

CIS 565: Final Project

Fall 2013

The final project gives you an opportunity to embark on a large GPU programming endeavor of your choice. You are free to select an area in graphics, GPU computing, or both. You can reproduce the results of recent research, add a novel extension to existing work, or implement something completely original.

Expect this to be 2-3x more work than the hardest project this semester.

Guidelines

- Form teams of two. Each member of a team will receive the same grade. Teams of one or three will be considered in extreme cases.
- Use github. We encourage, but do not require, you to develop in a public repo as open source.
- Use twitter to share screenshots, slides, and announce milestones.
- Language, APIs, and target platform are up to you.
- You are allowed to use existing code and libraries. These should be properly credited and your own code should be easily identifiable.

Grading

This is worth 40% of your final grade. Grading is the same as the projects: self-grading with student/instruction split 70/30%.

Alpha Presentation - 15%

Final Presentation - 15%

Paper - 15%

Code - 55%

Timeline

Monday 11/18: Pitch

Sign up for a timeslot:

<https://docs.google.com/spreadsheet/ccc?key=0AooWFPtY81iUdE96X0Y5NC1KdkhIZDI5bFNoV1pRM0E&usp=sharing>

Your project pitch is a 15 minute meeting with Patrick and Liam and a write-up no longer than one page that includes an overview of your approach with specific, measurable goals. First, focus on why there is a need for your project. Then describe what exactly

you are going to do. In addition to your write-up, you are encouraged to provide supplemental figures, images, or videos.

Before the meeting:

- Email your one page pitch and any supplemental material to Patrick beforehand
- Bring a hard copy of your pitch

After the meeting:

- Push your pitch to a new GitHub repo for your project
- Send a tweet about launching your project
- Email these to cis-565-fall-2013@googlegroups.com

Monday 11/25: Alpha Presentation (15%)

An in-class seven minute presentation of your work-in-progress after one week of research, design, and development. Your presentation can include a few slides, videos, and screenshots. You must demonstrate working code (videos and screenshots are OK; it doesn't have to be live).

Clearly state what you are trying to achieve, what you have done so far, and the challenges that lie ahead.

Present as a team. For a great example, see <http://www.youtube.com/watch?v=OTCuYzAw31Y>

After class:

- Push your presentation to your GitHub repo
- Send a tweet about it

Friday 12/13 (date tentative; time TBA): Final Presentation

A course happy hour will follow final presentations.

By midnight Thursday 12/12

- Push the following to GitHub
 - Slides for final presentation (15%)
 - Video
 - Paper (15%)
 - Final code (55%)
- Announce this on twitter

Presentation - seven-minutes showcasing your accomplishments and presenting your results with a live demo, performance numbers, and future work.

Video - Two to four minutes in length to show off your work. You can do a voiceover; play music and use descriptive text; or a combination of both. Your video should complement your paper and clarify anything that is difficult to describe in just words and images. Your video should both make us excited about your work and help us if we were to implement it.

On Windows, Windows Live Movie Maker can create videos compiled from other videos, images, and audio:

<http://explore.live.com/windows-live-movie-maker>

You can capture video with a tool of your choice, such as:

- FRAPS: <http://www.fraps.com/>
- Microsoft Expression Encoder:
http://www.microsoft.com/expression/products/Encoder4_Overview.aspx

Paper - up to six pages in length, included performance analysis, following the [JCGT guidelines \(examples\)](#) . This must be written in LaTeX using the JCGT template provided here:

<http://jcgt.org/files/jcgt-template.zip>

On Windows, proTeXt can be used to edit LaTeX and build a pdf:

<http://www.tug.org/protext/>

If you are new to LaTeX, check out these tutorials:

- <http://www.electronics.oulu.fi/latex/index.html>
- <http://www.tex.ac.uk/tex-archive/info/lshort/english/lshort.pdf>

Code - should be clean, documented, and tested.