v

# ISCG7441 – Advanced Game Programming

## Documentation and Journal

* Aquila Halpé
* Jesse Schollitt

### Contributors:

Commit history (as of 6:25 p.m. 28/06/2017)

|  |  |  |
| --- | --- | --- |
| Aquila Halpé | 21 minutes ago | added some comments on skybox class |
| Jesse Schollitt | 49 minutes ago | Added keybinds for the user inputs. Added mouse movement for pitch and roll. Pinned mouse to game window (limits max rotation speeds). Added cockpit overlay. |
| Jesse Schollitt | 3 hours ago | fixed untracked files |
| Aquila Halpé | 3 hours ago | TGUI BOI |
| Aquila Halpé | 4 hours ago | Merge branch 'master' of https://github.com/scholj05/AdvGameProg-Assignment2 |
| Aquila Halpé | 4 hours ago | basic TGUI class implemented |
| Jesse Schollitt | 4 hours ago | Merge branch 'master' of https://github.com/scholj05/AdvGameProg-Assignment2 |
| Jesse Schollitt | 4 hours ago | Deleted vehicle class (can't fix in time). |
| Jesse Schollitt | 4 hours ago | Removed vehicle class (not enough time to fix it). |
| Aquila Halpé | 5 hours ago | did some work on making the gradients look more realistic in the heightmap rendering |
| Aquila Halpé | 6 hours ago | Merge branch 'master' of https://github.com/scholj05/AdvGameProg-Assignment2 |
| Jesse Schollitt | 18 hours ago | removed some redundant code and fixed some comments |
| Jesse Schollitt | 18 hours ago | Added ocean. Changed colouring of height map to be more realistic (sand to grass to mountain). Added heightmap fade into ocean. Added get methods in heightmap and skybox to provide positioning of the ocean. |
| Jesse Schollitt | 23 hours ago | commenting for the main added. |
| Jesse Schollitt | 26 hours ago | Added complete height offset method to simulate an island land mass (made "edge pinning" redundant so will be removed. Commented out for now). Fixed "dim colours" on non-textured objects that are drawn post-skybox. |
| Jesse Schollitt | 2 days ago | added delta time |
| Aquila Halpé | 2 days ago | Merge branch 'master' of https://github.com/scholj05/AdvGameProg-Assignment2 |
| Jesse Schollitt | 2 days ago | Fixed edge faces on first row not being added. Fixed skybox rotating with camera by applying the inverse 4x4 matrix to the skybox and making it very large. |
| Aquila Halpé | 2 days ago | Merge branch 'master' of https://github.com/scholj05/AdvGameProg-Assignment2 |
| Jesse Schollitt | 2 days ago | Merged skybox into project. Currently stuck in local coords so there's no rotation. |
| Aquila Halpé | 2 days ago | Merge branch 'master' of https://github.com/scholj05/AdvGameProg-Assignment2 |
| Jesse Schollitt | 3 days ago | Started on loading .obj file for use as a vehicle for chase camera. |
| Jesse Schollitt | 3 days ago | Camera converted to quat (new class |
| Jesse Schollitt | 10 days ago | Fixed bug where running the camera.moveForward method in the first loop caused the camera position to be set to non-real numbers. |
| Jesse Schollitt | 11 days ago | Playing around with time's effect on camera movement |
| Jesse Schollitt | 11 days ago | Not using quat-based camera yet |
| Jesse Schollitt | 12 days ago | Playing around with quaternion camera instead of euler. |
| Jesse Schollitt | 2 weeks ago | Menu finished. Not particularly well done but as a first attempt at making a game menu from scratch |
| Jesse Schollitt | 2 weeks ago | Menu is now working (values from the menu are used as input parameters in the heightmap class). Bypasses number distribution at the moment to avoid invalid sigma input for distributors that don't use min/max parameters. Fixing it requires changing the min/max fields to mean/sigma fields in the menu (different input types in the arrays). Menu has its own update/render/draw system with separate window events to run the program entirely internally until it is complete. |
| Aquila Halpé | 2 weeks ago | Merge branch 'master' of https://github.com/scholj05/AdvGameProg-Assignment2 |
| Jesse Schollitt | 2 weeks ago | Added game menu (not fully working but implemented for testing purposes) that will control the vars of the heightmap class prior to its creation. Currently: Escape to close game and menu |
| Jesse Schollitt | 2 weeks ago | UI and heightmap are now in their own files with headers. Cleaned up main file of unused vars. |
| Jesse Schollitt | 3 weeks ago | Quick update for distribution |
| Aquila Halpé | 3 weeks ago | plan file + ignore file |
| Jesse Schollitt | 3 weeks ago | Slight adjustments to the camera class. The constructor now takes the initial camera position and angle as parameters |
| Jesse Schollitt | 3 weeks ago | Changed the square and smoothing methods to check for edges. If on an edge |
| Jesse Schollitt | 3 weeks ago | Overhauled land generation: Fixed the diagonal valley issue |
| Jesse Schollitt | 3 weeks ago | Changed camera operation from glRotate & glTranslate to using Matrix and glLookAt to avoid gimble lock and non-local z-axis camera roll. Camera has been moved to a separate file |
| Jesse Schollitt | 3 weeks ago | Initial commit. Wrking versions of height map |

Please be aware that the above commit history should be best used as a general guide of the development history, as some work was not committed / done in external files / research.

Timesheets / Time Spent

Aquila – estimated about 30-40 hours research and coding

Contribution : Skybox (built on John’s lighting demo) + some minor aesthetic elements to menu / heightmap colour gradients.

Jesse – estimated 30-40 hours of research and development per week (6 weeks)

Contribution :

Heightmap class (built on John’s code for diamond square).

* Random number generation.
* Smoothing algorithm.
* Edge pinning method (redundant).
* Colour buffering.
* Height-based colouring and smoothing.
* Height Offset algorithm.

Menu class.

MenuData class (holds data used in menu).

Euler camera class (redundant).

Quaternion camera class (replaced euler class to avoid jitter and gimbal lock).

Ocean class.

Overlay class.

Part of skybox class.

Object loader and object class (removed due to time constraint).

Postmortem and Team Reflection

Aquila: Working with OpenGL was certainly an interesting experience; I think that at a basic level getting started is seemingly a bit more daunting than it actually is mainly because of the weird method calls and lack of user-friendly documentation (and no hover-over method descriptions for the methods in the IDE), but beyond the basic display stuff things can still be quite confusing and complex. I didn’t contribute quite as much as I’d have liked to in this project due to a pretty severe workload from other assignments, but I still enjoyed the learning experience a lot and I’d still say that if I decided to go ahead with OpenGL and learn more I’d have fun exploring all the options of a very powerful graphics library. Getting the skybox and the heightmap to look nice and (somewhat) realistic was really satisfying though.

Jesse: Working with OpenGL and 3d has been a very rewarding and humbling experience. The amount of research needed to understand and implement OpenGL aspects of the project took far longer than I has originally anticipated, but I feel like I have developed (pun intended) as a programmer significantly because of it. The hardest part of this project has been understanding quaternions (or at least attempting to) for use with the “camera”. The most rewarding part was seeing the heightmap transform from a wireframe to a coloured and height-offset island in an ocean. From the start of the assignment hand-out to the submission date (apart from exam prep time) I have spent a good 30 – 40 hours either researching or working on the project, which has allowed me to spend time working on the algorithms, such as the height offset method and the quaternion maths.

Things to improve:

* Collision detection for the land.
* Get the object loader to work and use the SLERP method to produce a smooth chase camera effect.
* Get the normals for the heightmap so lighting can be used.
* Set the skybox to the X,Z coordinates of the camera as to not allow the camera to leave the heightmap.
* Better menu. Current menu is not ideal and has a below-standard MenuData class.
* Continuous, procedural land generation for infinite worlds.