

## Assignment 7 – Progress Report

### Contents

Intended Animations .....	1
Animation Objects .....	1
Bee .....	1
Bird .....	3
PauseScreen .....	3
Splitting up and Integrating the Work .....	4

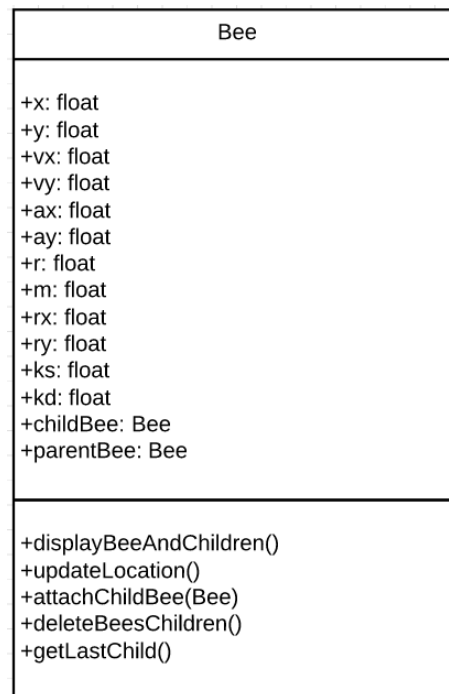
### Intended Animations

The story of our game is simple: the user controls a trail of bees, aims to collect more bees, and tries to avoid being eaten by birds. Bees are collected when the user-controlled trail of bees runs into a randomly spawned bee, similar to the game Snake. When one of the birds hits a bee, that bee will be eaten. That bee, along with any bees which are behind it in the trail of bees (its “children”), will cease to exist. The game is won when the user has attained a chain of 10 bees, the game is lost when all the bees have been eaten. A virtual display will show the current number of bees in the trail and a GUI will display the time elapsed. We will also include keyboard functionality, most likely the option to change the background color with the keys w, a, s, and d.

### Animation Objects

#### Bee

The bee class is completed and was implemented as a class of nodes, which each bee node being a particle which exhibits the behavior of a damped spring. By making an initial “queen bee” that the user controls directly and child bees which follow their parent bee, the trail of bees was implemented as a double linked list of these Bee nodes. The UML diagram for this class is shown below.

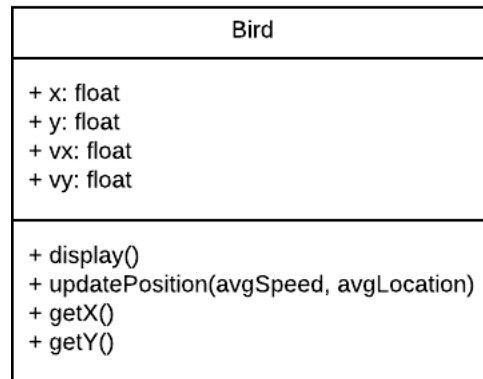


*Figure 1: Bee Class UML Diagram*

Because each bee node is modeled as a particle exhibiting spring-like behavior, each bee needs a position, velocity, acceleration, mass, spring stiffness, and spring damping coefficient. Also specified are the radius, child node, and parent node. These nodes allow the trail of bees to be implemented as linked list. The methods are fairly self-explanatory from the name but a few are worth mentioning. Because each bee is modeled as a particle attached to a spring, the spring needs a center of restitution. The first bee's center of restitution is the position of the mouse and each child bee's center of restitution is the position of the parent bee. The `updateLocation()` method uses the physics of a spring to move each bee. Attaching a bee to a parent bee causes the attached child bee to follow its parent. Deleting a bee's children deletes a bee's first child, which deletes all the children thereafter.

## Bird

The bird class is in progress and the UML diagram is shown below.

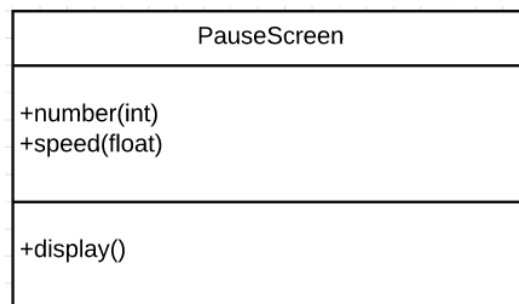


*Figure 2: Bird Class UML Diagram*

A new bird will be created in a fixed area, called the “bird nest”. It will then move in the general direction of the user’s trail of bees. If it hits a bee, the bird will disappear and the bees affected (along with their children) will be eaten. If the bird does not hit a bee, the bird will simply fly off screen and disappear before another spawns. There may be multiple birds on screen at any point.

## PauseScreen

The PauseScreen class encapsulates what will be shown on the screen when the user pauses the game. This class is in progress and its UML diagram is shown below.



*Figure 3: PauseScreen Class UML Diagram*

The pause screen shows animated bee sprites flapping their wing. This serves no purpose other than providing the user with a beautiful pause screen. An object of this class will take two arguments: the number of bees, and the speed at which the wings are flapped. This allows features of the pause screen to be easily customized.

## Splitting up and Integrating the Work

Each of the three group members is tasked with creating one of the class objects. Brandon has made the class Bee, Jonathan will be making the class Bird, and Alexis will be making the class PauseScreen.

In order to integrate implementations of these classes into a single scene, each object must have the ability to be animated without reliance on other scene objects. Additionally, the code for each class should be modular so that changes are easy to make. Once these classes have been designed, the team will meet to integrate the classes into a functioning game.