Winter 2016, CS 112
Programming Assignment 1 (100 points)

Submission Due: 11 January, Monday

1 Assignment Policies

Before introducing this assignment, let us review the policies which we will be enforcing for all assignments this quarter.

About software and hardware requirements: you are required to program in C++ using the GLUT toolkit and the OpenGL API. You may not use Java, C# or any other languages. Environment such as MFC and .NET are not allowed to use. You can program on any platform you prefer (Microsoft Windows, Mac, GNU/Linux, and any other Unix) and with any IDE supporting C++ and OpenGL (Microsoft Visual Studio, Xcode, Dev-C++, etc.). However, you are responsible to configure your machine and IDE to run your program correctly. Thus, claiming to lack access to a computer to complete the assignments is not an adequate excuse to gain an extension on the due date of an assignment.

About assignment material: all programming assignments related material will be uploaded to the EEE dropbox along with the assignment.

About cooperation and third party codes: on this and all future assignments you are encouraged to collaborate with your classmates, but you may not share final solutions and you must write up your own work, expressing it in your own words/notation. Please see the class web pages for further elaboration on the policies of this class, including collaboration. Third party codes are only allowed for extra features you want to introduce to your assignments and importing/exporting files (existing 3D models, textures, outputs) to/from your programs.

About submission deadline: the submission deadline will always be specified in the assignment. You do not need to submit it via EEE or email. There will be an appointment system through which you can make an appointment with the TA and show your work in person. We will send an email to the class and inform you about the time of presenting your work before the assignment deadline. You can do that by using your own laptop or one machine in the ICS 364 lab. Again, you are responsible to configure the machine you use to be a ready-to-go status.

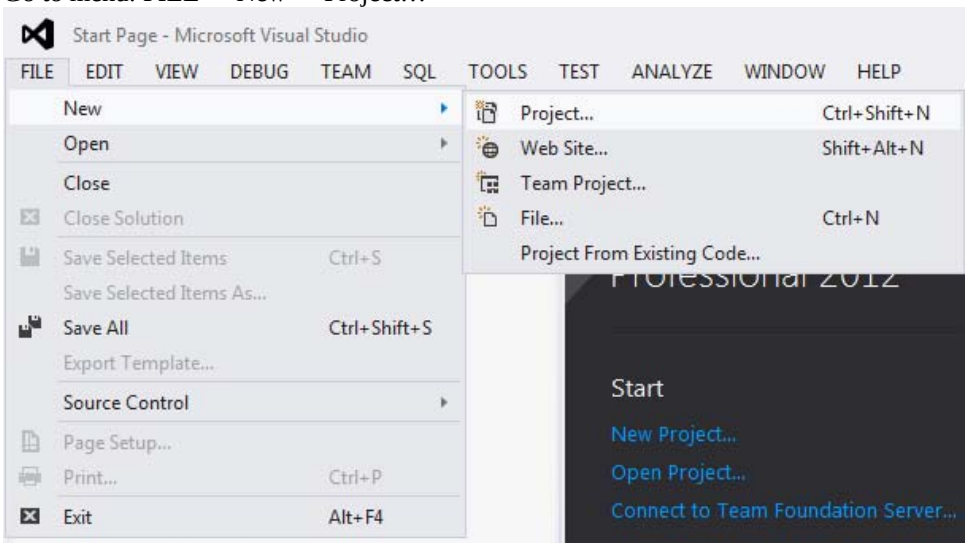
About grades: your grades will be determined based on your presentation. Fundamental mistakes, such as compiling errors, run time errors, and missing libraries or source files will lead to fail of an assignment. Lack of knowledge of how your own code works, on the other hand, will imply the possible cheating and reduce a significant portion of your grade even you have achieved all requirements. Fail to show up at your appointment time will be considered as late submission. Reduction of grades for late submission will be applied 10% for each weekday after the submission deadline which makes an assignment submitted two weeks later to be 0 grade regardless its quality.

2 Message Board (10 points)

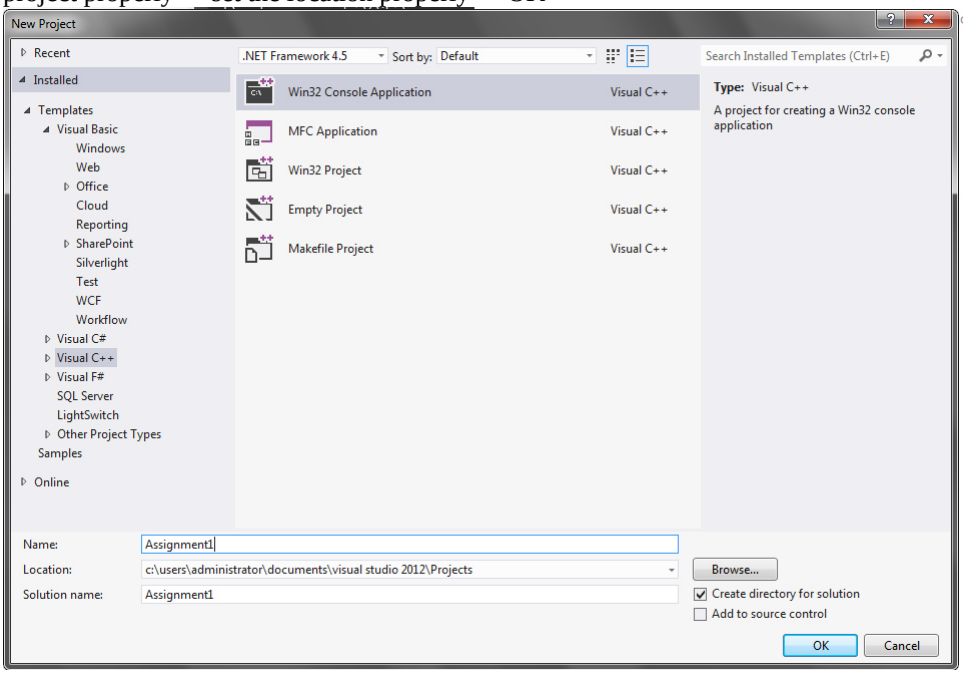
Access the class message board and post a message. For example, you might introduce yourself, ask or answer a question about the current homework, or about something covered in lecture, or share an interesting web site that is relevant to this class. The purpose of this assignment is simply to demonstrate that you know how to access the board and are able to do so. However, please do make the message germane to this class and with more content than “Hello, this is my message.”

3 Create a C++ Project (40 points)

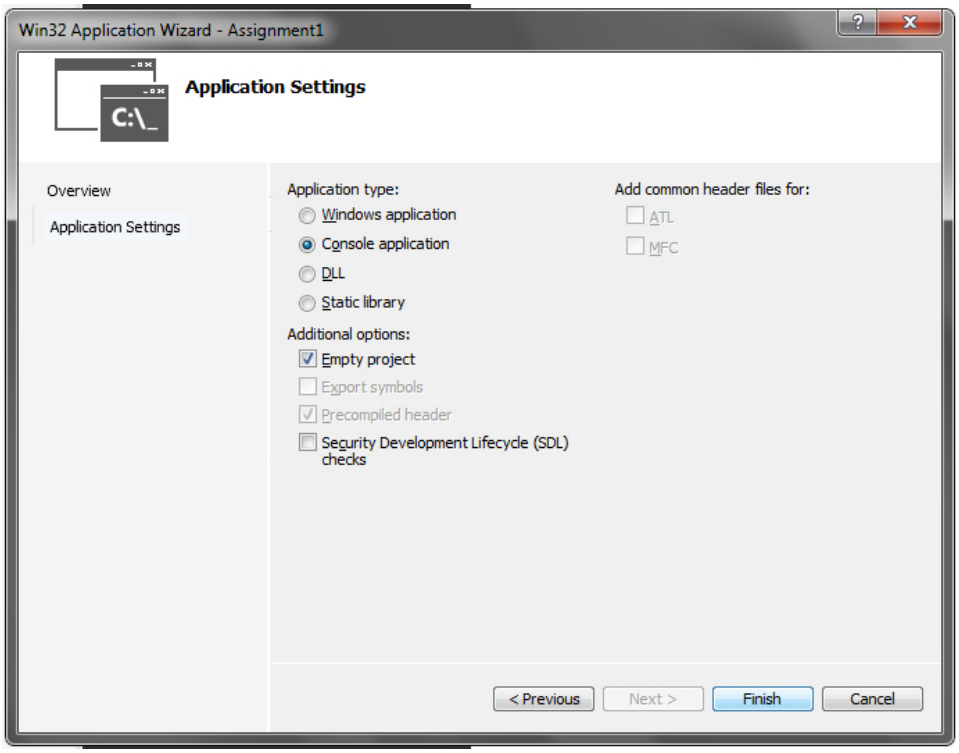
1. Login to EEE, go to drop box, download the CS112_A1Package.zip file. Extract it to your local space.
2. Create an empty C++ console project (command line tool in Xcode) and add the source files from this package to your project. Here is a tutorial for Visual Studio 2012. For other IDEs, there are plenty resources available through Internet. For example, http://www.onecore.net/dev-c-opengl.htm is useful for Dev-C++ users:
    a. Go to menu: FILE -> New -> Project…



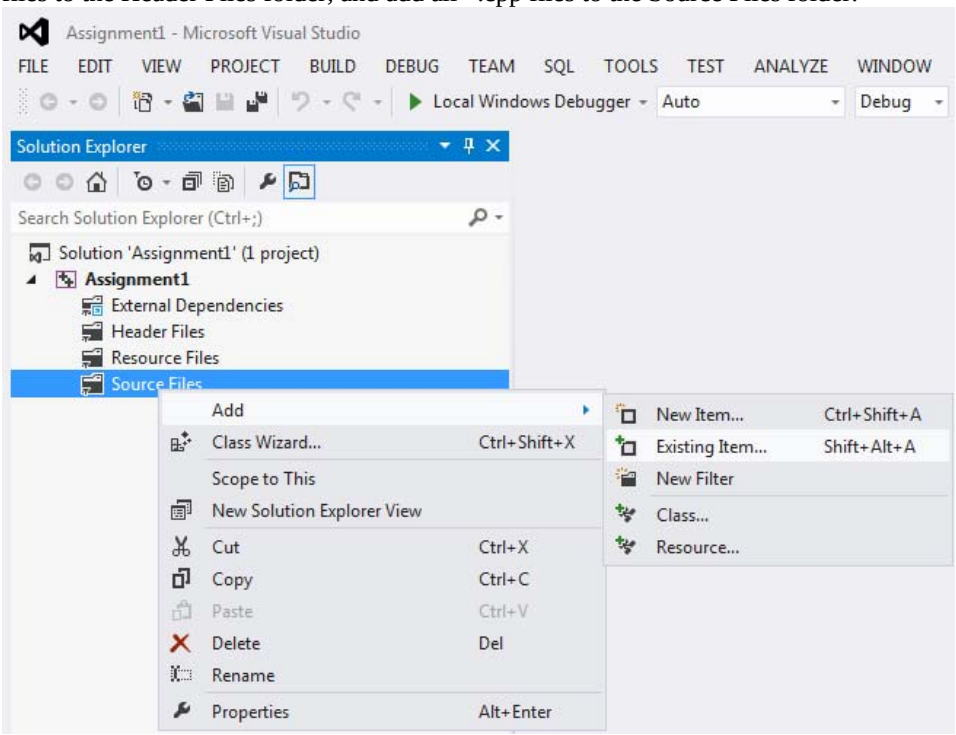
    b. Choose Visual C++ from the left panel -> Win32 Console Application -> name the project properly -> set the location properly -> OK



    c. Next -> Console application -> Empty project -> Finish



d. Right click on the folders in the Solution Explorer -> Add -> Existing Item…, add all *.h files to the Header Files folder, and add all *.cpp files to the Source Files folder.



e. Now you may notice a lot of syntax errors, we will solve them in the next part of this assignment.

4 Configuration of GLUT (50 points)

For Mac users:

1. GLUT should have already been installed in Mac OS X, so you do not need to download anything.
2. In your project navigator, select your project -> select your target -> Build Phases -> Link Binary With Libraries -> Add items -> OpenGL.framework -> Add
3. Repeat 2 for GLUT.framework.

For Windows users:

1. For Windows users, download GLUT from http://user.xmission.com/~nate/glut/glut-3.7.6-bin.zip and extract it to your local space.
2. Copy glut32.dll to C:\Windows\system32. Copy glut32.dll to C:\Windows\SysWOW64 if system32 does not exist. (This may happen for a 64-bit machine).
3. Copy glut32.lib to C:\ProgramFIles(x86)\Microsoft Visual Studio 11.0\VC\lib. (Assuming you are using VS 2012 and installed it in the default path)
4. Copy glut.h to C:\ProgramFIles(x86)\Microsoft Visual Studio 11.0\VC\include. Note: a few online tutorials may suggest you to copy it to …\VC\include\GL, which is not available by default for VS 2012.

For other IDE users, please search your IDE name along with glut for online help.

Once you have done this configuration, syntax errors you see from the previous part should have gone. Build and run your program, a window with a cube rendered in 3D should pop-up. Try possible interactions yourself.

5 Where to Get Help – Understanding the program

Besides that, you are strongly encouraged to search online and discuss with other students to go through the code yourself. This will save you a lot of time for your next assignment, since it will be built on this trivial program and you need to understand it well to add your own codes to it.

There are plenty of OpenGL tutorials, guides and references online. You will also find them extremely useful in future projects. Here are some examples for legacy OpenGL:

The red book: The Official Guide to Learning OpenGL http://glprogramming.com/red/
The blue book: OpenGL Reference Manual http://www.glprogramming.com/blue/

Perhaps you mutter to yourself '*heaven help me - what is all this madness!*' It is better to say this right at the beginning rather than three weeks into the course. If you are in this situation remember you can always post your problem in the course message board on EEE. Here you can get help from your peers. It is best to keep up to date with the discussions on the message board since many of your questions will be answered even before you know about the question.