Observations

Working on this game has been an amazing experience to really test out my programing skills alongside my problem-solving and mathematical ones too. My game “Bullet Blitz” is a fast-paced top-view shooter game, where the user must collect tokens around the beach before the crabs take them. The game includes powerups and upgrades to give the user an advantage and increases in difficulty every two tokens the user collects. I have faced many problems while developing this game that were relatively easy to fix and others that took way to long. The first problem I had was collision between the enemy, player and tokens. The problem I encountered was the detection did not recognize when the objects collided but with a little research and some YouTube videos, I was able to find the “dist()” function which solved my problem right away and helped with all other collisions with nay objects. I am glad that I came across this function as it allowed me to further develop and expand upon my computing skills, which helped me fix another problem regarding the enemies’ path towards the player. My initial approach was to equal the enemy x and y to the players x and y while continuously subtracting an integer which was dependent on what stage the enemies speed was, but that did not work whatsoever so I completely reworked my approach. I started with the obvious of calculating the difference in the x position between the enemies and player, then set that to a variable and did the same for the y position. From there I used dist() to calculate the distance between the player and the enemy object and updated the x and y coordinates of the enemy object by dividing the first calculated x and y variable by the dist() calculation and multiplying it by the speed. The speed is used to control how fast the enemy moves towards the player while the distance was used to normalize the movement so that the enemy moves at the same speed regardless of how far away from the player it is. Another major problem that I came across was resetting the enemies back to their previous speed after the player collects the freeze powerup and the timer runs out. My code was supposed to include a variable that tracks the enemy speed right before the enemy collects the powerup and sets the enemy speed to that variable when the timer reaches zero however, the enemies would just reset to the original speed, the one they spawn in. So as I was trying to fix this issue, I came up with an idea that instead of setting the enemies to the speed they were at before they froze, they would continue to increase their speed every time the player collected a token which creates brings “risk for a reward” to life. A part of my program that does not work is only a powerup can display if none are displayed in the moment. I had it working but then it stopped working for some reason but currently, I have no idea how to fix it or what knowledge I would need to make this work. There are some extra things I wish I could’ve included for fun if I had extra time such as instead of the enemy just passing by or over a powerup, it goes over it and if x amount of time passes it consumes it. Another feature would be to include random ammo drops at random times during the match and more powerups such as invincibility and OP mode where the user max’s out all upgrades for x amount of time. Other than that, I believe that’s all. I would like to express my gratitude for an amazing computer science class. The knowledge and skills I've acquired in this class have been truly valuable and I am grateful for the opportunity to learn from such a dedicated and knowledgeable instructor. The class has not only helped me to develop my technical abilities, but also has helped me to think critically and creatively. I am grateful for the support and guidance provided throughout the course, and I am confident that the lessons learned in this class will serve me well in my future. Thank you for making this experience so memorable.