BotClean

by dheeraj

Problem Submissions Leaderboard Discussions

Problem Statement

The goal of Artificial Intelligence is to create a rational agent (Artificial Intelligence 1.1.4). An agent gets input from the environment through sensors and acts on the environment with actuators. In this challenge, you will program a simple bot to perform the correct actions based on environmental input.

Meet the bot **MarkZoid**. It's a cleaning bot whose sensor is a head mounted camera and whose actuators are the wheels beneath it. It's used to clean the floor.

The bot here is positioned at the top left corner of a 5*5 grid. Your task is to move the bot to clean all the dirty cells.

Input Format

The first line contains two space separated integers which indicate the current position of the bot.
The board is indexed using Matrix Convention
5 lines follow representing the grid. Each cell in the grid is represented by any of the following 3 characters: 'b' (ascii value 98) indicates the bot's current position, 'd' (ascii value 100) indicates a dirty cell and '-' (ascii value 45) indicates a clean cell in the grid.

Note If the bot is on a dirty cell, the cell will still have 'd' on it.

Output Format

The output is the action that is taken by the bot in the current step, and it can be either one of the movements in 4 directions or cleaning up the cell in which it is currently located. The valid output strings are LEFT, RIGHT, UP and DOWN or CLEAN. If the bot ever reaches a dirty cell, output CLEAN to clean the dirty cell. Repeat this process until all the cells on the grid are cleaned.

Sample Input #00

```
0 0
b---d
-d--d
--dd-
--d--
----d
```

Sample Output #00

```
RIGHT
```

Resultant state

```
-b--d
-d--d
--dd-
--d--
----d
```

Sample Input #01

```
0 1
-b--d
-d--d
--dd-
--d--
----d
```

Sample Output #01

```
DOWN
```

Resultant state

```
----d
-d--d
--dd-
--d--
----d
```

Task

Complete the function next_move that takes in 3 parameters *posr*, *posc* being the co-ordinates of the bot's current position and *board* which indicates the board state to print the bot's next move.

The codechecker will keep calling the function next_move till the game is over or you make an invalid move.

Scoring

Your score is (200 - number of moves the bot makes)/40. CLEAN is considered a move as well.

Once you submit, your bot will be played on four grids with three of the grid configurations unknown to you. The final score will be the sum of the scores obtained in each of the four grids.

Education Links

- Introduction to AI by Stuart Russell and Peter Norvig
- Motor cognition

Solved score: 17.82pts

Submissions: 6699

Max Score: 17

Difficulty: Easy

More

Current Buffer (saved locally, editable)

Scala

```
object Solution {
    def main(args: Array[String]) = {
        val pos = Console.readLine
        val board = new Array[String](5)
        for (i <- 0 until 5) {
            board.update(i, Console.readLine)
        }
    nextMove(pos, board)
    }

  def nextMove(pos:String, board:Array[String])={
  }
}

```

Line: 1 Col: 1

Upload Code as File

Run Code

Submit Code

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Privacy Policy | Request a Feature