RITVIK CHAWLA
20930105G
SECTION: E
BATCH-1

Date: / /
Page No.

ARTIFICIAL INTELLIGENCE AND SOFT COMPUTING LAB

A-1- A problem which must be solved using an AI agent is "Rat in a Maze" Problem. ~~In~~
In this problem, a given maze of size N×N exists. The source and destination locations are top-left and bottom-right cells respectively. Some cells are valid to move and some cells are blocked. If one rat starts moving from the start vertex to the destination vertex, we have to find that is there any way to complete the path, if it is possible then mark the correct path for the rat.
The maze is given using a binary matrix, where it is marked with 1, it is a valid path, otherwise 0 for a blocked cell.

INITIAL STATE → The initial state of the rat in a maze problem is at whatever location the rat is placed (usually (0,0)) in a square matrix m[][] of order n.

FINAL STATE → The final state of the problem is to reach the destination at (n-1, n-1). The task is to find a sorted array of strings denoting all the possible directions that the rat can take to reach the destination at (n-1, n-1).

TRANSITION MODEL → The transition model consists of all the destination nodes the rat reaches through direction in order to reach its final destination. The ~~distinct~~ directions in which the rat can move are 'U'(up), 'D'(down), 'L'(left), 'R'(right).

PATH COST → It will turn out to be minimum when the shortest path to destination will be taken. Since here, heuristic values aren't taken, there is no definitive path cost.

RITVIK CHAWLA
2093010S6
SECTION: E
BATCH 1

Date: / /
Page No.

PEAS →

(1) PERFORMANCE MEASURE → This will be regarded as the minimum number of moves in which the rat can reach the final destination in the matrix (maze) from the initial state

(2) ENVIRONMENT → The environment will be the N×N matrix (maze) where the rat is supposed to move in various directional steps to reach a certain destination.

(3) ACTUATORS → The actuators for this problem will be the keyboard direction arrows on a laptop or our fingers for a touch phone where we can give directions to the rat so it gets into motion.

(4) SENSORS → As you type, the processor in the keyboard analyses the key matrix and determines what characters to send to the computer.

A.2 – SEARCH TECHNIQUE USED

I have used the Brute Force Approach as a search technique, where we can use simple backtracking approach without any extra space.

Here,

BACKTRACKING ALGORITHM → It is an algorithmic-technique for solving problems recursively by trying to build a solution incrementally. Solving one piece at a time, and removing those solutions that fail to satisfy the constraints of the problem at any point of time (by time, here, is referred to the time elapsed till reaching any level of the search tree) is the process of backtracking.

RITVIK CHABRA

209301056

SECTION: E

BATCH - 1

Date: / /

Page No.

A-3- To use a more efficient approach (to optimize the solution), we can modify the given matrix to treat is as a visited matrix.
(implementation showed in GitHub Project Link).

This approach, even though ~~it~~ makes the time complexity stay the same i.e. O(3^(N^2)),
~~it~~ it reduces the space complexity from O(3^(N^2)) to O(1), since we are not using visited matrix again.