Andrew Thomas, Brandon Turp, Colin Sullivan, Marisa Gilman, Jasmine Sullivan

INST326

4/20/2017

Wolves and Sheep Project Update #1

1. Your status update (Update1.doc for week 1) - provided numbered sections with bullet lists for each of:

* What you have done since last report
  + Andrew
    - Created a channel on Slack and posted a bulleted list of the steps required to create the program.
    - Created a GitHub repository for all the files used in the project. Also created a turtle window and drawn a 600x600 square for the objects (wolf, sheep, dog) to be bounded by. Displayed three animal objects and their respective images on the screen.
  + Marisa
    - Debugged the problems we had with getting the Sheep, Wolf, and Dog images to display in the turtle module.
  + Brandon
    - Created a google doc, walked through GitHub tutorials, and got animal images to display in the turtle module. Got Animals to move away from one another in a T-shape.
  + Colin
    - Corrected the file path for the sheep, wolf and dog images for our simulation with the turtle module.
    - Familiarized myself with GitHub so we can share code for our project
  + Jasmine
    - Debugged and added the wolf,dog and sheep gifs to my program.
    - Got myself familiar with GITHUB where we will use and post all of our important files.
  + TEAM
    - Successfully demonstrated random movement in all three cursors and have made sure that the animals don’t cross over the boundaries. The animal images move as the cursor does.
      * <https://codereview.stackexchange.com/questions/74495/random-walk-in-python-turtle>
    - Randomly generated the variables for dog\_strength and wolf\_strength on a scale of 1-50.
    - The code necessary for the interactions between the dog and the wolf, and the wolf and the sheep has been written.
* What you plan to do for next week
  + We must declare the initial sheep as a “leader and then will generate two additional sheep that will follow the leader. They will stay within a certain distance.
  + Once the three sheep operate together as a flock we will create reproduction between them. To save myself the realism of having male and female sheep, the sheep will generate a new sheep every time they occupy the same space. The sheep will need to spawn within the vicinity of the interaction itself.
  + Lastly, we will create a condition where if the current\_sheep\_pop reaches maximum\_sheep\_pop then the simulation should end in favor of the sheep. We will create a dictionary to keep track of the sheep’s victory.
* Roadblocks, problems, challenges, questions
  + One challenge to face is the movement of the flock.
    - First, we must find a way to make the sheep spawn near their leader as the program starts.
    - Then we must make their random movement follow a certain bound. They must not get “x” distance away from their leader.
    - If they get “x” distance away then they will use the “go to” function to move back towards the leader but not so much that they land on top of it.
  + Another challenge to face is the reproduction of the sheep.
    - We must tackle the obstacle of generate up to seven new sheep. We will either initialize seven sheep functions to be called when an interaction occurs, or we will need to start using objects in a way that I can create new instances.

2. A snapshot of all project files - code, docs, resources, links to Google docs, Github, etc.

<https://github.com/INST326WolvesVsSheepSimulationProject/Project-Files>