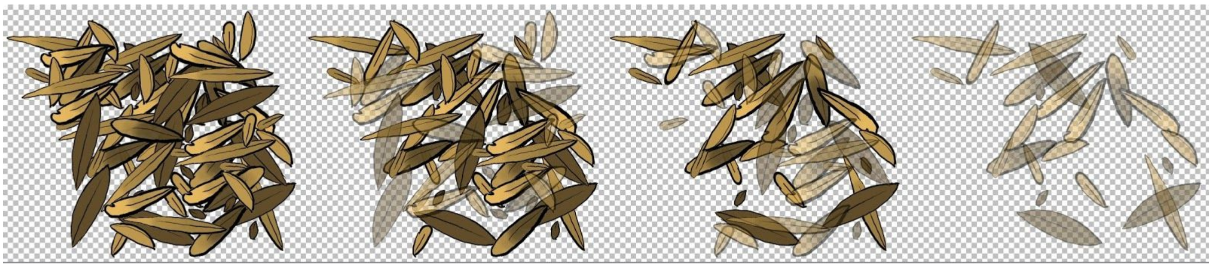
Órla Keating
15205679
COMP30540
Assignment 1

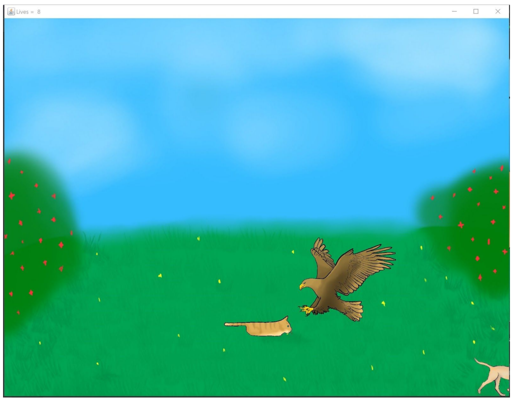
For this assignment I created my own images to use, these are all png files so that there is transparency in the space around the images. In these some of them I implemented animations of varying length. I also created an animation of an explosion of disappearing feathers, for when an eagle is killed.



Due to the theme of the game, it was changed from a top to bottom to more of a left/right game play. The main character could move in any direction but the enemies approached from the left or right.

The background image was changed and for this I made a slightly wider image than it is tall, as this gives more space to see enemies approaching, and gives more time for the player to react to events.

There are four difficulty options, easy, medium, hard and extreme. In the easy option there are two of every enemy which need to be defeated and the enemies move at a slow pace, except for the eagle which moves at the same pace as hovercat. The medium option has four of each enemy and the dog and intruder cat move twice as fast as in the easy option. In the hard option there are six of each enemy and they move at roughly the same as hovercat. The extreme option has 8 dogs and intruder cats and eighty eagles. The eagles also appear at the beginning of the game along with the dog in this difficulty. The image below shows the game on extreme difficulty.



For collision detection I created an areas class which detected collisions between two areas. Since each game object needed different behaviour depending on where they were hit, I separated out each object to its own class where they extend the game object class. These subclasses each had unique areas which were either attack areas or weak vulnerable areas. The collision detection method in areas was called from each of the areas in the enemies class and given an area of hovercat.

The first enemy encountered is the dog. This moves back and forth on the screen and needs to be chased off the screen, which only works when a weak area of the dog collides with an attack area hovercat. If both weak areas collide the dog will turn around and probably collide using an attack area. When an attack area on the dog collides with any of hovercats areas hovercat loses a life and is moved to a new location.

The second enemy is intruder cat. This enemy follows hovercat around the board and will keep attacking until killed. This does not have the weak area colliding with weak areas as intruder cat will turn around due to the following mechanics.

The last enemy is the eagle. This enemy is about as fast as hovercat and will fly to the spot where hovercat was when it appeared, once it reaches this point it will move upwards and to the side. This is the only enemy that exits the screen because of this a player could just avoid the eagles so instead I implemented in the code that the player needs to kill a certain amount of eagles depending on the difficulty, killing eagles also increases hovercats lives. This enemy can also enter from either side of the screen.

When, hovercat dies it will be go to a new area on the screen, for intruder cat and eagle this area will be random but for the dog this is an area out of reach of the dogs attack area.

There are two different types of controls for hovercat, one is the keyboard wasd which works for both lower and upper case and the mouse controller. The mouse controller works for touch screen laptops and tablets. This movement is achieved by using similar code to how intruder cat follows hovercat around. This is achieved by minusing the point where hovercat currently is with the point where the mouse currently is. The normal of this vector is stored in a variable, this vector is then applied to hovercats current location, with a multiplication of three. It is then checked if the sprites direction has changed, if the x variable of the direction vector to be applied is less than 0 then it is moving left depending on the previous direction, which is stored as a variable in the game objects class. This allows for a sprite facing a different direction to be generated.

I have also changed the dimensions of the screen to suit the new image and the gameplay. As enemies are attacking from the left/right direction more movement in those directions was needed.

Endings have been added for two scenarios, where the game is completed and where hovercat dies. These screens appear depending on the outcome.

Hovercat Adventures
Defend your territory
Mission status
Success

RIP
HOVERCAT