Spike Outcome Report

--------------------

Number: 12

Spike Title: Graphs and Search

Personal: Chloe Mitterer 9725725

Goals:

Clearly demonstrate the appropriate use of the Dijkstra’s (search for item) and A\* (search to position).

• Modify the graph search lab code, or create your own simulation.

• Add in a simple moving agent to moves to each way-point in a successful graph search result.

• Be able to demonstrate either search-for-item or search-to-point examples.

• Make sure your agents (or their graph searching algorithms) correctly consider wall, mud or water tiles in the map.

• Display path cost for comparison

• Clearly demonstrate the need for different search algorithms

Technologies, Tools, and Resources used:

* Code from lab 10
* Sublime Text 3
* Python v3

Tasks undertaken:

* Create an Agent class that allows for initialising, rendering and updating
* Make the agent update based on the destination
* Once the agent follows the path, create a copy of the files to edit to make into the search for item version
* Change box\_world to allow for multiple targets, by changing target to targets[] (a list)
* Change search to look for the first of multiple targets.

What we found out:

* How to search using multiple items
* How to make an agent follow a search path
* How Dijkstra’s and A\* work better for different tasks
* How Dijkstra’s and A\* differ