## Chris Nolan Assignment 4

## Problem One:

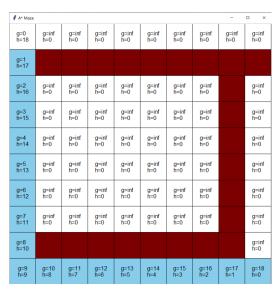
Modify AStarMaze to compare the behaviors of the Greedy Best-First and A\* search
algorithms. You need to modify the maze configuration so you can visually observe
differences in the optimum paths generated by the two algorithms. Your report should
include a side-by-side comparison of the two approaches similar to the graph shown
below along with your explanation. You only need to draw the shortest paths and not the
highlighted frontiers.

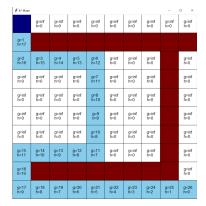
Maze Used in example:

```
maze = [
    [0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
    [0, 1, 1, 1, 1, 1, 1, 1, 1, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 1, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 1, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 1, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 1, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 1, 0],
    [0, 1, 1, 1, 1, 1, 1, 1, 1, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]]

Start position = (4,3)
```

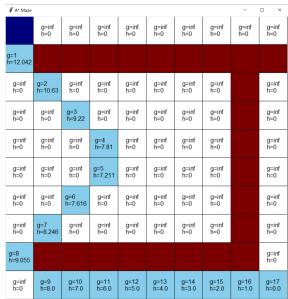
- Goal Position = Bottom right
  - Solution: To make the A\* greedy I just simply modified the algorithm so that it only uses the heuristic to make the movement decision
  - A\*



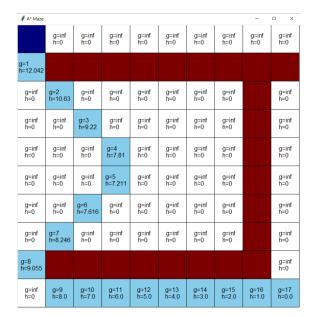


## Problem 2:

- Use the Euclidean Distance heuristic.
- The agent is allowed to make diagonal moves (i.e., NE, NW, SE, SW) in addition to the usual N, S, E, and W moves.
  - To do this I added the following directions to the move list (1, 1), (-1, 1), (-1, -1)
- The moves are made randomly and not in any specific order.
  - To randomize the move I used the numpy.shuffle to perform this before each direction is considered
- Greedy:



- A\* Euclidean Distance heuristic
  - I tried many different possible configurations and I could not seem to find one that produced a change in the way the maze was solved



Problem 3:

• The evaluation function in AstarMaze is defined as f(n) = g(n) + h(n). A weighted version of the function can be defined as:  $f(n) = A^* g(n) + B^* h(n)$ 

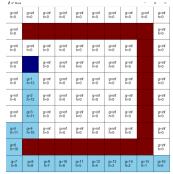
I added a scaler to the calculate function as described in the equation above.

B can be considered the algorithm's bias towards states that are closer to goal. Run the
algorithm for various values of the bias to determine what changes, if any, are observed
in the optimum path. Include screenshots of the path for each specific value of B along
with your explanation.

 Using the table below I found that changing the B value managed to increase the bias the program had to moving to the right first to find the goal state. In the first example where B is 5 it moved 1 space right before moving down and in the second example, it started to move 2 spaces to the right before moving down.

• See how changing A and B effect the model

No Changes made to the program (Control)



A B Behavior

		Ι
1	20	# A-Mage  gaint ga
		9+inf b=0   9-inf b=0   9-inf
		9+inf
		grind g=1 g=2 g=ind g-ind g=ind g-ind g-ind h-0 h-0 h-0 h-0 h-0
		g-inf g-inf g-inf g-inf g-3 g-inf g-inf g-inf g-inf g-inf h-0 h-0 h-220 h-0 h-0 h-0 h-0 h-0 h-0 h-0
		g-inf
		g-inf
		g:9 g=8 g=7 g=8 g=7 g=8 g=inf g=inf g=inf g=inf h=220 h=200 h=180 h=10 h=0 h=0 h=0 h=0 h=0
		g=10 h=200 b=0
		g=11 g=12 g=13 g=14 g=15 g=16 g=17 g=18 g=19 g=20 h=180 h=180 h=180 h=180 h=180 h=180 h=180 h=20 h=190 h=180
1	5	#AMM
		9-inf
		grief held
		gaint
		9-16 9-17 9-16 9-16 9-16 9-16 9-16 9-16 9-16 1-0 h=0 h=0 h=0 h=0 h=0 h=0 h=0 h=0 h=0 h=
		Grief   Grie
		Grief   Grie
		h=55 h=50 h=45 h=0 h=0 h=0 h=0 h=0 h=0 h=0 g=inf h=50
		p=9 g=10 g=11 g=12 g=13 g=14 g=15 g=16 g=17 g=18 h=45 h=40 h=35 h=30 h=25 h=20 h=15 h=10 h=5 h=10 h=5
		h=45 h=40 h=35 h=30 h=25 h=20 h=15 h=10 h=5 h=0
E	4	<b>#</b> A* Moce
5	1	grief
		g-liff h=0
		g:inf
		g-inf g-inf g-inf g-inf g-inf g-inf g-inf g-inf g-inf h=0 h=0 h=0 h=0 h=0 h=0
		g:inf
		g=inf g=inf g=inf g=inf g=inf g=inf g=inf g=inf g=inf h=0 h=0 h=0 h=0 h=0
		grinf g=3 grinf grinf grinf grinf grinf grinf grinf grinf grinf h+0 h+0 h+0 h+0 h+0 h+0 h+0 h+0
		g=5 g=4 g=inf g=inf g=inf g=inf g=inf g=inf g=inf g=inf h=0 h=0 h=0 h=0 h=0 h=0 h=0
		g-id h-10
		g=7 g=8 g=9 g=10 g=11 g=12 g=13 g=14 g=15 g=16 h=9 h=8 h=7 h=6 h=5 h=4 h=3 h=2 h=1 h=0
		No change noticed
		0
20	1	# A* Mass - D X
		g-inf   g-in
		guint h=0 guint h=0 aguint naint naint naint naint naint naint naint
		Grief   Grie
		grief
		5=0 h=12 h=0
		9:6 6:10 6:10 6:10 6:10 6:10 6:10 6:10 6:
		p=10 h=10 h=0  g=7 g=8 g=9 g=10 g=11 g=12 g=13 g=14 g=15 g=16 h=7 h=6 h=5 h=4 h=3 h=2 h=1 h=0
		No change noticed
		i e e e e e e e e e e e e e e e e e e e