

CI123 – Introduction to Games Development

Starship Fontana Documentation

For my solution, I decided to go with a simple game with the objective of collecting the coin without colliding with the aliens. Seeing as I am currently learning about programming and software development it took me a fair amount of time to get my head around the code given to me and understanding what was going on. Once I managed to achieve that I was able to move on and add additional features which are briefly explained below.

Features added into the game:

Score system

The score system was fairly easy to implement. I created 2 functions in SFApp which would add a value to the score and get the score to print it out at the end of the game. The testing for the score system went fine, there were no issues with it and the code written was very straightforward.

Player moves up and down in addition to left and right

This was also fairly easy to implement seeing as the code was essentially there previously with the player being able to move left and right. I just had to adjust the values in the axis in order to make it work. I also had to add the event in SFEvent. I had some issues with the testing seeing as the player would not stop at the border of the game world when moving up or down and I solved this relatively easily by noticing that I had missed out an if statement checking if the y axis is greater than the game height or less than 0. If either of these statements were true it would resent the bounding box and therefore not allow the player to move past that point. I had an issue with the projectiles however seeing as they would also stop once they hit the top of the in game border. This was a bit harder to solve but after numerous solutions I added an else if statement to check if the asset was a projectile and if it was then it would call the SetNotAlive() function which would remove the projectile.

Added walls with functioning collision from all directions meaning that the player or the projectiles cannot go through them

This was by far the most difficult and most time consuming task for me. Once I figured out how the collision detection worked I attempted to make the player go down when it hit the wall, but that was not a good solution because it only stopped the player moving through it when going up. I tried various solutions but none of them seemed to work as I intended. I then moved on to the SFApp file to see if I could change something there and I figured out that I could add the collision detection in the SFEvent queue. Essentially if the player was moving towards a direction, let's say east (right), then when colliding with the wall the player would go west (left). The player would respectively go the opposite direction depending on what direction it is going if it collides with a wall. This was a feasible solution to my problem and I decided to leave it at that. The testing went perfectly fine after my final solution and I had no issues with the wall collision.

Game stops running and prints out a message with the score on the terminal if the player collides with an alien or if the player collides with the coin

This also took a fair amount of time to figure out. To figure this out I tried to find the code which made the game run. Eventually I found it in the SFApp file. I also noticed that the game was running while the Boolean "is_running" was true. I tried making it false when the player collided with an alien to test it out and the game stopped running. After figuring that out I wanted the game to print

Harry Antoniadis
Student Number: 15806678

out an end game message when the player either collides with an alien or collides with the coin. For this I chose to go with the simple solution of “c-out”. I printed out different messages for when the player hit the alien or the coin and printed out the player score as well. The testing for this also went fine and I had no issues.

Reflection

Looking at my final product I believe it is a very bland simple product. But I did learn a lot about C++ and programming in general and I gained a better understanding of how larger projects with multiple classes are linked together and how to tackle the problems. Overall I gained a lot of knowledge and I hope that I gained at the very least a very basic understanding of how to develop software. Although my product was not great at least it was a simple and functioning game