|  |  |  |
| --- | --- | --- |
|  | **Rochester Institute of Technology**  **Golisano College of Computing and Information Sciences**  **School of Interactive Games and Media**  **2145 Golisano Hall – (585) 475-7680** |  |

**Data Structures & Algorithms for Games & Simulation II**

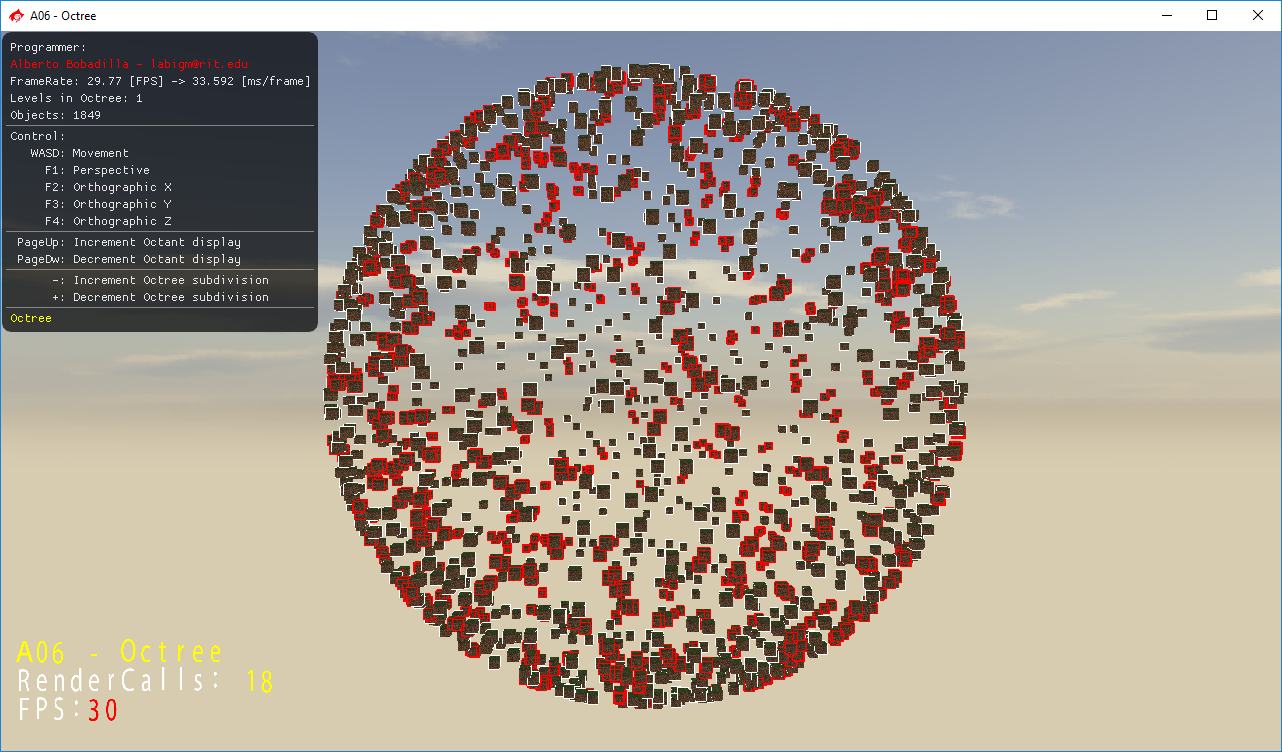
**IGME 309**

**A06 – Spatial Optimization**

The purpose of this homework assignment is to show how your spatial optimization code works. There is a short video in MyCourses, please refer to that video first.

Using the provided startup code (from the repository under A06) you should be able to subdivide the space into octants that will improve the FPS count. With no initial subdivision, your FPS should be low, if you have a really good computer at hand and without any spatial optimization you have a really good framerate please increment the object count.

The starter code will give you this out of the box:



All objects need to be checked for collision and your spatial optimization data structure needs to be displayed. With the press of a button your spatial optimization should be switched to a brute force check and backwards (you should also include the framerate in display and whether your spatial optimization data structure is enabled or not).

As usual, feel free to use your own solution or base your deliverable on the Simplex toolkit. There is an example binary under \_Binary in the repository.

All colliding objects should display it ARBB as red and all other objects should display it in white.

Your grade will be as follows:

10% Enable / Disable Octree check (Or increment and decrement the octree subdivision level).

15% Recreate data structure on the fly. (this is tied to the first point)

10% Enable / Disable Spatial Optimization visual representation.

10% Colliding objects highlight.

55% Increment FPS *visibly* by having SO check enabled.

***Submit to the dropbox labeled: A6 – Octree***

As usual the required submission asks only for the project folder, not the whole solution, it should be no larger than 200kb if you are using the starter code (and you remove this document from that folder). If you are using your own framework/engine please submit the whole solution. Push your solution to your repository with the comment “**A06 Deliverable**” then zip the project (or solution) and upload it to the dropbox, in the comments section you need to specify the address of your repository.

Example:



* Temp rigid bodies and find the max/min x/y/z == use them to create the Root octant.
* Each octant holds a center and size to measure the bounds of x/y/z:

ex) center = 0; size = 2; max = 1,1,1 / min = -1,-1,-1 etc…

* Entities hold index of which subspace they’re in, use simple bounds test to determine which bounds they are in.