**Spike:** 12

**Title:** Graphs and Search

**Author:** Steven Efthimiadis, 1627406

**Goals / deliverables:**

Demonstrate the use of Dijkstra’s (search for item) and A\* (search for position):

* Modify the lab 10 code
* Add an agent that follows a path for a successful search result
  + Movement doesn’t have to be fancy (same movement as planet wars is good)
* Able to search for an item or points on the map
* Display the cost of the path
* Demonstrate the need for both algorithms

**Technologies, Tools, and Resources used:**

* Knowledge of python
  + <https://docs.python.org/3/tutorial/>
* Python Interpreter
  + Visual Studio
    - <https://www.visualstudio.com/downloads/>
* Knowledge of search algorithms
  + <https://artificialintelligentsystems.wordpress.com/category/ai-searching-techniques/page/2/>
  + <https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_popular_search_algorithms.htm>

**Tasks undertaken:**

* Adjust the search file to only have A\* and Dijkstra’s alogrithms
* The target agent:

**What we found out:**

* The agent has good accuracy when the target is traveling in a straight line
* The different bullets types are working as expected
* A visual affect when the target is hit. You see it changes colour and slows down

**Open issues/ Risks:**

* When the target changes direction the agent has, trouble predicting the change
* Some bullets that clip the agent don’t disappear straight away. Change the radius to a slightly bigger one on the target so when you look at the screen it is removed before it passes through your target

**Notes:**

World Keys:

Q – Hanzo Main (Slow, Inaccurate)

W – Tracer Main (Fast, Inaccurate)

E – Ana Main (Fast, Accurate)

R – Symmetra Main (Slow, Accurate)

T – Make the target move

Y – Freeze Agent

S – Shot

D – Move Shooting Agent

G – Auto Fire (For the lazy person or for using Symmetra as seen in Overwatch)

H – Remove Bullets

### Appendix

#### Figure 1.1 Agent hitting the target

