



# Graphics Library Installation: OpenGL / Metal

You probably already have OpenGL (Linux/Windows) or Metal (Mac) on your machine. However, the operating system may not have set you up with the best version of the graphics driver for your particular hardware. You don't want to be stuck with OpenGL 1.X! We'd like to have at least OpenGL 3.2.

So make sure you have recent graphic card drivers. Download the specific driver for your card from NVidia, ATI, or Intel rather than the default Linux or Microsoft Windows driver.

*WARNING: OpenGL features beyond OpenGL 1.1 generally do not work over Remote Desktop. Similarly, running graphics/GPU programs in a virtual machine is often problematic.*

---

## OpenGL Graphics Libraries Installation

If you're using Linux or Windows, you'll need to install the following libraries:

- [GLFW \(for creating & managing OpenGL contexts\)](#).
- [GLEW \(The OpenGL Extension Wrangler Library\)](#).
- [GLM \(OpenGL Mathematics\)](#).

Please follow the detailed instructions below for each OS.

*Please take notes as you work. Share any instructions corrections, roadblocks, and workarounds on the Submitty Discussion Forum (to help your classmates, and help maintain the instructions and homework codebase. Thanks!*

---

## GNU/Linux

## Ubuntu 20.04

- First install or upgrade [CMake](#) to at least version 3.1.1.

Download `cmake-3.X.X-Linux-x86_64.sh`.

Run it.

In your `.bashrc`, add

```
export PATH="$PATH:path_to_the_cmake_bin_folder"
```

Confirm that you have the correct cmake by typing:

```
cmake --version
```

- Install the necessary packages:

```
sudo apt install libglfw3-dev glew-utils libglew-dev libglm-dev  
sudo apt install libxrandr-dev xorg-dev
```

## Arch

```
pacman -S glfw-x11 glew glm
```

## Fedora

```
yum install glew glew-devel glm-devel glfw-devel
```

---

# MacOS

*No longer supports OpenGL. Instead we'll use Apple Metal.*

- We recommend the [Homebrew](#) package manager. After brew is installed, just type:

```
brew install cmake
```

- Make sure you've installed XCode and XCode's command line tools (Google to find instructions).
- Update to MacOS Mojave 10.14.2 (or higher) and XCode 10.1 (or higher).
- In a terminal, run:

```
xcode-select -p
```

It should print:

```
/Applications/Xcode.app/Contents/Developer
```

If it doesn't, reset the path with:

```
xcode-select -r
```

- *NOTE: After installing CMake, XCode, XCode Command Line Tools, or after updating/resetting your xcode path it's probably a good idea to close and restart your terminal to make sure the changes take effect. It may or may not be necessary to also reboot /restart your computer.*

*Also, you may need to cleanup / delete your entire build directory and re-do the cmake and make commands if you have tweaked the configuration.*

## Windows

- Create a new folder on your machine named "C:\GraphicsLibraries\" and subfolders named:
  - "C:\GraphicsLibraries\include\"
  - "C:\GraphicsLibraries\include\GL\"
  - "C:\GraphicsLibraries\include\GLFW\"
  - "C:\GraphicsLibraries\lib\"
  - "C:\GraphicsLibraries\bin\"
- Download and unzip the following files and place them as instructed below.
 

*NOTE: Make sure you are consistent and grab either all 64-bit or all 32-bit versions of the libraries. You probably want to choose the 64-bit versions.*

  - Download a recent zip of the GLM headers (a headers only libraries) from <http://glm.g-truc.net/> under "Downloads". Extract the compressed file, which makes a glm folder. Inside of that folder is another glm folder. Copy the inner glm folder to "C:\GraphicsLibraries\".
  - To check:* The file "glm.hpp" should be in this folder  
"C:\GraphicsLibraries\glm\"
  - Download the GLFW Windows binaries (probably 64-bit) from <http://www.glfw.org/download.html> and unzip the files.
 

*NOTE: Make sure to select 64-bit and also select the version that matches your version of Visual Studio.*

Put "glfw3.h" and "glfw3native.h" in "C:\GraphicsLibraries\include\GLFW\"  
Put "glfw3.lib" in "C:\GraphicsLibraries\lib\"
  - Download the GLEW Windows binaries from <http://glew.sourceforge.net/> and

unzip the files.

Put "glew.h" and "glxew.h" "wglew.h" into "C:\GraphicsLibraries\include\GL\"

Put "glew32.lib" and "glew32s.lib" in "C:\GraphicsLibraries\lib\"

Put "glew32.dll" in "C:\GraphicsLibraries\bin\"

- Edit the "PATH" variable for your account (*not the system variables*) to append the location of the dynamic libraries (dll's) "C:\GraphicsLibraries\bin\". (Google search if you don't know how to do this already). You might need to make a new variable if you don't already have a PATH.
- The CMakeLists.txt line in each homework:

```
set(CMAKE_PREFIX_PATH ${CMAKE_PREFIX_PATH} "C:\\GraphicsLibraries")
```

points your compilation at these directories. You could choose to put all of the downloaded library files in another location, and then specify the new path on the command line when building each project for the first time, e.g.:

```
cmake ../src -D CMAKE_PREFIX_PATH="${CMAKE_PREFIX_PATH} C:\\My\\Path\\Here"
```