African Savanna

By Mehdi Nya Louhaichy

## General info about the classes

For the final project I decided to recreate the African Savanna. In this project we will find an abstract class called “Animal” and two concrete classes that will be derivate from “Animal” called “Lion” and “Gazelle”. These two classes would share some abstract methods, but they will behave in different ways. Before we start talking about the behaviors, first I need to talk about the class “Savanna”. This class will contain a matrix 100 by 100 of “Pair” objects, “Savanna” will work as our simulator class. To implement the class “Pair” we will use an auxiliary class called “Coordinates” that will help handle position inside the array and the 3 different regions inside the savanna (bush, ground, and water). The pair will be a couple Animal-Coordinate.

## Behavior Lion-Gazelle

Lion will be in the top ground region living and walking around with his herd, until his life gets low because of the hunger, then he will enter in hunting mode, moving towards the bush region where he will await patiently for his pray, a tasty gazelle. After a successfully hunt he will return to the ground region.

Gazelle will be in the low ground region (under the bush region) walking freely until he gets thirsty or scared by the appearance of a lion, then he will walk towards the water region where he will be able the drink (if he is thirsty) or relax if he was just scared by a lion.

They both will replicate a couple of times during their lifetime (amount of times still looking for a good number).

Color will change depending on their health (3 states).

## Coordinates, Pair, and Savanna

Coordinates will hold the position and the color of the position.

Pair will hold a pair Animal-Coordinates.

Savanna will be a matrix 100 by 100 were we will be calling a status method on all animals for each loop (or generation). During the filling process of the squares we will be calling a “getColor()” function that will be looking thru the pairs and filling with the animal color or the coordinate color (if there is no animal in the pair).

## Future changes

We will work on separate the characteristics of the habitat from coordinates into another class (“Habitat”). After that change, the pair will be Animal-Habitat.

We also still working on how to check if there is an animal on the surrounding pairs (during the moving and hunting process).

Solution: putting null animals in all pairs and check if the animal part of the pair is null in order to be able to move or hunt or attack.

## UML Diagram

\*Setters and getters (modifiers) not included on the UML diagram.

The UML diagram is on the next page.

