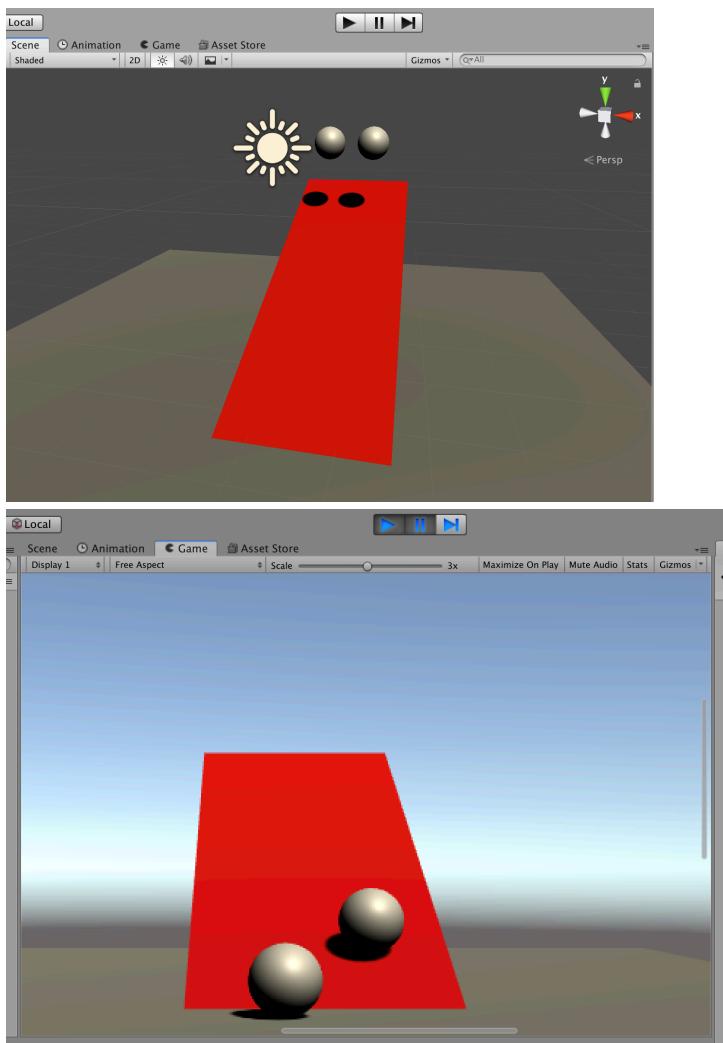
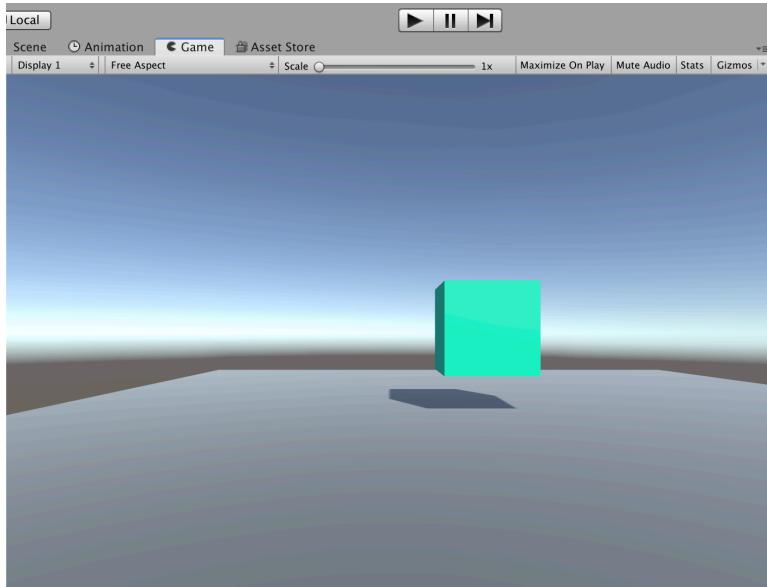


This Moving Parts project was very fun to do and test over and over until I was able to get everything right. The first phase was fairly easy, I just had to find the create physical materials tab in Unity. The second phase was a bit harder, trying to look up the AddExplosionForce on the cube, and getting it right. The last phase was the trickiest, I was able to get the ball rolling and whenever it struck a wall it would choose a random direction and then head there. The only thing that was wrong was it was that whenever it ran into a corner it would stop completely, so it took some time to fix that.

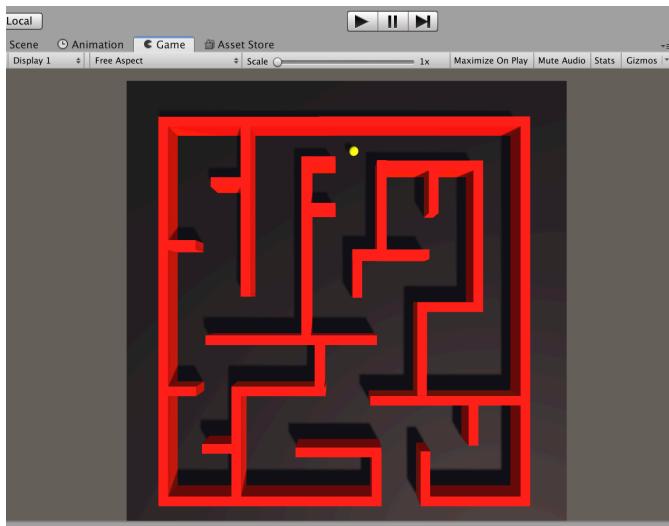
Phase 1 was the easiest out of all three phases to do and complete. I didn't add any color onto the spheres that are rolling balls since I didn't feel that it was necessary. The ball that has bounce and some friction was faster compared to the ball that has no bounce and no friction. Below is the scene in Unity and how they look when the balls reach the bottom of the ramp.

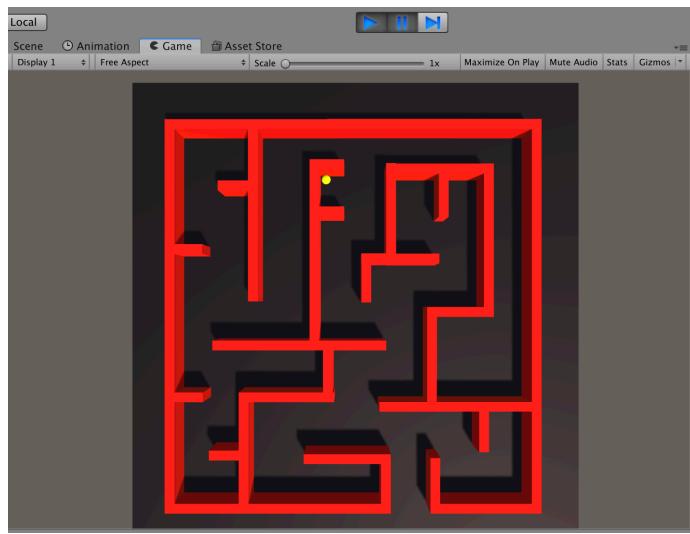


Phase 2 was trickier to figure out compared to Phase 1 simply because it took me a long time to realize that there was an AddExplosionForce for the Rigid Body of the cube. I'm not sure if I did the velocity to 10 on the cube in the right way since the maximum Y is the same for all three masses. Below is the cube, before the scene is played and what it looks like.



Phase 3 was the hardest to do since it required using OnCollisionEnter and figuring out how it worked. I was able to figure it out, but there was a problem where the ball would lose its velocity and stay still or where the ball would hit a corner and just stay there. To solve this problem in the update function I set it to look for whenever the balls x and z velocity was equal to zero and if it did it would choose a random direction and keep going. Below is the scene of the maze and the ball before it is played and during.





I was able to effectively learn how a rigid body on a 3D object works and how to properly script it.

A screenshot of a Mac desktop environment. On the left, a window titled "Moving Parts.pdf (page 2 of 2)" is open, displaying a table of data. On the right, a file browser sidebar shows various PDF files and folder icons.

Moving Parts.pdf (page 2 of 2)

	.1	1 (default)	3
Maximum Y-coordinate for normal force	510.9673	7.276898	2.813003
Maximum Y-coordinate for explosive force	510.6873	6.996897	2.276076
Maximum Y-coordinate for velocity	6.996897	6.996897	6.996897

Phase 3 –the silly bad guy

- Create a maze consisting of cubes and a plane
 - All of them will have a rigid body
 - All of them will be kinematic
- Create a sphere with a rigid-body.
 - Lock the constraints to prevent any rolling
- Make sure the sphere has a physics material with no bounce and no friction.
- Create a script that will set the velocity of the sphere to a random direction on the X/Z plane.
- Every time the sphere collides with the wall, randomly chose a new direction for it to go.
 - Look up “OnCollisionEnter”
 - You may need to use “OnCollisionStay” as well.
- The sphere should then move (howbeit erratically) around the maze.

File Browser Sidebar:

- PDF
- My Schedule.pdf
- C++ Programs
- Operating Systems Slides
- Computer Animation Slides
- Overlord
- Computer Network Slides
- Programming Languages Slides
- Games
- Resume
- HL2 - Sergio Torres
- Hope

Programming Languages Slides

Degree Planner.pdf

Resume

HL2 - Sergio Torres

Hope