

Student Corner:
[\[Grades Sec1\]](#)

[\[Submit Sec1\]](#)

[\[Lecture Notes\]](#)

[\[Discussion Board\]](#)

Course Info:

[\[Texts & Links\]](#)
[\[Topics/Outcomes\]](#)
[\[Outcomes Matrix\]](#)
[\[Grading\]](#)
[\[HW/Quiz Info\]](#)
[\[Exam Info\]](#)
[\[Regrades\]](#)
[\[Honesty\]](#)
[\[Additional Policies\]](#)
[\[Announcements\]](#)

HWs and Quizzes:

[\[Hw1\]](#) [\[Hw2\]](#) [\[Hw3\]](#)
[\[Hw4\]](#) [\[Hw5\]](#) [\[Quizzes\]](#)

Practice Exams:

[\[Mid\]](#) [\[Final\]](#)

HW#1 --- last modified

Wednesday, 27-Aug-2014 10:34:54 PDT.

[Solution set.](#)

Due date: Sep 15

Files to be submitted:

Hw1.zip

Purpose: To gain experience with the A* algorithm by making an agent capable of tracking a target.

Related Course Outcomes:

The main course outcomes covered by this assignment are:

LO1 -- By code or by hand find solution nodes in a state space using the A* algorithm.

Specification:

p>For this homework, you will build a food finding agent in python that uses the A* algorithm. Your python code should conform to the [Pep8](#) coding guidelines and should work if Python 2.7.* is being used. You can work in teams of up to three people. Your program will be run from the command line with a line like:

```
python food_agent.py file_name heuristic
```

Here file_name is the name of a file which contains a maze drawn using ASCII character graphics; heuristic is one of the words manhattan, euclidean, or made_up. Each line of the maze will be drawn using one of the characters @ - representing the agent, . - representing an empty square, # - representing a wall, or % representing a bowl of Ramen. Lines are distinguished from each other by a single newline character (not CRLF). The maze will contain at most one @ and at most one % and your agent's goal is to come up with a sequence of steps from the agent location to the Ramen location that only traverses empty squares. In one step, the agent is allowed to move up, down, left, or right and can only move into an empty square. After moving, the square the agent was in before becomes unoccupied and the one moved to becomes occupied. As an example, an initial board might look like:

```
@..#  
#.%#
```

If the agent is on the boundary of the bound it is not allowed to move in a direction that would take it off of the board. When run on such a file containing such a board your agent should run the A* algorithm using the given heuristic distance function (manhattan distance, euclidean distance, or your own made_up distance function). If there is no path that takes the agent to the Ramen, your program should output no solution; otherwise, it should draw a sequence of mazes showing the steps the agent needs to take to get to the solution. For example,

Initial:

```
@..#  
#.%#
```

Step 1:

```
.@.#  
#.%#
```

Step 2:

```
..@#  
#.%#
```

Step 3:

```
...#  
#@#
```

Problem Solved! I had some noodles!

Please be sure to add the words "Initial", "Step 1", etc in what you draw. Be aware your program will be tested on boards which are not solvable such as:

```
@#%
```

as well as board such as:

```
.....  
.#####.  
.#.....  
.#.###.  
.#.@.#%  
.###.##  
.....#
```

The teams with the best made_up function on the sequence of tests used for your homework will receive 1 bonus point. That completes the description the homework. Good Luck!

Point Breakdown

The team with the best made_up function on the sequence of tests used for your homework will receive 1 bonus point. That completes the description the homework. Good Luck!

Point Breakdown

| | |
|--|-------|
| PEP 8 coding guidelines followed. | 1pt |
| Code is split into reasonable function, classes, etc and is well commented. | 1pt |
| Program reads in input maze correctly. | 1pt |
| Can find in Python code an implementation of A* search. | 1pt |
| Can find in Python code an implementations of each of the three mentioned heuristics (1pt each). | 3pts |
| Output corresponds to spec above and should exactly match on examples above. | 1pt |
| Program works on all my tests cases where there is a solution. | 1pt |
| Program works on all my tests cases where there is a no solution. | 1pt |
| Total | 10pts |