Assignment 4.2 Report

Jordan Yuen

301278171

Created two obstacles in the environment, one concave using GeneralPath and a complex shape using recursion. (AnimalPanel.java/ objList.add(new Complex(500, 50, 0, 0, 0)) , objList.add(new Concave(500, 500, 0, 0, 0)))

Both fish and predators attempt to avoid these obstacles using FOV and applying force onto the animal to steer it away. (Animal.java/methods: pushedObstacle())

Implemented a background image using Imageloader in the util package. The background is loaded and then drawn before all other objects. (ImageLoader.java/methods: loadImage())

Implemented micro-animations through the tail wiggling for both fish and predator. This is done by using a boolean to change between the two states the tail is drawn in. (Animal.java/variable: tail)

Implemented two states for fish and predator, hungry and satiated. Animals have a total of 2000 energy. When fish or predators are satiated, they move towards the nearest spot and rest. (Animal.java/methods: traceBestSpot())

These spots are invisible markers near one of the corners of the environment. When fish get below 1750 energy, they turn into the hungry state and begin hunting for food. (Animal.java/methods: traceBestFood())

When fish are below 1500 energy, they become weak and the maximum speed of weak fish is lower than predators. (Animal.java/variable: isHungry)

This allows predators to catch weak fish without them running away. Fish have the capability of running away if a predator enters its FOV. The FOV for fish however, does not cover its backside, so it is vulnerable from the back if weak. (Animal.java/methods: run())

Did not implement functionality for weak fish trapped in corners or multiple predators

Predators seek weak animals, as they cannot run away fast enough. Implemented three sorting options, energy level only, energy level and size, energy level and distance and size. These functions are implemented through the usage of Comparator/Comparable. The control panel gives users the option to choose which sorting option will be used. If no option is picked, the default implementation will be used. (ControlPanel.java/methods: sortByEnergy.isSelected(), sortByEnergySize.isSelected(), sortByEnergyDistSize.isSelected())

Clicking on a predator will display the weakest three fish from a sorted list. Predators hunt the weakest of the three fish. (Predator.java/methods: @Override traceBestFood())

Display information for Animals shows the type of the animal, the status of the animal (hungry or satiated) and its energy level. The color of the text changes depending on the energy level. (Animal.java/methods: drawInfo())

Implemented sounds effects for eating interactions and background noise using Minim. Control panel has options to mute each sound individually. (Sound.java/methods: play(), loop())