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

**Final – A Star**

The purpose of this assignment is to show how in a visual manner the most inexpensive path between two points.

There is no starting code and there is no need to work on the framework we have being working on though the semester (although its way easier on you if you use it) but the following conditions should be met:

* Code should be in a Visual Studio 2019 project
  + In case you are not using the class framework you need to make sure I’m able to compile it.
* It should be written in C++ (no wrappers around C++, you cannot code a C# code and use a wrapper to translate it to C++)
* It should show the shortest path highlighted from all possible paths in a clear way, along with the starting point and the end point.
* It should use OpenGL 3.x onwards.
* All code should be commented.
* The path follower should be visible at all times and should be able to traverse the shortest path via linear interpolation or some sort of steering behavior.
* Not all pathways should be weighted the same, there should be clear obstacles.
* Incorporate other things discussed in class; Collision Detection and Resolution are a must but other things like Octree for spatial optimization are highly encouraged.

As there is no starter code nor specific desired goal, this project is pretty open, along with this document you should find a binary executable example of something you could potentially do. I do not expect you to replicate this example as is, all your ingenuity will be beneficial for your grade. The only restriction here is to use AStar and have collision detection and resolution working. Some students have implemented physics alongside with the solution in such a way that an NPC pushes boxes that are in their way. Optimizing the space is highly encouraged, but again, what you can do is pretty open. You need to include a READ ME file to outline what you did and what the capabilities of your program are.

***Submit to the dropbox labeled: Final Practical***

***If you are using other models, textures or shaders make sure you are giving me anything I need in order to compile your code.***

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 “**A\* 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.

Please make your submission in the following format:

*lastF\_As.zip*

What I mean by this is take the first four letters of your last name, append the first character of your first name, and then append the assignment code (in this case, As.) For example, John Smith would submit “smitJ\_As.zip”. This helps our graders not have to download twenty submissions all called “Solution.zip”, which makes them happy.

Example:

