Design Doc – Timber

The game I modified for my first CA as part of 2D Games development module was Timber. This is a game in which essentially you are a lumberjack, and you have to cut away as many logs as you can before either the timer runs out or you get squished by a log.

The modifications I made to the game include:

**A Super Axe pickup -** This is a pickup that doubles the player’s score. To implement this pickup first I created an array of 6. This means that there could be up to 6 of the pick ups displayed on screen at once. I named this array NUM\_AXES. Then I created the sprite object and set it so the NUM\_AXES was the value so it could display the 6 pickups at once. Then I added the array to a new variable called superAxePositions which was used to set the side it would display on. Then I edited the axe sprite that was already there to have a red blade on it. I set this new image to the texture for the super axe. Then I made the collision code for the pick up.

A computer screen with text and images

Description automatically generated

Here I used the .getGlobalBounds function as the getPosition function wouldn’t work for me. Essentially it just gets the dimensions of the selected objected then if it intersects or ends up sharing the same position on the x and y, it sets my Boolean value pickedUp1 to true which later on allows the score to go up by 2 instead of by 1. I also changed the texture of the players axe to the new axe here and I have it set so if you pick up the pickup, it’ll get rid of all the other pick ups on screen at the time.

I then made an if statement for the score, so if pickedUp is true score +=2, else score++.

I also made a for loop to position the pick ups on screen. The code is similar to that used for the branches as they function the same in terms of displaying.

In the function for updateAxe what I changed was I set it so if the pickUps position was the same as a branch, it would switch sides. This way you never have both branch and pick up overlapping each other.

**Next I added an if statement to change the background.**

A black background with white text

Description automatically generated

So, if the score goes above or equal to 50 the background for the game changes. For this all I had to do was make a new texture object for the replacement background and call it here in the if statement to set the old texture to the new one.

**I also changed the sound effects.** To do this I used BFXR sound generator to make new sounds. I changed the sound for everything here, the chop, death, out of time and I also added background music. (which was royalty free music found online). I set this to play in a loop.

To conclude, I feel I learned a lot from this CA. I find its quite difficult to edit code that’s already pre made. Its almost easier to start from scratch as I found myself using ctrl + f a lot to find certain things. I also had to go through each line of code individually to fully understand how the game itself worked. How to use the functions already there to improve the game and add to it.