

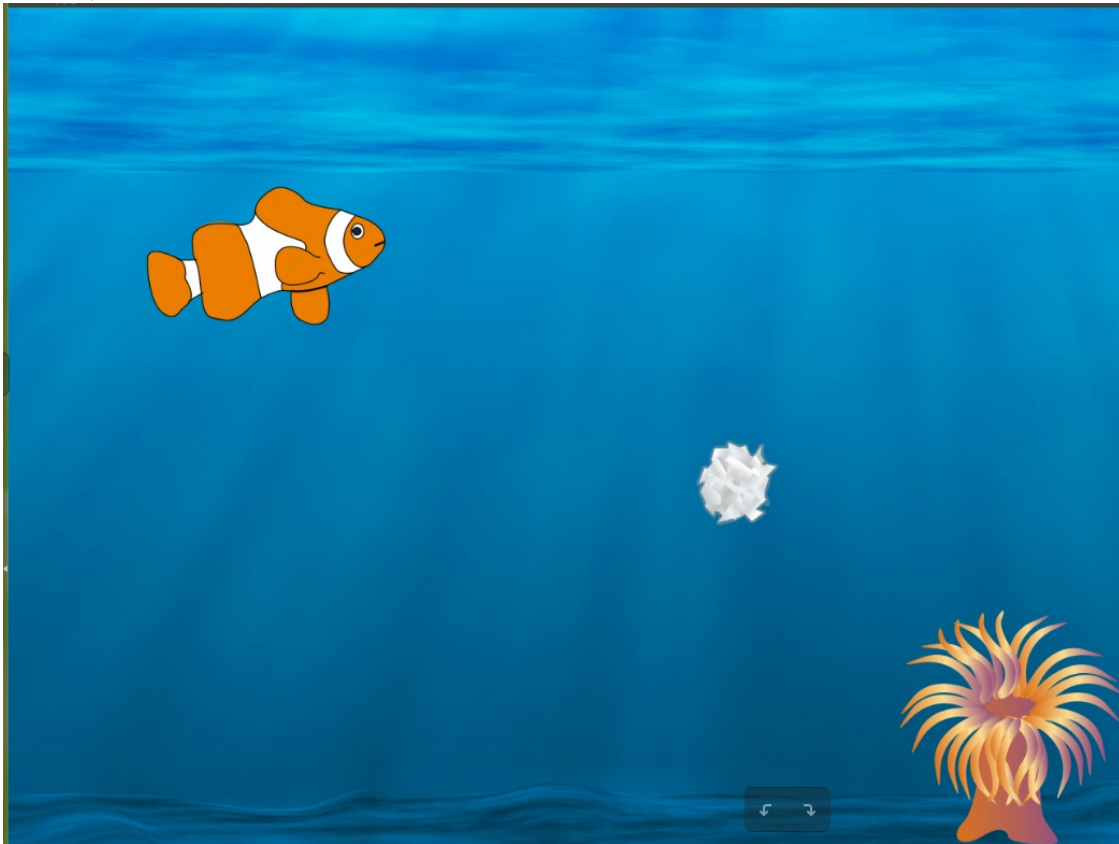
Project Reflection

Overview:

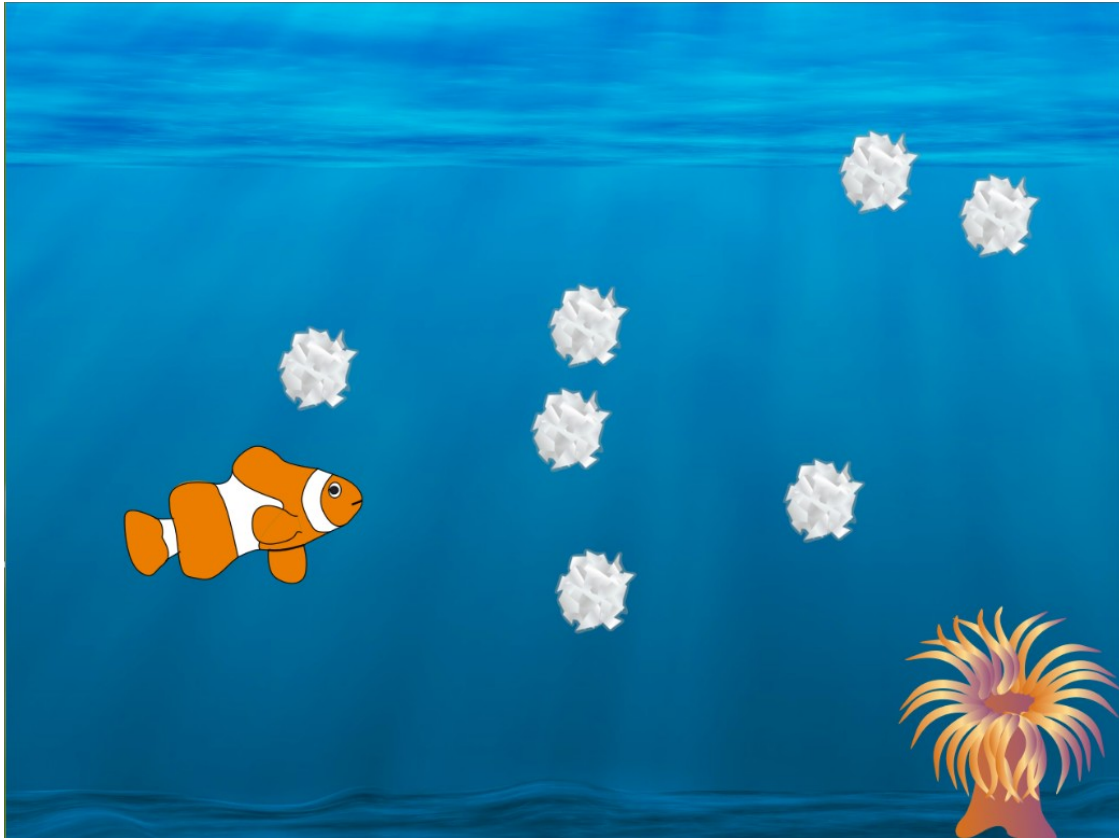
Our project is a cursor controlled collision game. A fish which is controlled by the player must be navigated around different ocean pollutants to get back to his sea anemone home on the other side of the screen. Reaching the sea anemone will move the player onto the next level, with more pollutants to avoid. Colliding with garbage will result in instance death for the fish and getting through five levels of the game means the player wins.

Results:

“Fishy Circumstances” is the game we created for Project 4. The player is a clownfish trying to reach his anemone home. The challenge that the player faces is the same that millions of fish in the ocean face: avoiding trash. The player moves by moving his cursor and has to touch his anemone home with the cursor, too.



The initial level of the game



A further level of the game

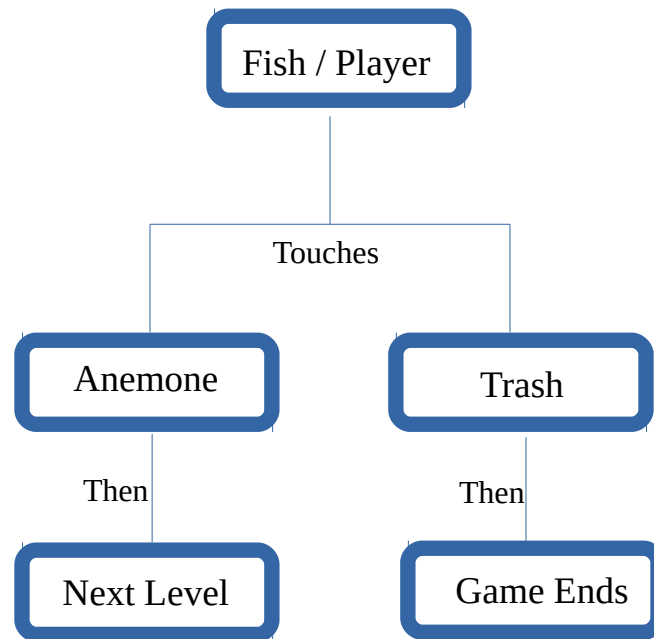
Hopefully, this game reminds the player of ocean pollution. Furthermore, this game should be fun because it should appeal to a player's sense of competitiveness, an aspect that most children studied by Massachusetts General Hospital identified as a motivating factor to playing video games.¹

In terms of programming, we accomplished having moving obstacles and a moving player. The player moves by tracking and following the position of the mouse and the obstacles move by randomly setting the position and the velocity. We also accomplished having the obstacles stay on screen by changing their velocity when they reach the end of the screen. Next, we got the screen to initialize a new level when the target was reached using a function.

¹http://ocw.metu.edu.tr/pluginfile.php/11519/mod_resource/content/1/gpr-14-2-180.pdf

Implementation:

UML Diagram:



The program consists of class objects for the target(the anemone home), the obstacles (trash), and a window viewer. The obstacles' position and velocity is random and the anemone home is set to always be at the bottom right. At each frame, the program checks if any value in the range of the player's image is in any value of the obstacle's position for each obstacle. There is also a score that is initialized to 0 at the start of the game and goes up by one with each level. Each new level is create using a function that resets the background and obstacles. Similarly, for each change in position of the player or obstacle, a new image of the characters and the background are blitted onto the screen.

The biggest design choice that we made was pivoting from our original game design, which was drawing a line for a ball to follow. The choice to make the game we did was a result of a few smaller design choices, from deciding to get rid of the drawing line aspect to getting rid of the following aspect of the ball to changing the story to match the new tools we were using.

Reflection:

Elements of our projects that went well were the graphics, team cohesiveness, and our ability to brainstorm way to work-around error. We struggled with identifying whether the strategies we were picking to solve problems were valid solutions. Due to our lack of background in pygame we ran into a lot of dead ends. This caused us to change from a line drawing where a character travels along a line to avoid obstacles to a character collision game. In our final game, a fish is controlled by the cursor and player navigates the fish around obstacles in the ocean with the goal of getting the fish to a target. A way to have avoided the large shift in our plan would have been to spent more time researching the different components of our project to confirm that everything we thought was possible was within our ability.

We originally planned on doing most of the programming together and while we always made design decisions together some of the code was worked on separately. Its sometimes took us a little while to make critical decision because we were both uncertain of the best way to continue.