16.317: Microprocessor Systems Design I Using Visual Studio

In this tutorial, we will create a sample project to help illustrate the use of Visual Studio. Note that this document assumes the use of Visual Studio 2012.

After starting Visual Studio, select **File→New→Project** from the main menu, or simply click the "**New Project …** " link on the welcome screen.

The dialog window that appears allows you to choose the type and name of your project. After selecting **Visual C++** in the list of templates on the left, choose **Win32 Console Application** from the list of project types in the middle. (Note that your choices may not exactly match those shown in Figure 1.) Use the boxes at the bottom of this window to specify a name and location for your project.

Important note: Your project name **will not need to match the assignment <u>name</u>. To avoid confusion, I suggest choosing a generic name for the project (for example, "program1") and then naming your source file appropriately later.</u>**



Figure 1: Creating a new Win32 Console Application by selecting "Visual C++" from the list of templates, "Win32 Console Application" from the list of project types, and specifying a name and location for your project. In this example, the project—not the source code file—is called "test_project" and stored in "C:\Users\Michael_Geiger\Documents\Visual Studio 2012\Projects\".

After accepting these settings, a window appears that you can use to set application settings. Click **Next**, then select the check box next to **Empty project**, which is under **Additional options**, in the following window. Click **Finish** to create your project.

Win32 Application Wizard - test_project						
Applicati	on Settings					
Overview Application Settings	Application type: <u>Windows application</u> <u>Console application</u> <u>DLL</u> <u>Static library</u> <u>Adduonal options</u> <u>Empty project</u> <u>Empty project</u> <u>Precompiled header</u> <u>V Security Development Lifecycle (SDL) checks</u>	Add common header files for:				

Figure 2: Initial application settings. After completing the steps shown in Figure 1, click "Next" in the first window that appears to reach this window—<u>do not choose "Finish."</u> Once you reach this window, be sure to select the "Empty project" check box before clicking "Finish".

To create your first C file, right click on **Source Files**, in the **Solution Explorer** window. Choose Add \rightarrow New Item. In the list that appears, choose C++ file (.cpp), then name your file. If the assignment requires you to write a program from scratch, the file name will be specified for you. You must choose a name with a .c extension so that the program will be saved as C source code, not C++ source code. See Figures 3 and 4 below.

						Solution Explorer Image: Constraint of the second seco
г <u>.</u>	New Item	Ctrl+Shift+A	1	Add		•
*0	Existing Item	Shift+Alt+A	B.	Class Wizard	Ctrl+Shif	ft+X
2	New Filter			Scope to This		
*	Class		ē	New Solution Explorer View		
*	Resource		Ж	Cut	Ctrl+X	
_			D,	Сору	Ctrl+C	
			â	Paste	Ctrl+V	on Team Exp., Class View
			×	Delete	Del	
			10	Rename	F2	rties + 4 X
			ş	Properties		e Files Filter Properties

Figure 3: Adding a new source file to your project

16.317: Microprocessor Systems Design I UMass Lowell

Add New Item - test_pro	oject	XX 8
▲ Installed	Sort by: Default 🔹 🏭 🧮	Search Installed Templates (Ctrl+E)
▲ Visual C++ UI	C++ File (.cpp)	Visual C++ Type: Visual C++ Creater a file containing C++ course code
Code HLSL Data Resource Web Utility Property Sheets Test Graphics ♪ Online	Header File (.h)	Creates a file containing C++ source code Visual C++
Name: Location:	testfile.c C:\Users\Michael_Geiger\Documents\Visual Studio 2012\Projects\te	st_project\test_project\ * Browse Add Cancel

Figure 4: Naming your source file. Note that the file name must end with ".c".

At this point, you should be able to write a C file with embedded assembly code. Note that assignments in the later part of the semester may require you to generate different types of projects (i.e., CLR code for more Windows-specific programming using graphical interfaces) or source files (i.e. assembly only, C++ with embedded assembly). Details about these files will be provided as needed.

Adding an existing file to a project: In some assignments, you will be given code to execute (and possibly modify). To incorporate an existing file into a project, choose Add \rightarrow Existing Item instead of Add \rightarrow New Item after right clicking on **Source Files**. You can then navigate to the directory containing the file and add it to your current project.