

Computer Graphics Shaders Project Notes



Oregon State
University
Mike Bailey

mjb@cs.oregonstate.edu



This work is licensed under a [Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License](https://creativecommons.org/licenses/by-nc-nd/4.0/)



Why Are These Notes Here?

These notes are here to:

1. Help you setup and run your projects
2. Help you get everything in the right format for submission
3. Help you get a **better grade** by doing all of this correctly!

better grade!

better grade!

better grade!

better grade!

better grade!



Office Hours

- The TAs and I will host Office Hours on Zoom. We will set those hours as soon as we see (from the web form on the Class Resources Page) what hours people are most likely to attend.
- There are no Office Hours exclusively assigned to particular sections. *Anybody* can go to *any* Office Hours.
- Try different people's Office Hours. The TAs and I all have different OHs days, times, and styles. See whose OHs best fit your schedule and needs.
- If Office Hours are not working for you, let me know right away. We will find a way to make them work for you.
- We enthusiastically encourage Office Hour “**Lurkers**”. That is, it is OK to come to OHs without any specific questions and just listen to the other questions and answers. Because we will be on Zoom, you can get other things done while you listen in. I have noticed that lurking on OHs is an extremely effective way for you to catch hints and advice on doing the projects. Take advantage of this! Others will. Don't be left out.

Running Your Projects

- As this is a computer graphics course, you need to find *somewhere* to run your programs that have graphics display hardware on them. ***flip is not one of these places.***
- If you don't have access to your own graphics system, then you can use what we have at OSU. **On-campus users** will have access to Windows and Linux systems on-campus.
- **Off-campus users** can access our Citrix system. This is a good solution, but not a great solution. Your life will be smoother if you can find a local graphics system. See the next slide.



Compiling and Running Your Projects via Citrix

5

Citrix allows you to get remote desktop access to other systems. To put Citrix on your own machine, go to <https://citrix.com/downloads>, select your operating system, and click on **Download**

Click **Add Account** and enter your ONID email (e.g., jgraphics@oregonstate.edu)

Click on **Continue** to configure your account

Enter your ONID email and password in the dialog box, and click **Logon**

When you run Citrix, click on the **Desktops** icon at the top

Go to: <https://it.engineering.oregonstate.edu/citrix/> for more information



Running Your Projects on Windows

6

- Get the **ShaderSampleWindows.zip** file from our Class Resources Page
- Un-zip it
- Double-click on the .sln file
- Select **Build**→**Clean Solution**
- Select **Build**→**Build Sample**
- Select **Debug**→**Start Without Debugging**



Getting Visual Studio for your own Windows System



7

If you are on your own **Windows system**, and don't have Visual Studio 2022 already, you can get it going to:

<https://azureforeducation.microsoft.com/devtools>

Click the blue **Sign In** button on the right.

Login using your onid@oregonstate.edu username and password.

I recommend you get **Visual Studio 2022 Enterprise**. Don't get Express.

Note that *vscode* is not a compiler. It is a way to interface to your file system.

Once you have Visual Studio, download the file **ShaderSampleWindows.zip**, unzip it on your system, and double-click on the **.sln** file



Compiling and Running on Your own Linux System

8



- Get the **ShaderSampleLinux.tar** file
- Un-tar it: **tar -xvf SampleLinux.tar**
- Then **cd ShaderSampleLinux**
- Then **make sample**
- Then **./sample**

If your system does not have GLEW or GLUT, do this:

```
sudo apt-get install libglut-dev  
sudo apt-get install libglew-dev
```



Compiling and Running on Your own Mac System

9



Get the **ShaderSampleMac.tar** file

Un-tar it: **tar -xvf SampleMac.tar**

Then **cd ShaderSampleMac**

Then **make sample**

Then **./sample**



Project Turn-in Procedures

10

Your project turnins will all be electronic on *Canvas*. We will not be using Teach.

Your project turnins will be done via Canvas and will consist of:

1. Source files of everything (.cpp, .vert, .frag)
2. A report in PDF format.
3. No other files!

Submit these files separately. Don't zip or tar (etc...) anything!

Electronic submissions are due at 23:59:59 Pacific Time on the listed due date.

Your PDF report needs to include:

1. A title area on the first page: your name, email, project number, and project name
2. A description of what you did to get the display that you got
3. A couple of screen captures to show your program in action
4. An accessible web link to a video showing your program in action

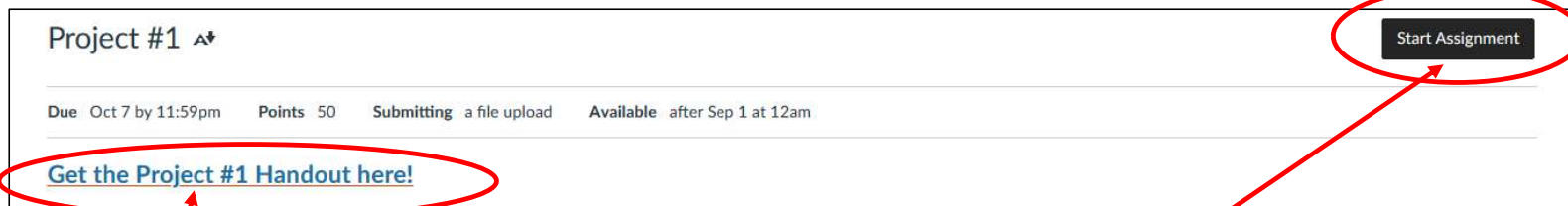
Your project will be graded and the score posted to Canvas.

0
Cor

If you did not get full credit, your grade will have an attached Canvas note telling you why.

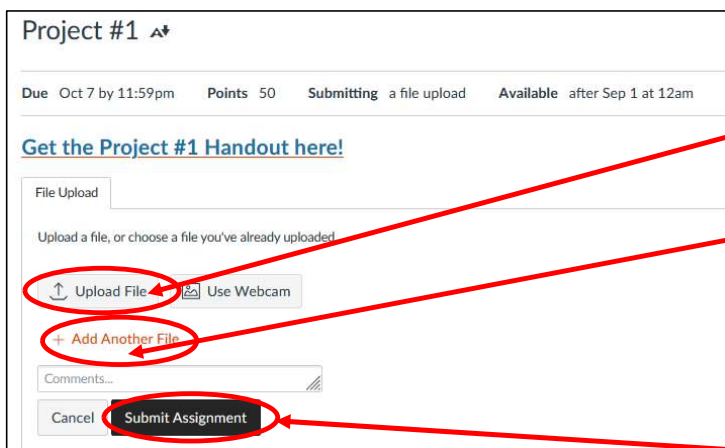
How to Turn In a Project on Canvas

On Canvas, the assignments will look like this.



Click this link to see the handout. I do it this way so that, if the handout needs to be clarified or corrected, I can do it quickly and instantaneously for all sections.

When you are ready to submit your files, click the **Start Assignment** button. That will bring up a dialog box that looks like this:



1. Click the **Upload File** button, browse to your first file (the order doesn't matter), and select the file.
2. Click **Add Another File**, click the new **Upload File** button, browse to the second file, and select the file.
3. Do this as many times as you need to (only submit your PDF and your code source files (.cpp, .vert, .frag)).
4. When the files are all uploaded, click on **Submit Assignment**.



Oregon State
University

Computer Graphics

Project Videos

In order to get your project graded, you need to make a video of your program in action:

- You can use any video-capture tool you want. If you have never done this before, I recommend **Kaltura** or **Zoom**.
- You can get Kaltura instructions here: <http://cs.oregonstate.edu/~mjb/cs557/Handouts/kaltura.1pp.pdf>
- If you use Kaltura, be sure your video's permissions are set to **Unlisted**. If the permission isn't set to **Unlisted**, then we won't be able to see it and we can't grade your project. *You need to proactively do this through <http://media.oregonstate.edu> -- this is not what the default setting is.* A good way to see if this is set properly is to give the link to a friend and see if they can open the video.
- Although not required, we love it when you narrate your video so you can tell us what you did. We have found that our grading of your project is far more fair and accurate when there is narration..
- **Don't make your video overly long! Show what we need to see to grade it. Do not walk us through your code!! If we want to see your code, we will look at your .cpp file.**
- Be sure that you include the accessible web-link to your video in your PDF report!

Project Video

- If you use Kaltura, be sure your video's permissions are set to **Unlisted**. *You need to proactively do this -- this is not what the default setting is.*
The best way to do this is to go to <http://media.oregonstate.edu> , then:
 1. Login
 2. Go to My Media
 3. Click on the video
 4. Using the **ACTIONS** pull-down menu on the right, select **Publish**
 5. Select **Unlisted**
 6. Select Save
- If the permission isn't set to **Unlisted**, then we won't be able to see it and we can't grade your project
- Don't make your video overly long! Show what we need to see to grade it.
- **Do not walk us through your code!!** If we want to see it, we will go look at it.
- Be sure that you include the web-link to your video in your PDF report!

Silly Ways to Lose Points on Your Project

- You didn't put your name and email on the title page of the PDF report (-5)
- You submitted some other file type for your report other than a PDF (-5)
- You zipped or tarred (etc.) some of your submission files (-5)
- You didn't put a link to your video in your PDF report (-5)
- You didn't change your video permission to *Unlisted* (-5)

Basically *anything* (that isn't my fault) that makes it take longer for me to score your project is a loss of 5 points.



Bonus Days

Projects are due at 23:59:59 on the listed due date, with the following exception:

Each of you has been granted **5** Bonus Days, which are no-questions-asked one-day extensions which may be applied to any project, subject to the following rules:

- No more than **2** Bonus Days may be applied to any one project
- Weekends and holidays count as “days late”
- Don’t worry if *teach* tells you it’s late because it is between 23:30:00 and 23:59:59. But, *after* 23:59:59 on the posted due date, **it’s late!**
- Really what I do is look at your turnin **date**. Your turnin date minus the due date is how many “days late” your project is.
- Bonus Days cannot be applied to tests or quizzes
- Bonus Days cannot be applied to the Final Project Proposal, the Final Project, or the CS 557 Paper Project

If you turn in a project three or more days late, your score is a zero.

If you turn in a project late and you don't have enough Bonus Days left to cover it, your score is a zero.

You don't need to ask me, or even tell me, that you are using Bonus Days. Just turn your project in two-or-less days late. I have a script that will check your turn-in date and deduct the Bonus Days.

It is up to you to track how many Bonus Days you have used up. However, I also keep a spreadsheet of your Bonus Days. If you lose track, send me an email and ask.

Bonus Days

To use Bonus Days on a given project:

- You don't need to let me know ahead of time.
- Turn-in promptness is measured **by date, not by time**. After 23:59:59 Pacific Time on the posted due date, **it's late!**
- *Canvas* will allow your turn-in, no matter when you do it. But we will only grade it if it is turned in ≤ 2 days after the due date.
- I run a script to identify the projects that need to have Bonus Days deducted.
- I keep my own spreadsheet of who has used how many Bonus Days. If you are not sure how many you have left, send me an email and ask.





Keys to Succeeding on Your Programming Projects, I (based on me having been through this a lot with your predecessors)



17

Start early! Stop rolling your eyes at me. You don't have to *finish* early but make a small start right away. Get the project folder setup. Bring in any other files you will need. Make sure it still compiles. Once you are over the speed bump of *starting*, you will be surprised by how easy it is to do little bits of the project during those nooks and crannies of time we all have. Also, you will be surprised by how much more meaningful all those hints are that you are going to be hearing in class, Live Lecture, and Office Hours. Starting early really does smooth the path to finishing.

Lurk in Office Hours! Because they are on Zoom, you can still be doing other things but just lurk and listen for hints or for questions you realize you need to know too.

Work through the projects a small chunk at a time! Don't try to do the whole project at once. It won't work, and you will be stuck with a big, fat, confusing debugging mess. Do something really small and get that to work. Draw an object all by itself at the origin. Use simple, trusted, objects at first (e.g., a sphere). Then add another small piece. Then another. Then another. If a step doesn't work, you will know *exactly* what additional code was at fault.

