**Project document for AI vacuum project**

For this project I began with creating a simple Boolean array with dimensions 4x4, this was done in the main method. I added a vacuum class which has stored three variables. It has its x and y coordinates as integers between 0 and 3, as well as an integer indicating what the previous move was. This was added so that the vacuum couldn’t move back and forth. The reason why I chose to store its location variables in the vacuum class was to only have to store it in one place rather than every single square needing another Boolean variable for occupied/unoccupied. When the board is created, I wanted the vacuum to start on a random square so I added a randomizing method that would randomize where the vacuum started.

Next, I added a method that simply assigned a given square a state dirty(true) or clean(false) by using a randomizer with 50/50 odds. To print the matrix, I added a gridPrinter that would call upon a method that would print “Dirty” for true, and “Clean” for false. This method also puts square brackets, “[]”, around the square in which the vacuum is currently located. This is done by checking if the vacuum.locationX/Y is equal to the current square and then printing the square brackets as well.

The vacuum needs to move, but to make sure that the vacuum suddenly doesn’t exit the board and to prevent null exceptions, an isLegal method had to be added. This simply makes sure that the new location variables aren’t greater than 3 or smaller than 0. This method is called upon in the move() function. The move() function randomly picks either up/down/left or right and then checks whether that move is legal and then moves. If it isn’t legal, then it cycles through to the next move and checks whether it is legal, if it is, it then changes the location to the new square and stores what the opposite move would be for the next move so that it doesn’t move straight back to this square.

Finally, the main function of the entire program adds all these functions together to make one coherent method in which the vacuum’s location, percept and action is as well as the current state of the board. In the main method a Boolean matrix and vacuum object are created and randomized, and then this main function is repeated x-times depending on how iterations I choose.

**1.     When does your vacuum stop cleaning?**

My vacuum is set to perform x-iterations of precepting and performing an action, and in my testing methods I set it to stop cleaning after 25 iterations.

**2.     Will your vacuum clean all of the squares in all instances?**

No. Since I made the vacuum move at random, imitating these automatic vacuum machines or lawnmowers, the vacuum may or may not clean all of the squares, its all random.

**3.     Does it clean all of the squares in the most efficient number of moves?**

No. Since it moves randomly and can only percept the square in which it is currently in, it does not clean all of the squares in the most efficient number of moves.

**4.     What if the grid was larger?**

If the grid was larger it could end up just circling in one corner and not clean efficiently, but this is all random.

**5.     Will your agent continue moving over clean squares.?**

Yes, the vacuum will continue moving over clean squares. It might not even clean half of the squares if we are unlucky with the randomization.

**6.     What is a good performance measure for your vacuum agent?**

Some good performance measures for my vacuum agent could be the state of the board when its randomized and then after it is done cleaning. If the board is really dirty at first, and then quite clean, then the vacuum has done a good job. Another performance measure could be the ratio of cleaning and moving to simply moving, meaning how many clean squares it passes per dirty square, but this might not be as viable because the squares cannot get dirty again, so we would eventually cross more clean squares.

**Output:**

This output was a good run by the vacuum in that it managed to clean all the squares, on average it would be around every 6th or so run would yield a fully cleaned grid, and then on average there would be around 1 or 2 dirty squares left and sometimes more.

Here is the sample output:  
"C:\Program Files\Java\jdk-14\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.1.2\lib\idea\_rt.jar=55483:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.1.2\bin" -Dfile.encoding=UTF-8 -classpath "C:\Users\Bruker\IdeaProjects\Artificial Intelligence project 1\out\production\Artificial Intelligence project 1" ProtoVacuum  
----------------------------------------------------------------------  
Current location: (0,2)

Percept: The square is dirty

Action: Clean then move

Dirty Dirty [Dirty] Clean

Dirty Dirty Clean Dirty

Dirty Dirty Dirty Dirty

Clean Clean Clean Clean   
----------------------------------------------------------------------  
Current location: (0,1)

Percept: The square is dirty

Action: Clean then move

Dirty [Dirty] Clean Clean

Dirty Dirty Clean Dirty

Dirty Dirty Dirty Dirty

Clean Clean Clean Clean   
----------------------------------------------------------------------  
Current location: (1,1)

Percept: The square is dirty

Action: Clean then move

Dirty Clean Clean Clean

Dirty [Dirty] Clean Dirty

Dirty Dirty Dirty Dirty

Clean Clean Clean Clean   
----------------------------------------------------------------------  
Current location: (1,0)

Percept: The square is dirty

Action: Clean then move

Dirty Clean Clean Clean

[Dirty] Clean Clean Dirty

Dirty Dirty Dirty Dirty

Clean Clean Clean Clean   
----------------------------------------------------------------------  
Current location: (0,0)

Percept: The square is dirty

Action: Clean then move

[Dirty] Clean Clean Clean

Clean Clean Clean Dirty

Dirty Dirty Dirty Dirty

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (0,1)

Percept: The square is clean

Action: Move

Clean [Clean] Clean Clean

Clean Clean Clean Dirty

Dirty Dirty Dirty Dirty

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (1,1)

Percept: The square is clean

Action: Move

Clean Clean Clean Clean

Clean [Clean] Clean Dirty

Dirty Dirty Dirty Dirty

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (1,2)

Percept: The square is clean

Action: Move

Clean Clean Clean Clean

Clean Clean [Clean] Dirty

Dirty Dirty Dirty Dirty

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (2,2)

Percept: The square is dirty

Action: Clean then move

Clean Clean Clean Clean

Clean Clean Clean Dirty

Dirty Dirty [Dirty] Dirty

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (3,2)

Percept: The square is clean

Action: Move

Clean Clean Clean Clean

Clean Clean Clean Dirty

Dirty Dirty Clean Dirty

Clean Clean [Clean] Clean

----------------------------------------------------------------------

Current location: (3,1)

Percept: The square is clean

Action: Move

Clean Clean Clean Clean

Clean Clean Clean Dirty

Dirty Dirty Clean Dirty

Clean [Clean] Clean Clean

----------------------------------------------------------------------

Current location: (2,1)

Percept: The square is dirty

Action: Clean then move

Clean Clean Clean Clean

Clean Clean Clean Dirty

Dirty [Dirty] Clean Dirty

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (2,2)

Percept: The square is clean

Action: Move

Clean Clean Clean Clean

Clean Clean Clean Dirty

Dirty Clean [Clean] Dirty

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (2,3)

Percept: The square is dirty

Action: Clean then move

Clean Clean Clean Clean

Clean Clean Clean Dirty

Dirty Clean Clean [Dirty]

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (1,3)

Percept: The square is dirty

Action: Clean then move

Clean Clean Clean Clean

Clean Clean Clean [Dirty]

Dirty Clean Clean Clean

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (0,3)

Percept: The square is clean

Action: Move

Clean Clean Clean [Clean]

Clean Clean Clean Clean

Dirty Clean Clean Clean

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (0,2)

Percept: The square is clean

Action: Move

Clean Clean [Clean] Clean

Clean Clean Clean Clean

Dirty Clean Clean Clean

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (0,1)

Percept: The square is clean

Action: Move

Clean [Clean] Clean Clean

Clean Clean Clean Clean

Dirty Clean Clean Clean

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (0,0)

Percept: The square is clean

Action: Move

[Clean] Clean Clean Clean

Clean Clean Clean Clean

Dirty Clean Clean Clean

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (1,0)

Percept: The square is clean

Action: Move

Clean Clean Clean Clean

[Clean] Clean Clean Clean

Dirty Clean Clean Clean

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (2,0)

Percept: The square is dirty

Action: Clean then move

Clean Clean Clean Clean

Clean Clean Clean Clean

[Dirty] Clean Clean Clean

Clean Clean Clean Clean

----------------------------------------------------------------------

Current location: (3,0)

Percept: The square is clean

Action: Move

Clean Clean Clean Clean

Clean Clean Clean Clean

Clean Clean Clean Clean

[Clean] Clean Clean Clean

----------------------------------------------------------------------

Current location: (3,1)

Percept: The square is clean

Action: Move

Clean Clean Clean Clean

Clean Clean Clean Clean

Clean Clean Clean Clean

Clean [Clean] Clean Clean

----------------------------------------------------------------------

Current location: (3,2)

Percept: The square is clean

Action: Move

Clean Clean Clean Clean

Clean Clean Clean Clean

Clean Clean Clean Clean

Clean Clean [Clean] Clean

----------------------------------------------------------------------

Current location: (3,3)

Percept: The square is clean

Action: Move

Clean Clean Clean Clean

Clean Clean Clean Clean

Clean Clean Clean Clean

Clean Clean Clean [Clean]

----------------------------------------------------------------------

Process finished with exit code 0