|  |  |  |
| --- | --- | --- |
|  | 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 309-02, 2016-17 Spring**

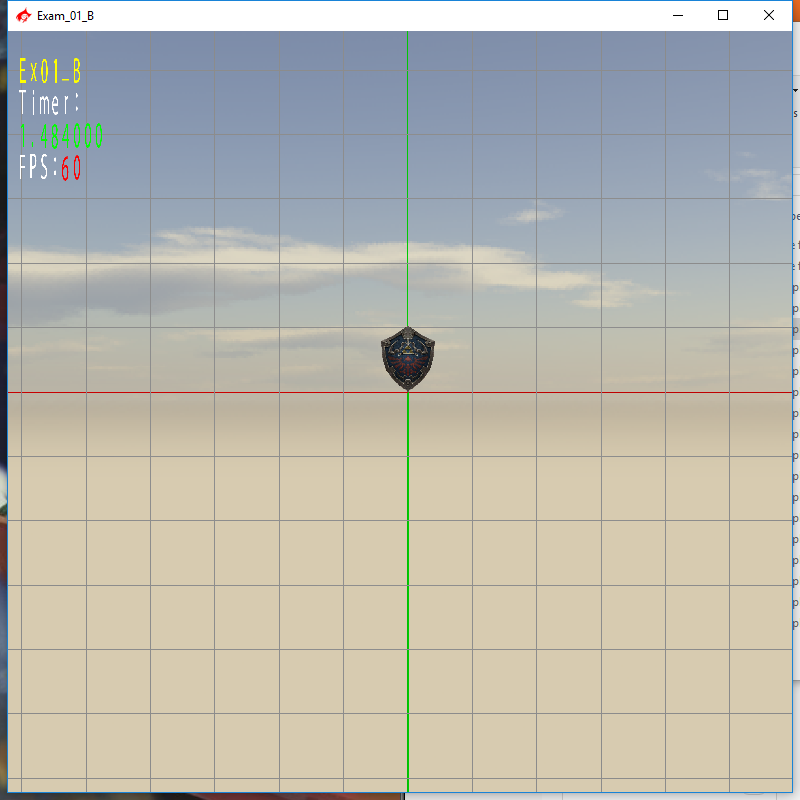
**Midterm – Practical (version B)**

Instructions:

A) Read this whole document before you start.

B) Using the provided code, implement a program that replicates the behavior described below and in the provided sample (MIDTERM DEMO.exe under \_Binary)

You are handed out code that will draw this out of the box:



Your goal is to make the model revolve around the origin while having the tip of it always pointing down and in a time-based manner. (there is an example demo under the Binary folder)

You are provided starting code and you will only need to focus on the AppClass.cpp, no other files need any changes, assume that nothing in the provided code need any changes (other than setting up the values for the variables).

**Your animation need to last 2.5 seconds. In that time, it needs to perform 1 complete rotation around the center and needs to be separated from it by 3 units, after that the animation restarts. There is a clock in the project you may use it, modify it or completely ignore it.**

Requirements:

* All your orientations should be performed using quaternions, although it is acceptable to perform rotations using matrices points will be taken off if you do, you need to translate the quaternions to a matrix4 in order to send it to the model, that is the only acceptable use for a rotation matrix.
* Your code MUST compile AND execute. I will not take points out of the program if it doesn’t compile AND/OR run, I will simply not grade it. If your program does not run it will receive a 0/100. If you are having trouble with something in the code comment out the lines, say what you wanted to do and what you suspect the issue is. That will result in partial credit, which is better than not having a grade.
* Memory Leaks are acceptable, points will be taken off, but the code will be reviewed.
* You only need to modify AppClass.cpp; there is no need to modify anything else.
* You get rid of the “trash files” (intermediary files).
* Zip your project and upload it to the dropbox in my courses. The project should be less than 1 megabyte in size.

Grading:

**50** points shield rotates around the origin while being 3 units away from it lasting 2.5 seconds.

**50** points shield tip always point down (you can only get these points if the previous requirement works).

**-50** points rotations are performed not using quaternions.

(-???) Cheating:

Talking with anyone in person or online. You are only allowed to use MyCourses to download this file or upload your solution. Anything else is considered cheating.

(-100) Code not compiling or executing.

(-10 to -20) Memory leaks (You are not reserving new memory for this test so this shouldn’t be an issue)

(-10) For each uncommented method; I need to know what you are doing or trying to do.

(-10) You forgot to delete the \_Delete folder

(-10) You forgot to delete the .sdf file

(-50) Your application is not time based

(-50) Your model only translates or rotates, but does not both at the same time.

Extra points:

There are no other extra points specified for this exam, except for one thing: surprise me (in a good way). As I don’t know how surprised I will be I can’t tell you how many extra points I will give you, just do your best, and as usual, in order to get the extra points you need to have a satisfactory degree in the required part. If you are going for the extra credit make a new project within the same solution and name it something else.