Lecture 13

SIGGRAPH 2019 Trailer!!

- https://www.youtube.com/watch?v=EhDr3Rs5fTU

- A longer version of one of the clips (from one of my lab mates):
    - https://www.youtube.com/watch?v=INri-x2nK7o

SIGGRAPH 2019 Trailer!!

- Simulations like that used advanced Numerical Linear Algebra
    - 200 level courses in the Math department.
- Some of our math concepts help prepare you to take that
- Look at how these concepts from QR / SVD matrix decomposition use our familiar rotation and scale matrices:
    - https://en.wikipedia.org/wiki/Householder_transformation
    - https://en.wikipedia.org/wiki/Givens_rotation

Announcements

- Who are the teams of three?
- Remember to create your team GitHub repo at: https://classroom.github.com/g/QnMzcLaS

Homework 4 Tips
(Piazza clarifications)

JavaScript Tips

JavaScript Tips

- You might find our JavaScript crash course only got you through A3; you need a little more JavaScript now.
  - for-of loops
  - concepts like iterators, pointers, and more are also borrowed from C++. Don't worry; ordinary "for" loops also work fine for your own code.
  - how iterator functions work
  - several project coding pitfalls that we encountered during office hours
  - tiny-graphics's high coverage of new JavaScript features

Finish Last week’s slides (Blinn Shading)

GuerrillaCG Tutorial: Bump Map and Displacement Map

https://www.youtube.com/watch?v=1mdR2imNeZI

Texture Sampling and Mip Maps

Udacity Course: Mip mapping explanation

Start at this video, #376 in the Udacity Course, and watch until video #383:

https://www.youtube.com/watch?v=ZlzXX8cLAds&list=PLAwxTw4SYaPlaHwnoGxJE7NFhEWRCIyet&index=376

Scene Timing Ideas

Advice on organizing time during scenes that have several distinct events happen sequentially

Freeze your Scene

- You can set `program_state.animate = 0` anytime. This freezes animation_time.
- You can do this inside your code at a certain time, as a debugging / diagnosis trick.
- If you're paused in the debugger inside your scene, you can just enter that line into the console to freeze your Scene.
- Easiest option: You could make a button in your Scene that does it.

Scene timing ideas

- Use “if” statements for switching scenes
- Keep multiple copies of the **animation_time** variable with different offsets subtracted from it.
- Name the copies like "**time since (scene #17) started**"
  - If it’s **negative**, the scene shouldn’t start yet, so skip those lines for this frame.
  - If it's **positive**, you know to proceed with the lines of code that animate that scene
  - If it’s **too positive**, it’s past the scene’s end time, so manually overwrite it to some negative number, making the scene go away again.

Scene timing ideas

- Concrete example:

...In display(), set up lots of variables like:

```
var     time_sinceFallingSceneStart = graphics_state.animation_time - 5000;   // Start the scene 5 seconds in.
    if( time_sinceFallingSceneStart >  7000                                    )        // This scene will last 7 seconds...
            time_sinceFallingSceneStart = -1;           // ...after which, it will go away again.
```

- Farther down:

```
if( time_sinceFallingSceneStart > 0 )
      ... // Go on with displaying the falling scene
```

Scene timing ideas

If your program is interactive:

- May want less of a time-dependent scene system -- and more of a **finite state machine (FSM)**
- Conditionals in your display() function would consult the **state** before proceeding with any drawing code sections.
- Certain key inputs will change your state variable to a different enum value, causing those if's to act differently.
    - Multiple offset copies of your time variable will again be needed to track how long each animation has been going on for, so you can change state once it's done.