|  |  |  |
| --- | --- | --- |
|  | **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**

**Recursive grid**

This exercise follows lecture D13

1. Your code will start like this:

A screenshot of a computer

Description automatically generated

1. Under \_Binary look for the example solution. It will look like this:

A screenshot of a computer

Description automatically generated

1. Out of the box the subdivision on the grid will be already functional, and the connection between the entities and the spaces is as well. What you must do in this convert the functionality to be a recursive method. Both grids have the same level of subdivisions, but the algorithm for this new one will divide the space in 8 and then it will divide the new ones in 8 and lastly once again each one of them in 8. The old way of doing it will take the space and divide it in equal parts (in this case 3 parts instead of 3 levels)
2. This exercise only requires the Node.h and Node.cpp files for submission. Please zip them up and push to your repo as well.

As a heads up to what is following, this is what an Octree would look like, this is your next homework assignment.

A red circle in a cage

Description automatically generated

An octree will only divide the space when its necessary, based on two conditions, how many obects I ideally want per space/node/octant and what is the maximum level I want to reach to