Practical Exam

Object Pooling (odd)

C. To use object pooling I first implemented a pool class in order to receive from the inspector all the information about the object that's going to be using the pool including the name of the pool, the tag, and the amount needed. Then from the object pooler class the object was added into a dictionary and added into the scene through a queue. Before using the object pool, there's only 3 enemies on the scene but the pooling we can add many more. From figure 1, by using object pooling it is easier to add in more enemies without any lags. From figure 2, there is a significant increase in lag and GC allocated in frame from adding objects manually.

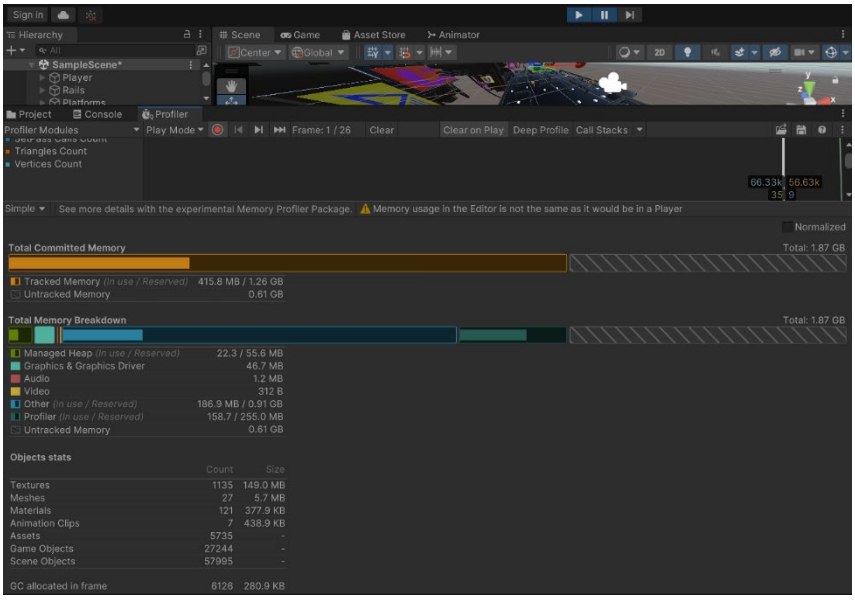
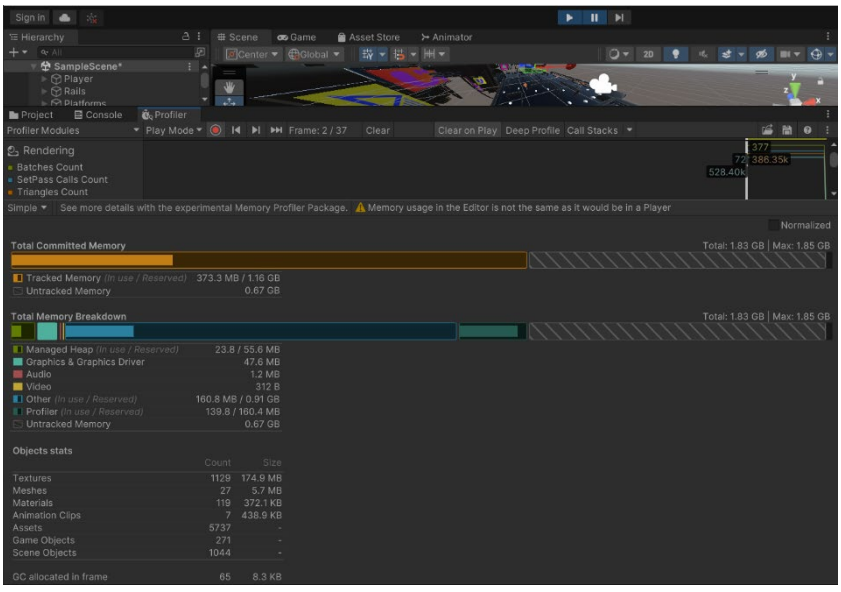


Figure 1

Figure 2



Question 7.

I used a game and editor manager to keep the scripts and components of my game together. The game manager currently holds the movement script for the enemies, player input controller, and object pooler. This manager allows the scripts to be implemented without having to be attached to an entity. I also added an editor manager so the developer can see what the whole level looks like. This manager is what we learned from our labs, except it only shows the player how the whole level looks like. In order to implement this script, an extra camera was added from a top-down position and then was enabled and disabled through player input actions.

Without editor camera enabled:

With editor camera enabled: