

OpenGL, freglut, GLEW Installation Guideline

In order to write graphics programs for this course, you will need three things:

OpenGL: OpenGL is the graphics library specification we will use to create our graphics programs. Note that OpenGL is simply a specification, which means it just defines the behavior of all the different OpenGL implementations. There is no official implementation of the library to download and use. Instead, operating system and GPU vendors generally provide users with their implementations.

Window manager(freelut): OpenGL only deals with purely graphical tasks such as rendering, shading, etc. It does nothing to create and manage application windows or take input from the user. There are many window management libraries to serve this purpose such as GLFW or SDL, but during this course we will be using **freelut** for its simplicity. Freelut is a successor to an older library called glut, which stands for OpenGL Utility Toolkit.

Glew: OpenGL Extension Wrangler Library is a library that loads the supported OpenGL extensions for your platform. You will need it to load the supported OpenGL functions.

This document will assist you as you install OpenGL, freelut and GLEW for Ubuntu, Windows, or MacOS systems.

Linux (Ubuntu)

These instructions are for Ubuntu 20.04 64 bit. You might need to change the commands if you are using a different distro.

OpenGL

If you have a GPU on your machine and you have sufficiently updated drivers, your system should already have OpenGL version 4.x available to you. You can check this by using these commands:

```
sudo apt install mesa-utils  
glxinfo | grep "OpenGL version"
```

If you do not have OpenGL 4.1 or later available on your device but you have a GPU, you should attempt to install the latest drivers from your GPU vendor. Here are some links to let you do that:

- AMD: <http://support.amd.com/en-us/download/linux>
- Nvidia: <http://www.nvidia.com/object/unix.html>
- Intel: <https://01.org/linuxgraphics>

After installing the latest drivers, please check again to make sure you have OpenGL version 4.1 or later available to you.

Freelut

To install freelut you can use the command:

```
sudo apt install freelut3-dev
```

Glew

To install glew you can use the command:

```
sudo apt install libglew-dev
```

Compilation

Once you have all three libraries available on your machine, you can compile your graphics programs as follows:

```
g++ example.cpp -o example -lGL -lglut -lGLEW
```

Windows

OpenGL

If you have a GPU on your machine and you have updated drivers, your system should already have OpenGL version 4.x available to you. You can check this by using this free application:

<http://realtech-vr.com/home/glview>

If you do not have OpenGL 4.6 or later available on your device but you have a GPU, you should attempt to install the latest drivers from your GPU vendor. Here are some links to let you do that:

- AMD: <https://www.amd.com/en/support>
- Nvidia: <http://www.nvidia.com/Download/index.aspx>
- Intel: <http://www.nvidia.com/Download/index.aspx>

After installing the latest drivers, please check again to make sure you have OpenGL version 4.6 or later available to you.

Freeglut & GLEW using NuGet

First, create a Win32 Console Project: File > New > Project. From the dialog left pane select Visual C++, then select Console Application. Choose Empty Project and uncheck the SDL option. Then, you can use NuGet Package Manager to add dependencies to your project. It allows a very easy setup of freeglut and glew. To install freeglut and glew with Nuget, in Visual Studio select **Tools -> NuGet Package Manager -> Package Manager Console** and then type "Install-Package nupengl.core". Both libraries should then be available in your project environment.

Troubleshooting for Windows

- To disable precompiled headers for your project do; Select your project, use the "Project -> Properties" menu and go to the "Configuration Properties -> C/C++ -> Precompiled Headers" section, then change the "Precompiled Header" setting to "Not Using Precompiled Headers" option.
- When running in debug mode, If you see "cannot find or open the pdb file" error, Go to Tools -> Options -> Debugging -> Symbols and set the checkmark. Prefer running without debugging (Ctrl + F5) to avoid this.
- If you get an error regarding "_CRT_SECURE_NO_WARNINGS," Go to Project -> Properties and then All Configurations -> C++ -> Preprocessor. Select the first field, and add "_CRT_SECURE_NO_WARNINGS" without quotes to the first field. Make sure to add a semicolon (;) after your addition.

Alternative Freeglut installation for Windows (Advanced)

If you do not wish to use NuGet, you can download the freeglut library from the following link: <https://www.transmissionzero.co.uk/software/freeglut-devel/> . You will have to then configure your project environment to include the library. This step will vary based on your IDE/compiler.

Alternative GLEW installation for Windows (Advanced)

If you do not wish to use NuGet, you can download the glew library from the following link:

<https://sourceforge.net/projects/glew/files/glew/2.1.0/glew-2.1.0-win32.zip/download> . Like you did with freglut, you need to configure your compiler/IDE so that it can access the necessary .dll, .lib and .h files.

MacOS

* These instructions were made for MacOS Monterey. If you use a previous version of MacOS you might run into issues. We suggest updating your system before attempting the installation. **GLEW** Install glew using Terminal and Homebrew. You can get homebrew from <https://brew.sh/> if you do not have it installed:

```
brew install glew
```

OpenGL & Freeglut

Install Xcode from App Store

Launch Xcode, from the File menu -> New -> Project

Select macOS then Command Line Tool

Select C or C++ as language

Enter name, click Next, choose folder location and click Create.

Click on Project on the left pane, Select Build Phases, under Link Binary with Libraries click + button

Use search, find and add **OpenGL** and **GLUT** frameworks

Add **libGLEW** by clicking **Add Other**

Default path for brew installs is `/usr/local/Cellar`

Select **libGLEW.{version}.dylib** (e.g. **libGLEW.2.2.0.dylib**) under:

`/usr/local/Cellar/glew/{version}/lib/` (e.g. `/usr/local/Cellar/glew/2.2.0/lib/`)

Add your header and source, shader files.

To use OpenGL 4.6 the following init mode should be supplied in `glutInitDisplayMode`:

```
glutInitDisplayMode( GLUT_RGBA | GLUT_DOUBLE | GLUT_DEPTH );
```

Troubleshooting for macOS:

- If your program window appears black when you first launch your app, try to resize the window, this forces it to redraw.