

FIT2099 Assignment 3

Design Rationale

By: Chong Chiu Gin (28842022) & Chin Wen Yuan (29975239)

1. New game map

A new game map is created and drew out in Application class. Since it was required that travelling to the north of the map will result with the player ending in the south of the new game map, we implemented a new type of Ground instance called EdgeMap with the symbol '^' as its display character. This EdgeMap extends Ground and does not allow any actor to enter. It is used to signify that the Player has reach the border to the other map and allows Player the option to choose if they want to go to the other map. The EdgeMap type is for all first row of ground in the old map (North) and the last row of ground in the new map (South).

2. Water terrain with reeds and fish

The new map introduced has a new type of terrain which is the water terrain. Water terrain is for marine life only and land dinosaurs are not allowed to move into water terrain area. Water class extends Ground and makes up more than 3 quarter of the new map so most of the marine lifeforms are implemented in the new map.

Every Water ground can that is next to land which are Dirt, Tree and Grass class has a chance to grow into reeds. In order to show this, we made Reed class extending Ground type too. Reed has a function called Overcrowded to check if a Reed has more than 6 adjacent neighbours. If yes, the Reed will die and the ground at that location will turn back into Water type. Just like Water, Reed does not allow land Actors to enter.

At every turn, a Reed has a chance in generating fish. Fish is used as food for marine dinosaurs. Fish extends Actor class since it will be moving around at random, it will call WanderBehaviour to return a MoveActorAction to move around at random. After 20 turns, Fish will call DieAction and disappear.

3. Plesiosaurs

Plesiosaurs are the marine dinosaurs implemented in this game. Just like the dinosaurs introduced in Assignment 1 & 2, Protoceratops and Velociraptors, this marine dinosaur also performs breeding and eating so this class extends Dinosaurs. The only difference is Plesiosaurs are not allowed on land. So Dirt, Grass and Tree class had override their canActorEnter method to not allow Plesiosaurs (as well as Fish) to enter this Ground type.

Plesiosaur eggs bought from Shop has to be placed on a land next to Water or Reed in order for the eggs to hatch. This is so that the hatched dinosaur can start living in water terrain. In the Egg class, the tick function for Plesiosaurs is implemented slightly different to allow this functionality.

Plesiosaurs behaves just like the other dinosaurs. So getAllowableAction and playTurn of Plesiosaurs is implemented in the abstract class Dinosaurs to avoid repeating code. When

hungry, it will have a HungryBehaviour to find nearest food. Plesiosaurs will look for Fish, Marine Food, Corpse to eat. If not hungry, Plesiosaurs will breed egg in water.

4. Pteanodons

Pteanodons are flying dinosaurs and the only Actor that can travel to both land and water terrain. Pteranodons can move fast so it is implemented in WanderBehaviour that it will move two steps. When it is HungryBehaviour, it will look for nearest food and is able to move to locations that is within two steps in one turn. Pteranodons eats fish just like Plesiosaurs and is able to breed anywhere on the game. It's implementation for getAllowableAction and playTurn are the same to avoid repeated code.

5. T-Rex

Trex class also extends Dinosaur. Since it is also a carnivorous dinosaur, most of its implementation is similar to Velociraptors except for its attributes like foodLevel capacity and hungerLevel. In order to win the game, we need to know if we managed to successfully raise a captive- breed dinosaurs into adulthood. So we introduced a boolean attribute to dinosaurs to indicate if they are captiveBreed or not.

6. Ending the game

There is three way to end the game.

- a) Player can choose to quit game manually in the displayed menu. A QuitGameAction that extends Action class is implemented to allow Player to choose the option if they want to quit playing at every turn. QuitGameAction removes the player from the game map so the game will not run anymore and ends.
- b) Game will end when Player wins the game. Game is considered a win if Player can raise a captive bred T-Rex into adulthood. A new class called GameWorld that extends World from engine code is created. In this new class, there is a method call checkWin() that is used to check if Player has reached the condition to win. This method is placed under run() method and will be called after every round of turn. If the player win, the Plyer is removed from the GameMap and game ends by printing a 'Player wins!' message.
- c) Player can also lose in this game if the hitpoints of Player reaches 0. Player can lose hitPoints if they got attacked by dinosaurs. Only T-Rex, Velociraptors and Pteanodons can attack Player. This may happen when these dinosaurs are hungry and Player is standing next to them. Once Player loses consciousness (hitpoints = 0), the game ends and an endgame message will be displayed.