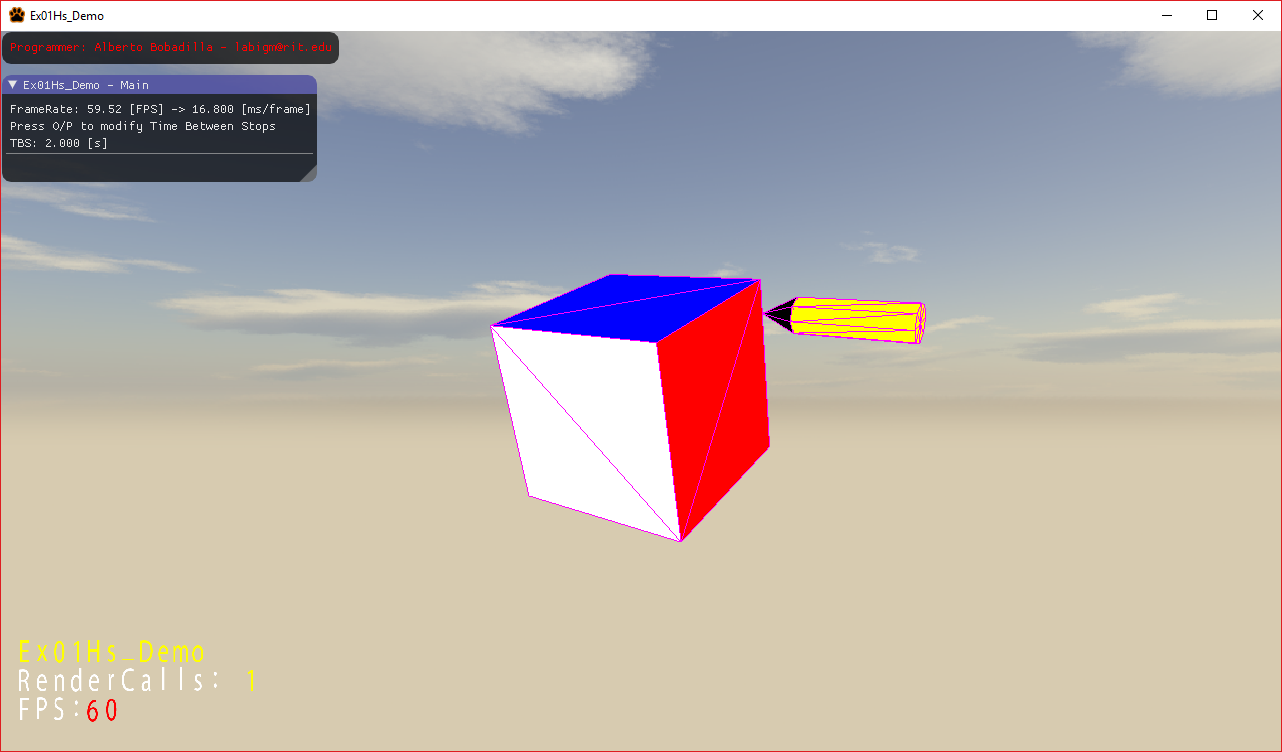
|  |  |  |
| --- | --- | --- |
|  | **Rochester Institute of Technology**  **Golisano College of Computing and Information Sciences**  **School of Interactive Games and Media**  **2145 Golisano Hall – (585) 475-7680** |  |

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

**IGME 309**

**Midterm – Practical (version h)**

I’ve included a solution under the \_Binary folder. Please take a look at that before continuing reading this document.



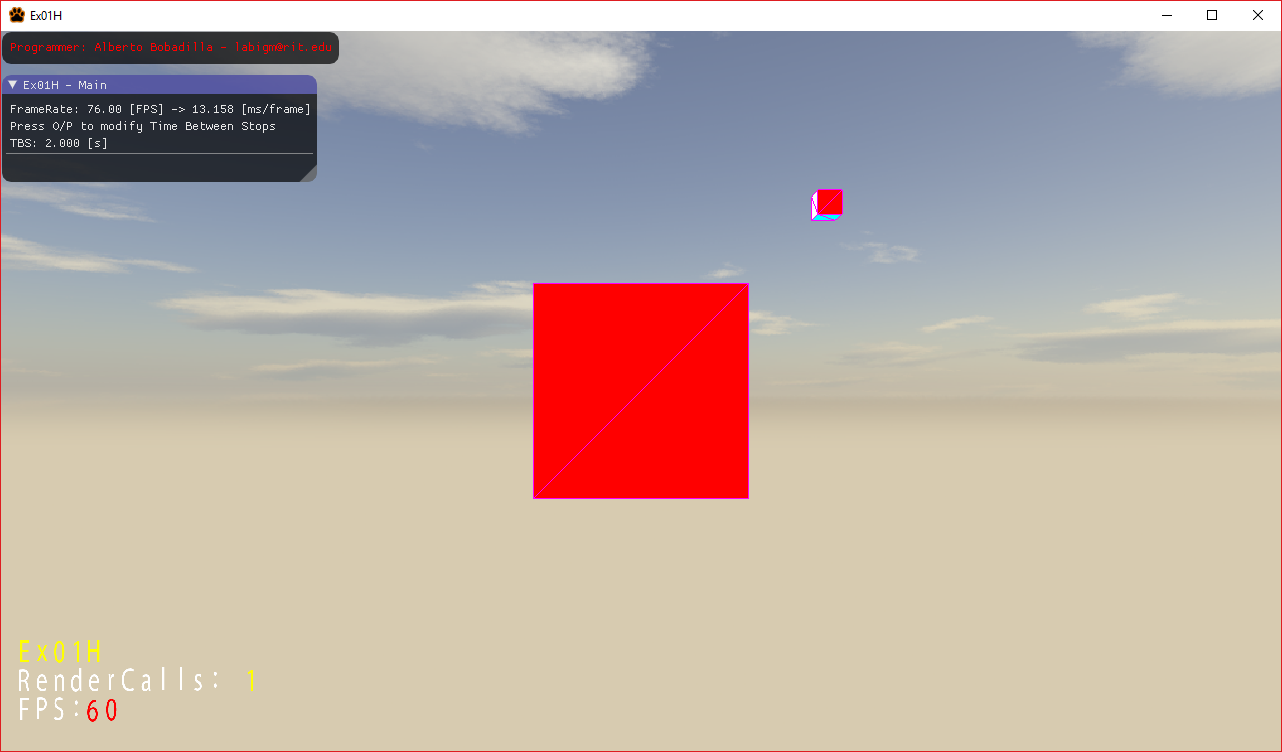
You are working on a drawing app that will teach children how to draw basic triangles and make shapes with them for a company, unfortunately the project director only gave you his artistic idea for what it should look like:

“I want something that looks like a pencil is drawing a cube made out of triangles”; In his artistic mind he wants the pencil to be drawing the contour of the triangles and therefore the tip of the pencil should always be touching a line of the cube and nothing else but it does not matter if you backtrack or go about the same line multiple times as long as the pencil keeps moving all the time and restarts at the beginning of the shape once it’s done (basically looping forever).

What you are expected to do is replicate the demo to the best of your abilities.

Fortunately for you, the programming team have a primitive drawing framework that is almost complete, it can generate cubes and nothing else but can in fact be expanded to generate other shapes.

You will start with a cube at the center of the world, if you middle click you can rotate it around its center. There will be another cube that will be floating away from it. This smaller cube is in fact a cone but the framework is incomplete so its up to you to complete the shapes required.



From the starter code there are some things you need to do:

1. Change the name of the programmer (you)
2. Modify any shapes required for this task, it is not necessary to finish every method of this mesh class so focus on what you need to make this work.
3. There are only two mesh objects initialized by default, you are also expected to initialize any other mesh that you think you will need.
4. There is an example of a counter in the code similar to what we did in class, you may use it or construct your own as needed.
5. There are some sections of code that are marked under pragma regions that do not need any changes, please leave them as so.
6. There is an example of a lerp based on the timer this is just an example it needs modifications, in fact everything outside of the pragma regions may require modifications.
7. As you can see in the GUI pression O and P in the keyboard should increase the speed of the pencil (reduce the time between stops really) you will need to add that behavior, this should be performed in the AppControls.cpp
8. You are allowed to add more variables and methods as needed.

Your grade will happen as follows (up to 110%):

20% (up to) if you generate the required shapes

15% (up to) if you draw the shapes as required (it should look like a pencil)

20% (up to) if you locate the cone (tip of the pencil) in the right position and make it go around the lines of the shape at all times without stops, even if the pencil is facing any other way or the tip is not pointing where it should

15% (up to) if the tip of the cone is always facing a line (even if its not oriented as it should)

15% (up to) if the pencil is always perpendicular to the face its currently drawing

5% (up to) if you can decrease the speed of the pencil

5% (up to) if you can increase the speed of the pencil (should never take less than .5s between stops)

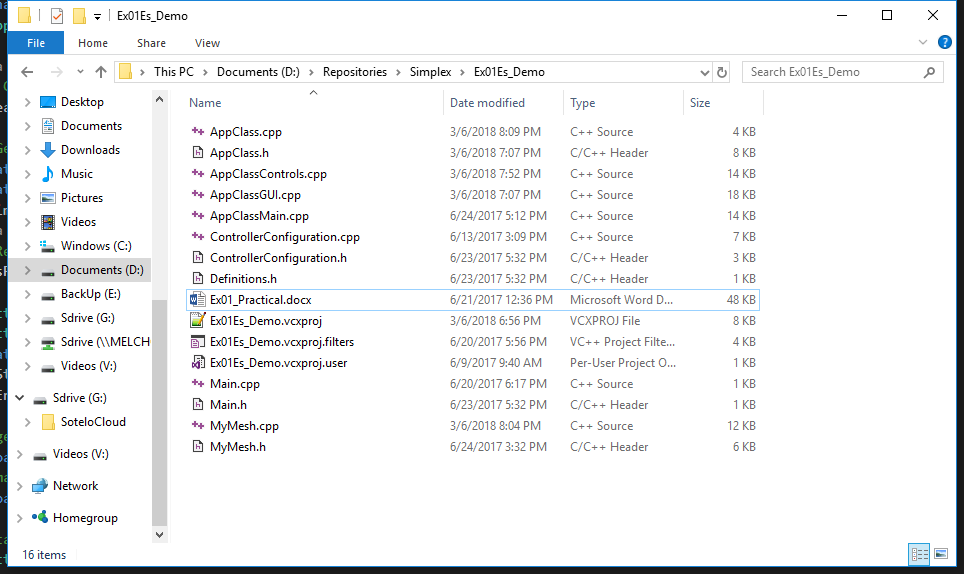
5% (up to) if the whole system of pencil and cube can be rotated with the middle click like in the example.

5% (extra, up to) if you draw an eraser at the other end of the pencil and it moves with the pencil as expected; the extra credit is only applied if you complete the other parts to a satisfactory degree.

5% (extra, up to) if you add a sound effect as the pencil moves; the extra credit is only applied if you complete the other parts to a satisfactory degree.

***Submit to the dropbox labeled Ex01 –Practical***

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 class framework, if you are using extra media files (i.e. for music) please include them and let me know in a README file the right location relative to the executable. The content of your .zip file in your submission should look like this (names might be different):



After you submit your file it is your responsibility to download your submission and make it is what you worked on and not the starter code, **it has happened to other student before, do not let it happen to you.**