Assignment 6: Octree

Course: IGME 309 – Real Time Simulations for Games II

Golisano College of Computing and Information Sciences

School of Interactive Games and Media

Rochester Institute of Technology

Due: Check in MyCourses

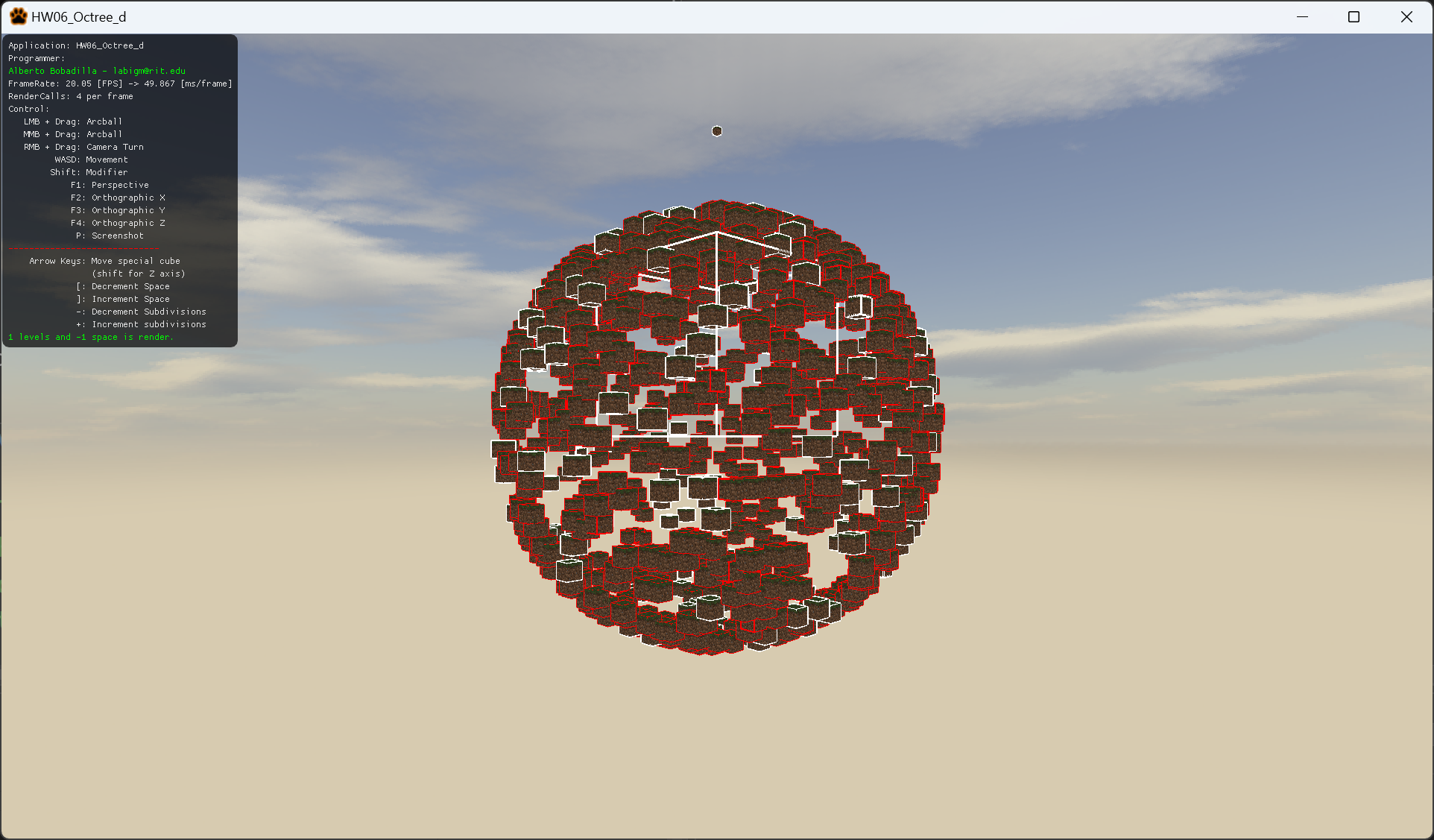
Deliverable: Unzipped Space.cpp file

## Getting Started

Video description: <https://www.youtube.com/watch?v=MQRQtbUQMJ4>

In the class repository, I’ve included a solution under the \_Binary folder.

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



## Assignment Overview

The objective of this assignment is to demonstrate the functionality of spatial optimization using an octree. You will be provided with an example executable located under the \_Binary folder. Using the provided starter code (from the repository under A06), you will need to subdivide the space into octants to improve the frame rate. Initially, the frame rate will be low, but if your system's performance is high without spatial optimization, increase the object count for testing purposes.

The starter code will provide you with the necessary structure and setup for this task.

## Grading Breakdown

|  |  |
| --- | --- |
| Criteria | Percentage |
| Enable/Disable Octree Check (or adjust octree subdivision level) | 30% |
| Recreate Data Structure on the Fly | 15% |
| Increment FPS by Enabling Spatial Optimization | 55% |

## Additional Details

Note: Your code will be graded in Debug mode, and performance may naturally increase when compiled in Release mode. The primary goal is to visibly increase FPS with spatial optimization enabled. While performance gains will vary across systems, if the FPS improvement is less than a few frames, consider optimizing your tree search.

If using the provided starter code, ensure that you fully implement the following methods:  
 - Node::Node(uint a\_uMaxLevel, uint a\_uIdealEntityCount)  
 - Node::Node(vector3 a\_v3Center, vector3 a\_v3Size)  
 - bool Node::IsColliding(uint a\_uRBIndex)  
 - bool Node::ContainsAtLeast(uint a\_nEntities)  
 - void Node::Subdivide(void)  
 - void Node::AssignIDstoEntities(void)  
Some methods may already be partially implemented but are heavily commented. You can base your solution on the grid code.

Note: There is no extra credit available for this assignment.

## Submission Instructions

Submit your Space.cpp file (not the entire solution) through the dropbox labeled 'A6 – Octree'. If you are using your own framework/engine, please submit the complete solution. Push your solution to your repository with the comment 'A06 Deliverable' and then upload the mentioned file it to the dropbox. In the comments section, specify the repository address.