Neil McCuaig Assignment 5:

Pong, Lunar Lander, or Asteroids?

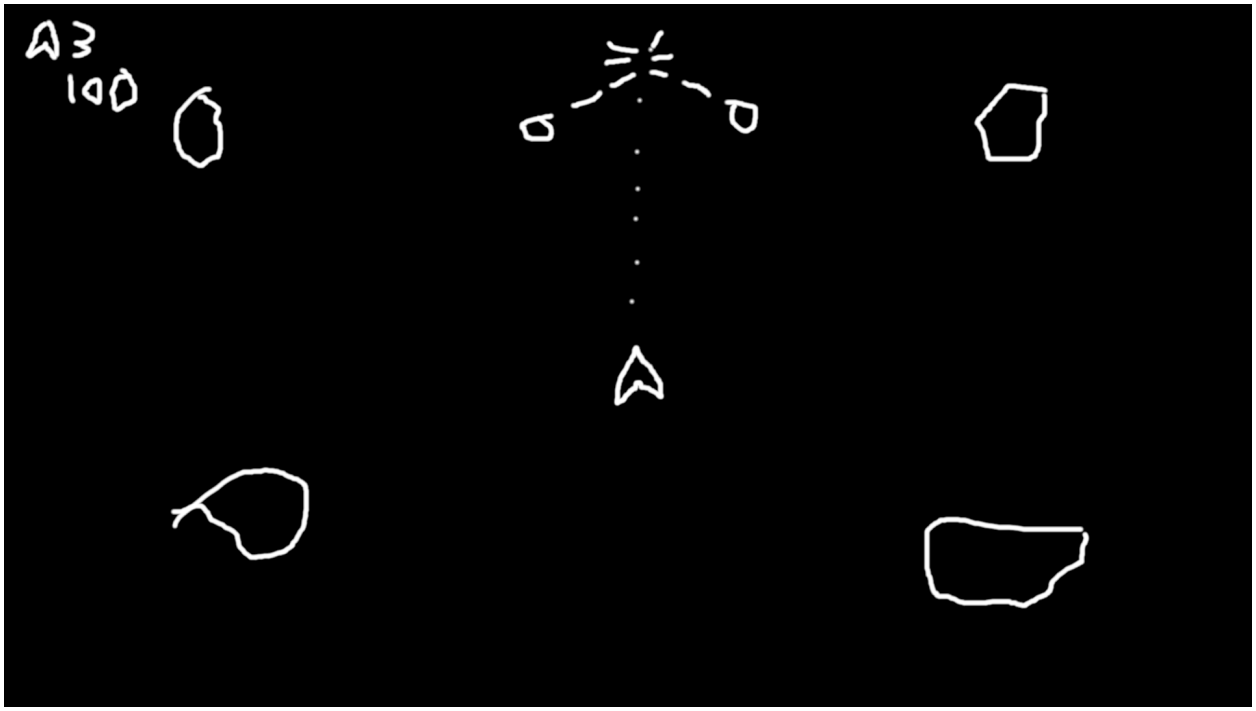
Pong. Nice and simple and you want to try to make an AI in unity

Lunar Lander seems entirely around physics.

Asteroids. More complex, and you learn how to randomly generate things.

Asteroids seems easier to accommodate the physics portion of the assignment at first glance then pong. Lunar Lander has no decent tutorials in Unity. Asteroids it is.

It will be a simple classic style game of asteroids, with wireframe models and the ship represented by a triangle. The ship will fire bullets, die on impact with asteroids, and will not be able to fly off screen. The asteroids will spawn at a random size using one of four models. There will be a score and lives, which will be visible in the top right. The asteroids will not collide to reduce difficulty.



For some reason OneDrive has taken a dislike to Github Desktop. The entirety of onedrive had to be unsynced to stop it from freezing the file explorer. A terribly annoying waste of two hours.

The issue with the 20,000 files is solvable by sticking the Gitignore file into the unity project. Neat.

Seeing as the background will be black, the main camera background color could be made black too. No background work to do.

Do I want a custom aspect ratio? Not for now, maybe test it later.

Do I want to let the player fly off the screen? Probably not, might be fun to implement if I have the time.

Found some sprites for the player and rocks in one of the tutorials. Should make life easier.

Scene should not need anything else beyond the offscreen walls and the player, since the bullets spawn at a button press and the asteroids spawn in randomly.

Visual Studio 2022 is complaining that the particular script I made for the player movement is unsupported.
“

> **Assembly-CSharp.csproj:** The application which this project type is based on was not found. Please try this link for further information: http://go.microsoft.com/fwlink/?LinkID=299083&projecttype=E097FAD1-6243-4DAD-9C02-E9B9EFC3FFC1

“

Oh. It just did not have the unity component installed anymore because I had to strip out onedrive. Oh well.

After installing unity Component, it still says incompatible, but the codespace is showing up. Let’s see if it is truly incompatible.

Scrapped the first script, lets try making another.

It still says it is incompatible.

Turn it off and on again.

Still incompatible after reboot. Damn it.

Wait a sec the tutorial has visual studio code instead. Lets try that instead.

Updating visual studio.

You updated visual studio and went edit -> preferences -> external tools -> regenerate project files.

The button seemingly did nothing and vs is still broken.

It was right click in visual studio on the busted project and click “reload with dependencies”. You have got to be joking.

Seems there was some sort of an issue with the bullets. They don’t go anywhere when I shoot em and complain about there being no rigidbody.

The tab to input suggested skipped the 2D on the end of everything, so Rigidbody2D was just Rigidbody.

For some reason pressing your touchpad makes it fire really fast and the spacebar has a massive firing delay.

Fixed the spacebar, it was tied to the fixedupdate rather then the update. The touchpad is still wonky.

Got some asteroids from the same spot as the player sprite.

You can recycle some code from the bullet prefab into the asteroid prefab.

You can make an empty object to spawn things around it. Maybe even a few of them if you want to really ramp the difficulty.

Could make the speed randomize, but since the mass will change with size that will make things slower won’t it?

Once again, the bloody autofill did Rigidbody instead of Rigidbody2D. Seems you need to make a habit of checking on that.

Make sure to put in less speed compared to the tutorial. His were zipping by.

The asteroids are colliding with the boundary boxes.

The “On click” fire jank was because you needed to add Down to the end of it. So GetMouseButtonDown instead of GetMouseButton. Also, it caused an added bit of jank when you fired those projectiles at an asteroid, spawning like 100 same size asteroids. Fixed now, but interesting.

Should probably decrease the speed of the spawning rocks.

Avoid making that same respawn invulnerability bug that the tutorial did. Remove the this.player. From this.player.Invoke(nameof(TurnOnCollisions)

Change the color of the asteroid's death explosion.

The explosion effect has to be tied to the game manager in the Unity UI.

You can use the old defunct text feature to make it easier.