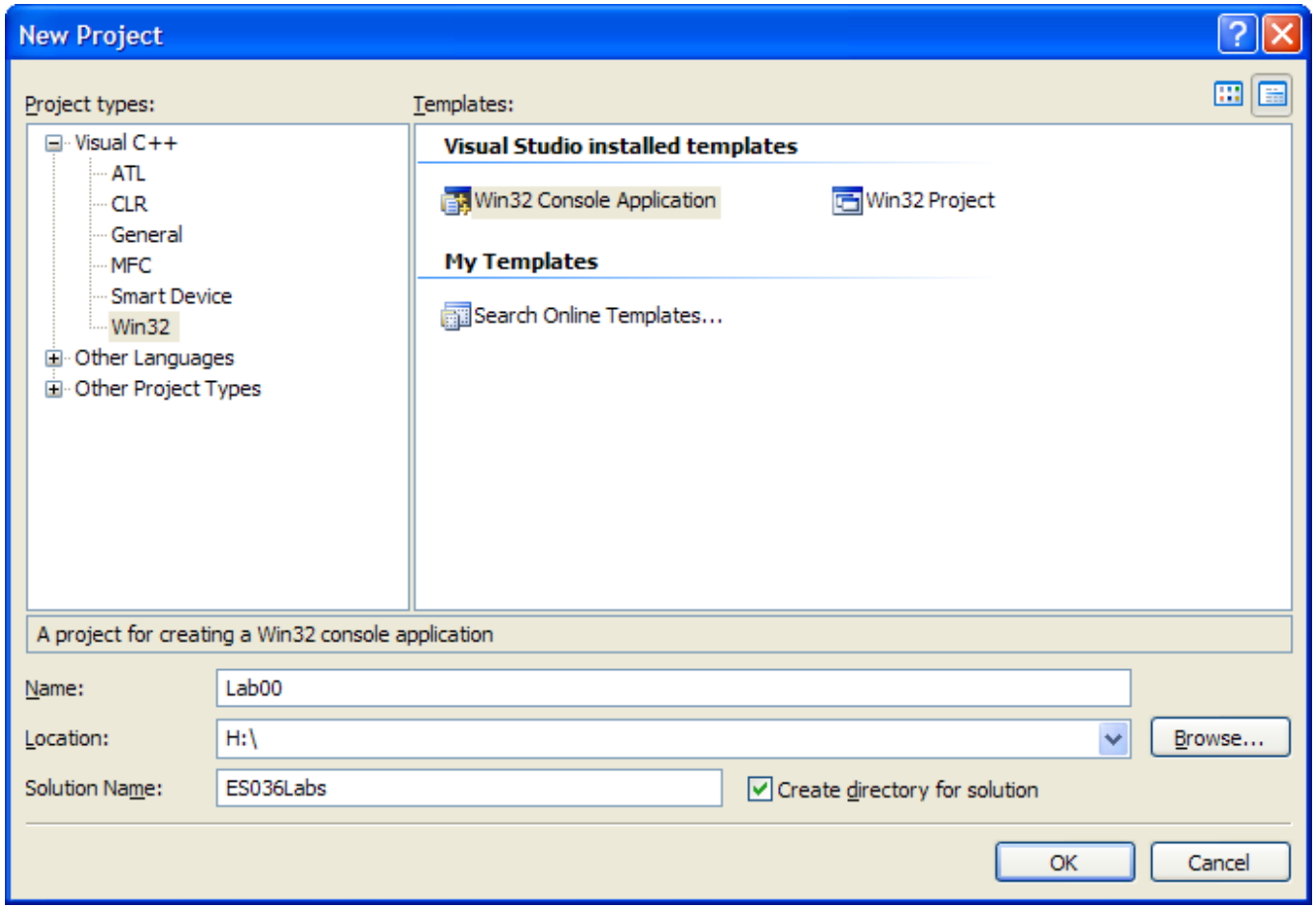
Even though we wish only to compile and run some fairly small and simple programs, we still need to create a *project*. In general, this enables a programmer to go back later and open and modify the source code of this project. Different programs can be contained in different projects. In Visual Studio 2005 or 2008, a *Solution* can contain more than one project. In the following, let's create a solution called *ES036Labs* and a project called *Lab00*. When you want to work on your Lab01 next time, you can open the solution *ES036Labs* and add another project called *Lab01*.

To create a solution and a project:

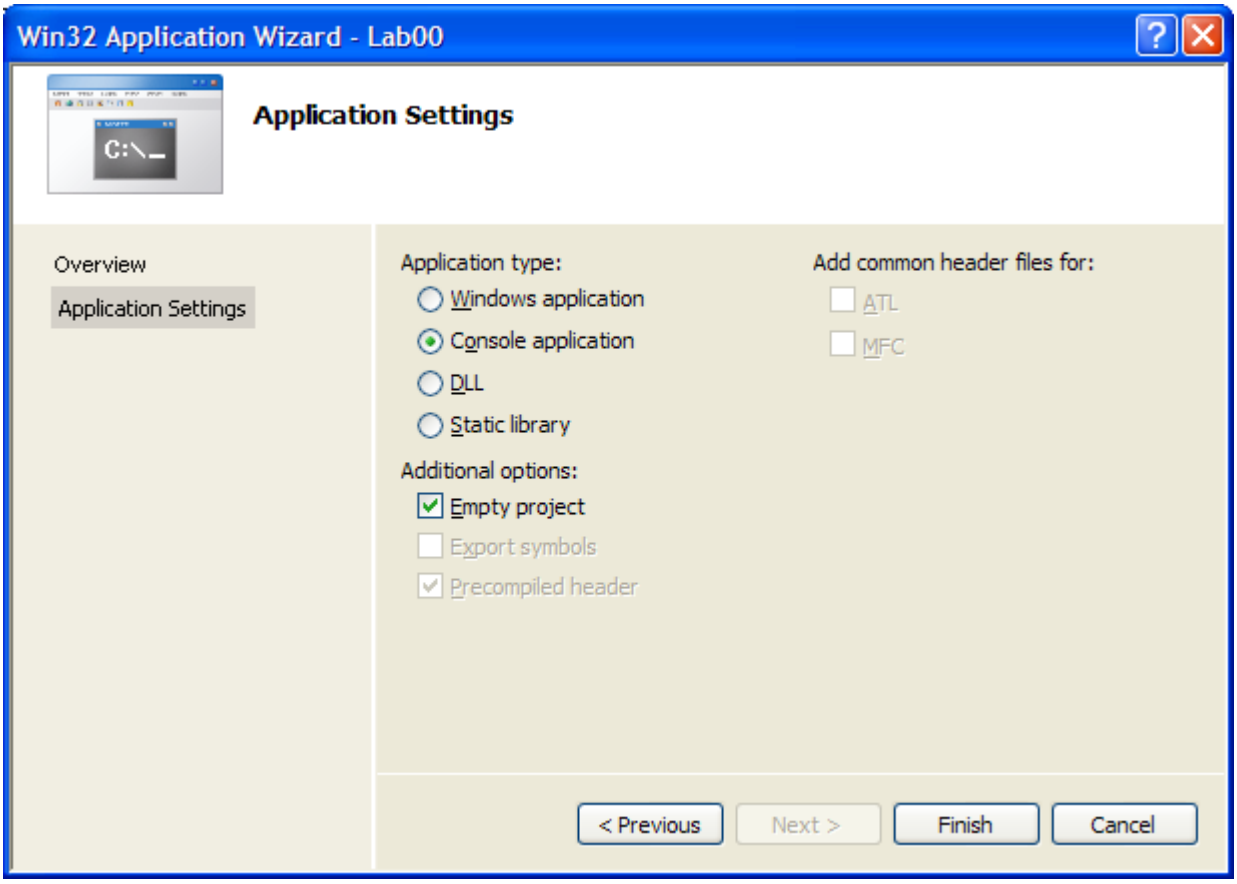
1. Click **File** in the top menu bar of Visual Studio, and then select the **New...** option, and select project (shown on the next page).



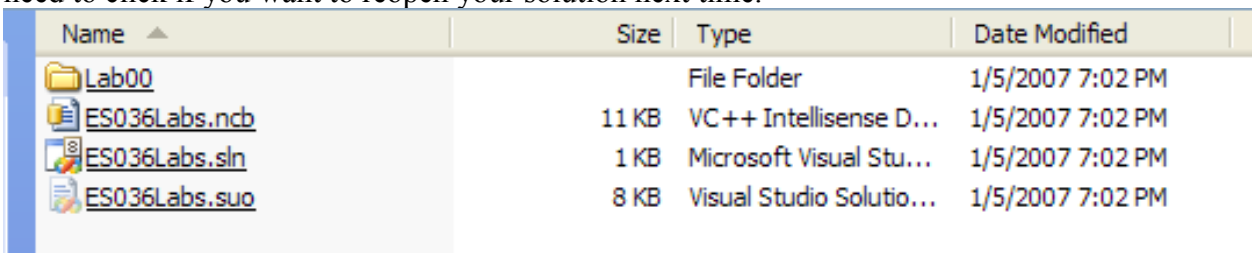
2. This will bring up a dialog box that contains many project possibilities (**shown below**).
3. Click on "Win32" in the left pane of the window (**shown below**).
4. Select "Win32 Console Application" from the right pane of the window (you may need to scroll down in the window to find this option) (**shown below**).



5. Enter a name for the *project* in the lower part of the windows that says <Enter name> (see above): **Lab00**.
6. Make sure that the location specified for the project is on drive H: If it is not, replace the location with H:.
7. Now enter the name for the *solution*: **ES036Labs**.
8. Press **OK** to create the solution and the project.
9. A Win32 Application Wizard window will appear (see on the **next page**). Select "Application Settings" on the left side of this window.
10. Under additional options, check the "Empty Project" box (see on the **next page**).
11. Select Finish (see on the **next page**).

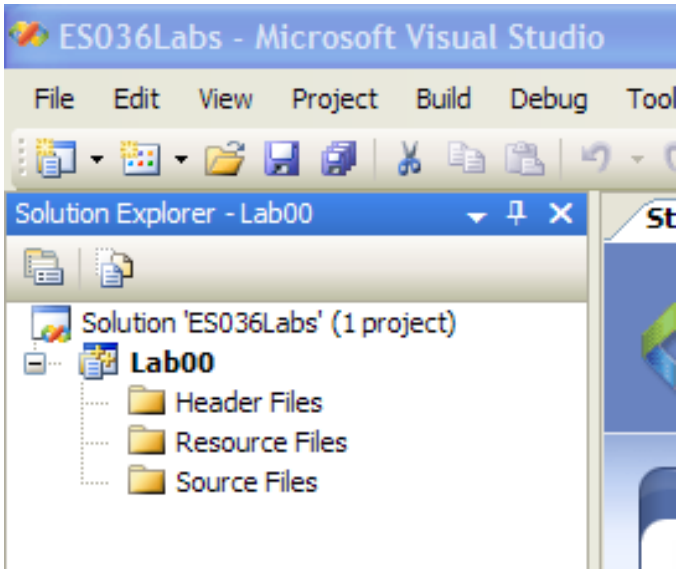


When we complete theis process, Visual Studio create a folder H:\ES036Labs. This is the folder where the solution is located. You will see a file with extension .sln here (ES036Labs.sln). This is the file you need to click if you want to reopen your solution next time.

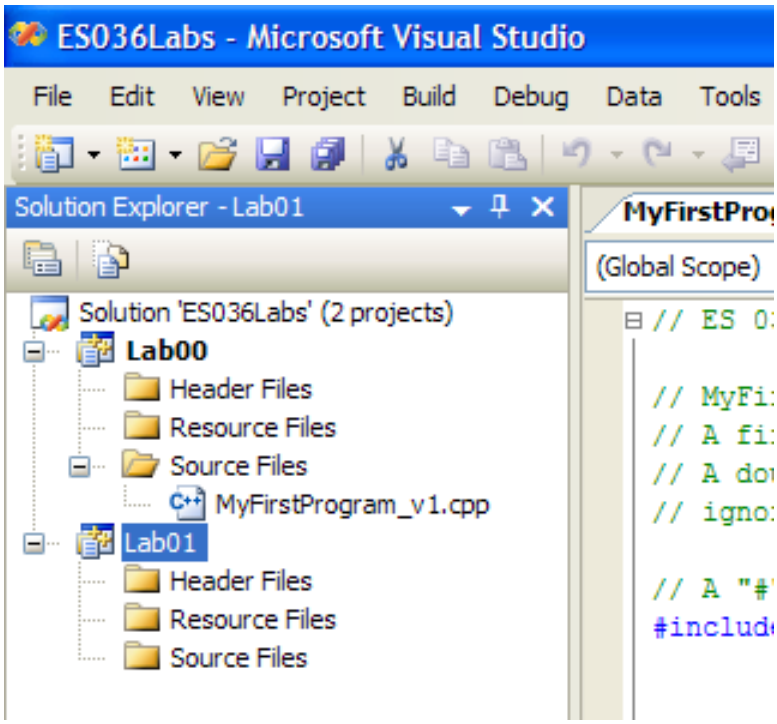


Notice the folder called Lab00 is located in our solution folder. This is the project folder for our Lab00 and the source files (.cpp and .h) are saved inside this folder. If you have your files elsewhere (e.g your USB pen or My Documents) you should copy them into this folder. When you create your Lab01 project next time, another project folder called Lab01 will be created in the solution folder. In your Visual Studio window you will see a view like this at the top left in the following figure[1].

[1] If you do not see this view select **View** > **Solution Explorer** or click on the solution explorer button.



When you have create your Lab01 next time, the view will look like this.

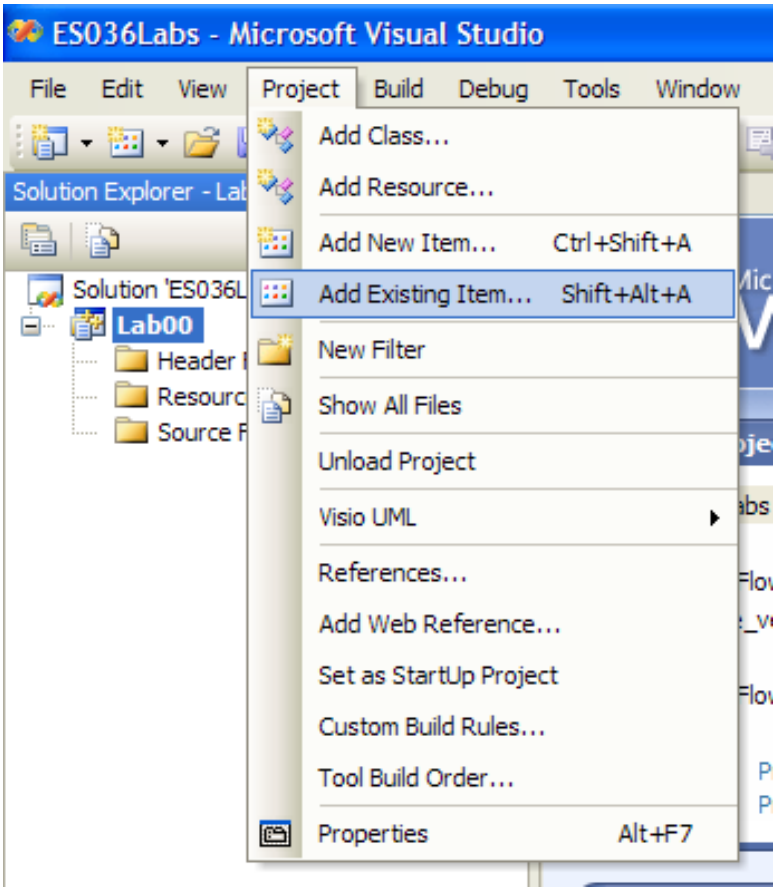


Now we see how Visual Studio helps us to manage our projects and files.

You wish to add some source code to this project:

We have two options when we want to add source code to our project. First is to create new file (**Add New Item**) and the second is to use an existing item (**Add Existing Item**).

1. Copy the .cpp files that you downloaded from the WebCT to folder H:\ES036labs\Lab00
2. Click **Project** in the top menu bar of Visual Studio, and then select the **Add Existing Item** option (see the following figure).

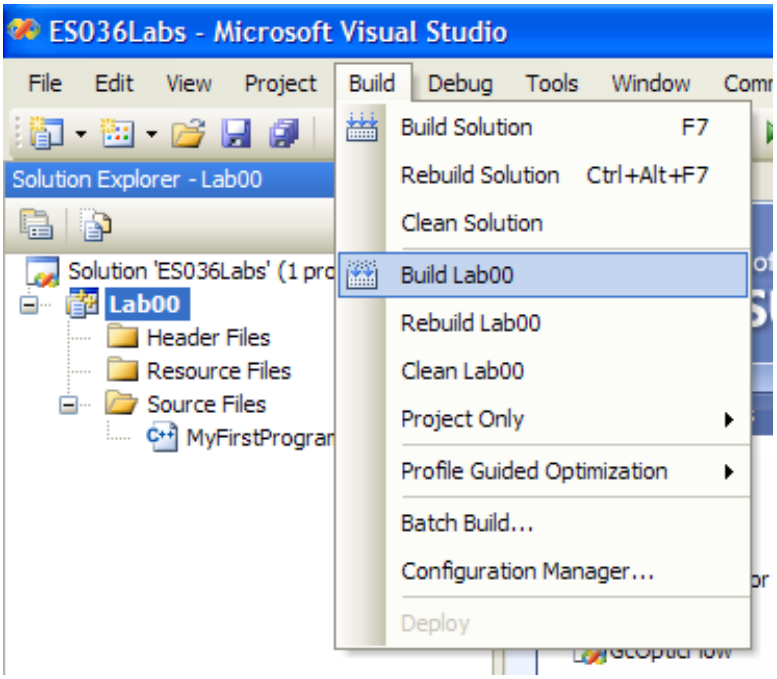


3. Since we have already copied the files to the right place (H:\ES036labs\Lab00) you will see the icon of the file you need to add. Select MyFirstProgram_v1.cpp. If you wish, by selecting the arrow beside the **Look in** address box, you can navigate to the folder in which you saved the MyFirstProgram_v1.cpp file on your network disk area (H:). To see the C++ files in this folder, set the Files of type (see bottom box of **Look in** menu) box to contain Visual C++ Files (which may already be the default value).
4. You should now see the name of this file (MyFirstProgram_v1.cpp) displayed under the folder Source Files in the upper left panel.
5. Double click on MyFirstProgram_v1.cpp under "Source Files" on the left-hand side to display the contents of the file in the main portion of the Visual Studio development environment.

To get the program working we need to do two important things. First we need to compile (and link) the program, then we need to run the program. Here is how to do that.

You wish to compile the source code of this project (MyFirstProgram_v1.cpp):

1. To compile, select the Build menu in the top menu bar (see the following figure).

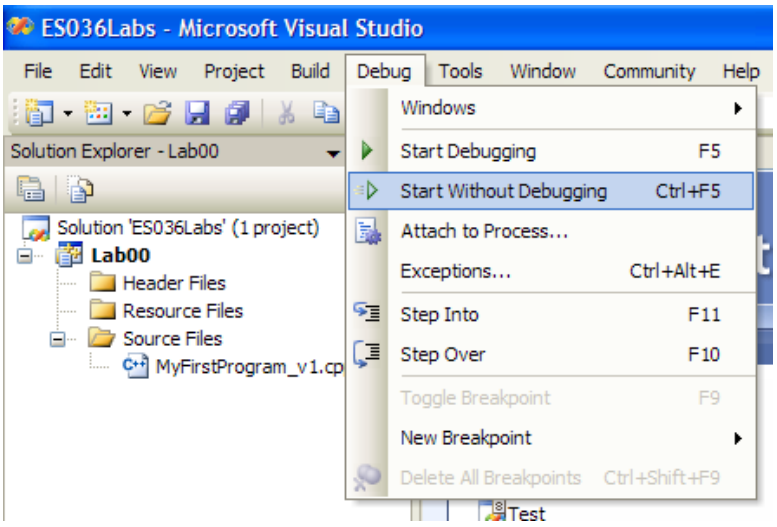


2. Select the option **Build**[2] **Lab00** (assuming that **Lab00** is the name of your project).
3. A lower window will show the results of this process. (As it is, MyFirstProgram_v1.cpp file will compile successfully.) You will see an output like this. You should have 0 error(s).

```
1>Lab00 - 0 error(s), 0 warning(s)
========== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped ==========
```

You wish to run the executable file created in the compilation process:

4. To run (execute), select the **Debug** menu in the top menu bar.



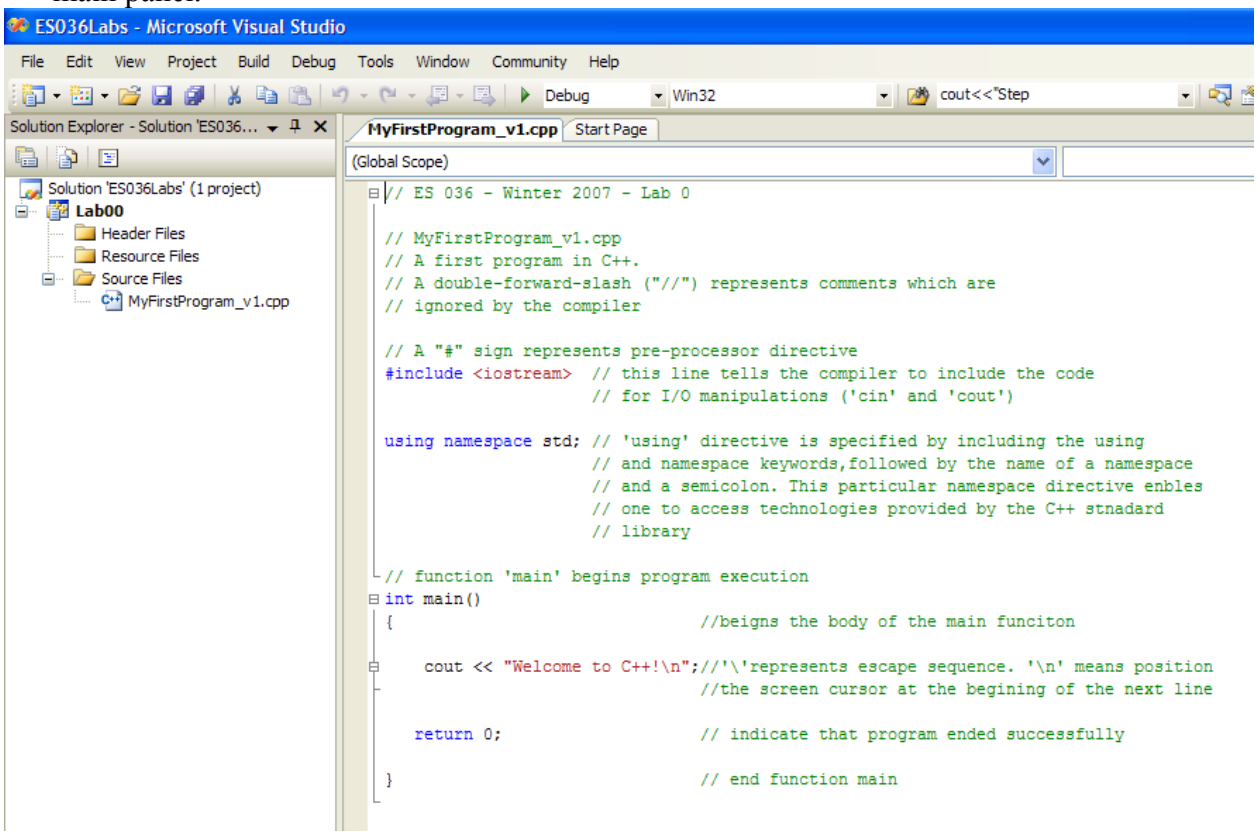
5. Select the option **Start Without Debugging**.

[2] Rebuild Solution will rebuild all the projects in the solution as defined by the configuration manager. If your projects are small this is a convenient option.

6. A black 'DOS-type' window will appear showing the output of your program.

To edit the source code (modify the program):

7. To edit, double click on MyFirstProgram_v1.cpp under the folder Source Files in the upper left-hand panel if it is not already displayed in the main panel. This will load your file into the main panel.



8. Modify the "Welcome to C++!!" data to read something a little different, say "Hello there! Welcome to C++!!"

To *recompile* the source code of this project:

9. To recompile, select the **Build** menu in the top menu bar.
10. Select the option **Rebuild Lab00** (assuming that Lab00 is the name of your project).
11. A lower window will show the results of this process.

You wish to *rerun* the executable file created in the compilation process:

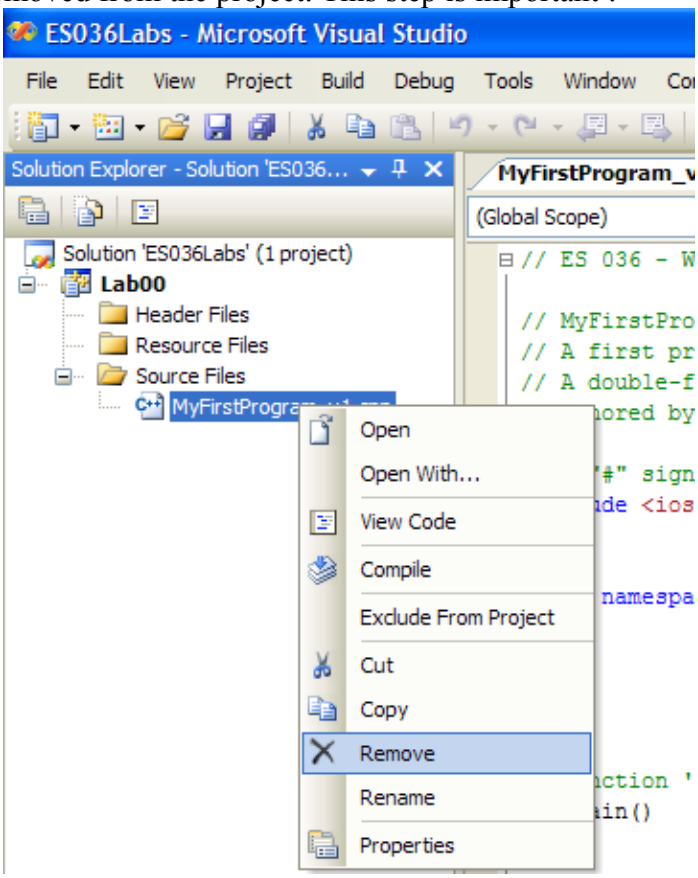
12. To run (execute), select the **Debug** menu in the top menu bar.
13. Select the option **Start Without Debugging**.
14. A black 'DOS-type' window appears showing the *new* output of your program.

Reviewing the Visual Studio Environment

Now let's try everything again from scratch!

1. Go to the ES036 web site and download to your network disk area (H:\ES036labs\Lab00) the file ArithmeticOpearation.cpp.
2. We need to remove the existing source file to add the new one. To do that right click on the

existing source file and select remove. Removing does not mean that you are deleting the file. The file is just removed from the project. This step is important[3].



3. Add the file ArithmeticOpearation.cpp as the source code.
4. Compile and run this code. Input (type and hit enter key) the integer values 4 and 2 and see the output.

Modifying the File ArithmeticOpearation_lab1.cpp

5. Uncomment one 'result' statement at a time while others will be commented out and check the output with integer values 4 and 2 respectively.
6. This time, try to use integer values 2 and 4 respectively and check the output for each case. Find a reasonable answer, since the results are not same as the previous set
7. Try to uncomment the second **`cout`** statement and comment out the first one and then run the program to check the output. In both cases, the output will be the same. This will give you an idea of the `endl` function. Make a reasonable guess and be ready to respond to TA's question on this.

You can practice with a couple of more programs downloaded from the course web site! Possible things to try:

a) If you modify a file with a deliberate error, you can see what sort of message this error generates when you try to recompile.

For example,

[3] If you get an error saying

```
1>Arithmetic_Operations.obj : error LNK2005: _main already defined in MyFirstProgram_v1.obj
```

you have more than one source file containing a main function. Remove all except the one you want.

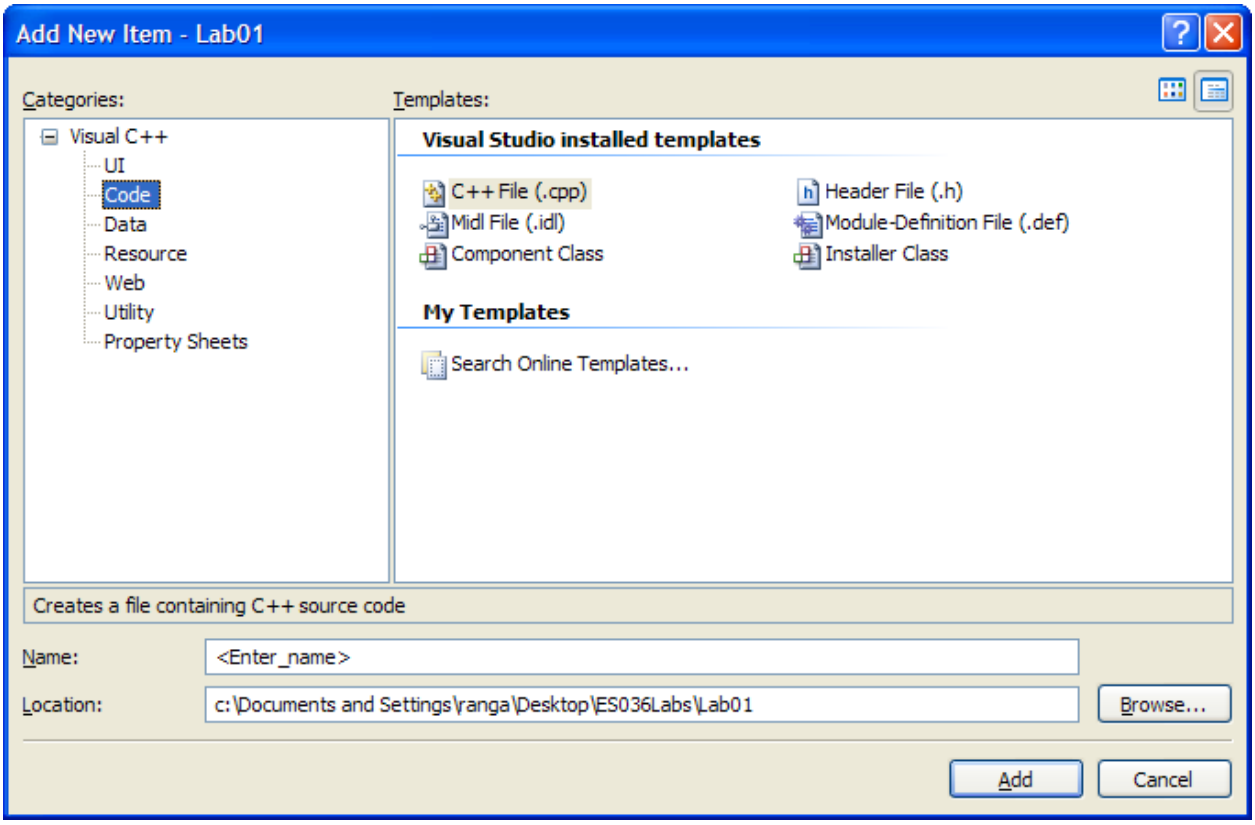
- try omitting a semicolon from the end of a statement (and then recompiling).
- try a different integer name

b) Can you start a new program from scratch? (use the editor to open a new file, then edit the file, compile and run)

A.Creating a New Program from Scratch

After completing this section you will be able to create a new program in C++.

1. Remove existing source files. (If you do not already have a project, you need to create one. If you followed this lab from the beginning, you should have a project called Lab00 within the solution ES036Labs.)
2. Click **File** in the top menu bar of Visual Studio, and then select the **Add New Item** option.
3. Select C++ File



4. Type in the code listed below.
5. Compile the program and execute it.

```
/*************************************************************************
                    Name: Your Name
                    Date:

This program computes the distance between two points

*************************************************************************/

#include <iostream>
#include <cmath>

using namespace std;

int main()
{
      //Declare and initialize objects
      double x1(1), y1(5), x2(4), y2(7),
               side_1, side_2, distance;

      //Compute the sides of the triangle
      side_1 = x2 - x1;
      side_2 = y2 - y1;
      distance = sqrt(side_1*side_1 + side_2*side_2);

      //Print the distance
      cout << "The distance between the two points is "
             << distance << endl;
      //Exit the program
      return 0;
}
```