Project #5

Robot Trash Collector

Introduction

- NASA hires your robot explorer to travel to another remote planet called Earth
- Your robot has been given the job to clean up the Earth's environment
- Your job: To learn the optimal strategy for picking up cans in the environment, without crashing into walls.

Your Task

- Design a learning algorithm based on the following (input.dat):

  MAP width height
  - Defines the width and height of the world where (0,0) represents the top, left corner of the map

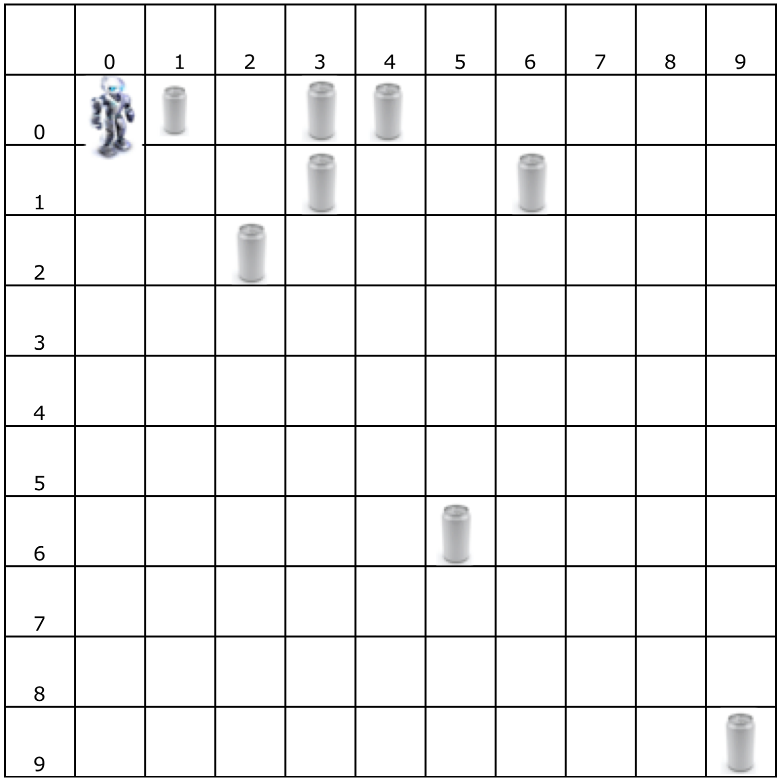
  ACTION maximum
  - Defines the maximum number of actions possible for the robot to implement

  Start x_position y_position
  - Defines the starting (x,y) position of the robot

  Can x_position y_position
  - Defines the (x,y) location of a can

Example input.dat

```
MAP 10 10
ACTION 25
START 0 0
CAN 1 0
CAN 3 0
CAN 4 0
CAN 3 1
CAN 6 1
CAN 2 2
CAN 5 6
CAN 9 9
```



Output.dat

- Given the map, your program must output to “output.dat”
  - sequence of *x, y* grid points that your robot traverses
  - corresponding action at each of those grid points
  - total number of cans collected
  - total number of actions
  - total score associated with the traversed sequence
- Note
  - The robot has seven possible actions: move north, move south, move east, move west, move in random direction, stay put, or pick up can. (Hint: A robot can perform sequential actions in one cell)
  - Score Computed By:
    - Picking up can = 10 points
    - Picking up non-existent can = -1 point
    - Moving = -1 point
    - Crashing into a Wall = -5 points

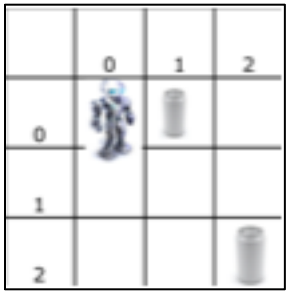
Grading Criteria

- Optimal credit (20 points for each case): Program outputs an optimal strategy based on based on collecting all of the possible cans in the minimum number of actions and bounded by a maximum number of actions. The program must run within 1.5 minutes.
- Suboptimal credit (17 points for each case): Program outputs a sub-optimal strategy based on collecting all of the possible cans but not in the minimum number of actions. Robot is bounded by a maximum number of actions. The program must run within 1.5 minutes.
- Partial credit (14 points for each case): Program outputs a sub-optimal strategy based on achieving a correct score (as discussed above) bounded by the maximum number of actions. If you believe your program will take longer than 1.5 minutes to run, you must indicate that via a message to the console.
- Deductions
  - Program will not compile (60 pts)
  - Program crashes or does not terminate (20 pts)
  - Program does not contain at least one class file (20 pts)
  - Proram does not comply with good design constraints or requirements (2-10pts)

Example Scenario

Input.dat

```
MAP 3 3
ACTION 18
START 0 0
CAN 1 0
CAN 2 2
```



Learned Strategy

2 6+2 1 2 2 1 2 2 6+5

0 = move north

1= move south

2 = move east

3 = move west

4 = move random

5 = stay put

6 = pick up can

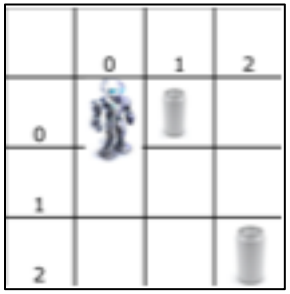
Optimal Output.dat

```
0,0 move east
0,1 pick up can
0,1 move east
0,2 move south
1,2 move south
2,2 pick up can
Actions: 6
Can: 2
Score: 16
```

Example Scenario

Input.dat

```
MAP 3 3
ACTION 18
START 0 0
CAN 1 0
CAN 2 2
```



Learned Strategy

6+2 6+2 6+1 6+3 6+3 6+1 6+2 6+2 6

0 = move north
1= move south
2 = move east
3 = move west
4 = move random
5 = stay put
6 = pick up can

SubOptimal Output.dat

```
0,0 pick up can
0,0 move east
0,1 pick up can
0,1 move east
0,2 pick up can
0,2 move south
1,2 pick up can
1,2 move west
…
Actions: 18
Cans: 2
Score: 4
```

Submission

- A zip file containing your code is to be submitted via T-Square (ECE3090-Assignments-Project5) by the DUE date of November 16th/21st
- We will test your code on the Jinx cluster so make sure your program correctly compiles and runs on that system (development can be done on your machine - e.g by using GNU ARM). Information on the cluster is located at: http://support.cc.gatech.edu/facilities/instructional-labs/jinx-cluster
- Do your own work and abide by Georgia Tech's Honor Code!!

Dates

- Nov. 14th:
  - Final Project Released + Exam #2 Review
- Nov. 16th:
  - Project 5 Early Due Date (+10 extra credit points)
- Nov. 19th:
  - Exam #2
- Nov. 21st:
  - Project 5 Due Date
  - No class - Work on your final project
- Week of Dec. 3rd – “Dead Week”
  - In-Class Help Session: Dec. 3rd and 5thth. No class Dec. 7th.
- Dec. 10th:
  - Class Survey Due (75% class participation)
  - Final Project Due Date