EE422C Project 4 (Critters) Code Structure

Marcus (Yilin) Zhu, *yz22778*

Andrew Wong, aw27772

Git URL: https://github.com/Marcus-Zhu/EE422C\_HW4\_Critter/

Fall 2016

1. Did you create any new classes, and if so, what fields and methods are in it?

We have added the following classes:

* Four new critter classes:
  + Each MyCritter method contains a doTimeStep(), fight(), and toString() method. There fields for each critter class are listed below.
  + MyCritter1
    - Fields: lastFightTime = -1, lastDir = 0
    - Methods:
      * MyCritter()
      * doTimeStep() - keep away from fighting and reluctant from moving, but likes reproducing.
      * fight() – only fights if it has enough energy to reproduce and it has been more than 10 steps since its last fight
      * toString() – returns “1”
      * runStats() – returns the number of happy and sad critters
  + MyCritter2
    - Fields: mainDir = 0
    - Methods:
      * MyCritter2() - constructor
      * doTimeStep()
      * fight() – always returns false
      * toString() – returns “2”
      * runStats() – returns the number of left and right critters
  + MyCritter3
    - Fields: myDir = 0
    - Methods:
      * doTimeStep()
      * fight() – only fight if facing north or south
      * toString() – returns “3”
  + MyCritter4
    - Fields: myDir = 0
    - Methods:
      * doTimeStep()
      * fight() – always fights
      * toString() – returns “4”

1. What is the data structure that you used to hold your Critters?

We used the provided population ArrayList to hold Critters. When solving map conflicts and displaying critter world, we use a hashmap to store pairs of (position, critter).