|  |  |  |
| --- | --- | --- |
| DATE AND TIME | TIME SPENT | DESCRIPTION |
| 11/11/2020 | 3h | I have started week 6. I have updated the model class adding a new vector containing the transformed vertices and I have added three methods. Then I have created a camera class, containing a camera matrix and the accessors to retrieve it. Then I created two methods in the rasteriser class to generate the view and perspective matrix and I have added the code in the .cpp file. In the view matrix I incurred in an error when I tried to divide width by 2(width/2) saying that is not possible to divide a float by an int. To resolve this I have added a “(float)” in front of width to output a float as result. |
| 12/11//2020 | 3h | I have added the drawWireframe method to the rasterizer class to draw the triangles that compose the figure. After that I have added the code in the Update and in the Render method in the rasterizer class calling the methods that I have created in the previous tasks. At this point, after I have completed all the steps, I have tried to compile the program and I had some errors to resolve. The compiler was reporting that the Vertices inserted in the transformed\_vertices vector where greater than the actual vector size, at this resulted into the crash of the program. I found out that this error was appearing because in the loop responsible in creating the transormed\_vertices Vertices, I had set that the loop would have continued until “i” was smaller or equal to the \_vertices vector size. I have corrected it by setting “I” smaller than the original vertices size. After this adjustment the program was showing a rotating cube. After 20/25 seconds the cube suddenly stopped. I tried to run the program again for a second and try test and the same error occurred again and again. This was happening because I wasn’t deleting the “pen” used to draw the triangle so I added a “DeleteObject(pen)” that fixed the problem. |
| 25/11/2020 | 4h | I have tried to implement the code to add the backface culling to hide the polygons that I should not be able to see making the object more solid. I have created a new Vector3D with methods to calculate the Dot and the Cross product and a methods to subtract two vertex returning a vector object. Then, I have added the calculatebackfaces() function that mark the polygons for culling. After that, I have updated the Render and DrawWireFrame methods in the rasterizer and tried to run the program but I didn’t get the result that I expected. Indeed, I was still able to see the polygons that should be hidden. Moreover, some of the vectors disappeared when they was facing toward the camera while they should clearly be visible. I found that the error was in the DrawWireFrame method I had set that the polygons are drawn when they are marked for culling. Therefore I had the opposite result of what I needed. So in Getculling() ==true, I changed the “true” with false resolving the problem. I changed the md2 model choosing the marvin.md2 to test if the program was running in the right wat with other models and I have notices that only some of the polygons that should be hidden disappear during the rotation because they needed a depth sorting. |
| 26/11/2020 | 5h | I have implemented the sort method in the model class and called it in the rasterizer, but I came to user the sort() function, I had an error telling me that I have used parameters than allowed in the sort function. The error said that I used three parameters when I was supposed to use two of them. The error was given for the third parameter where I used a called the binary predicate. To resolve the error I declared the binary predicate directly into the sort() function. I tried to run the program and this time it didn’t have any error. However, after running the program there weren’t any changes compared to when the depth sorting was not implemented. |
| 27/11/2020 | 2h | I realized that to check if the depth sorting was working I have to debug the program and I found that all the z value are sorted correctly. I have inserted a counter to translate, a scalate and rotate the figure after a certain amount of time. Between the translation and the scalation there was a problem when the figure was scalating from a different starting point, even if the value for the y and x were correct. The error was that the x, y and z values were initialized to 0 instead of 1. |
| 7/12/2020 | 5h | I have implemented a method to a solid figure in the rasterizer class with the result of getting figure which is completely filled, but plain. |
| 9/12/2020 | 6h | I have added the directional lighting to the model. When I tried to test it I’ve noticed that during the rotation some polygon where changing their colour at the minimum movement and this wasn.t supposed to happen. I found that the problem was that I had only clamped the r g b values to be smaller or equal than 255 and not to be greater or equal to 0. I have added the missing part and the direction lighting was working in the right manner. I have then added the ambient light, and tired to see if it was working changing the values of the directional lighting and the ambient lightning, getting a successful result. |
| 10/12/2020 | 4h | I have added the point light to my figure. When I tested I noticed that instead of making the figure brighter it was making it darker, even if the colour of the point light was set to (255 255 255). I found that I subtracted the position of the first vertex in the polygon from the position of the light source when I was supposed to do the opposite. I made the adjustment and now it was working fine.  Note. I set the compensating value to 1 instead of 100 because with 100 it was turning the figure completely white. ( In this case the compensating value can be removed, but I left it in case the value and the figure need to be changed) |
| 11/12/2020 | 3h | I have inserted the on-screen text, improved and simplified the code where possible, created and tested the release version, registered the demo video and set up the folder for the submission. |