|  |  |
| --- | --- |
| Shri Vile Parle Kelavani Mandal's  **INSTITUTE OF TECHNOLOGY**  **DHULE (M.S.)**  **DEPARMENT OF COMPUTER ENGINEERING** | |
| **Name:** Anushka Paras Jain **Roll No.:** 01  **Subject :** Artificial Intelligence Lab **Subject Code :** BTCOL707  **Class:** Final Year Comp. Engg. **Expt. No. :** 06  **Title :** Solve Robot (traversal) problem using means End Analysis. | |
| **Problem Statement:**  **Software Required:**  **Theory:**  **Conclusion:** | Solve Robot (traversal) problem using means End Analysis    Prolog  The traversal problem for a robot is one of the many challenges that can be solved using the Means-End Analysis (MEA), a popular approach in artificial intelligence. In this case, the robot must use all available tools or actions to go from its current position to a target place. We'll use a streamlined Prolog example to demonstrate the idea of MEA for a robot traversal problem.  Assume you have a straightforward grid environment with movable robot that can move left, right, up, and down, and that its objective is to go to a predetermined spot. This is a sample of Prolog code that illustrates a fundamental MEA solution for this issue:  % Define the initial state (robot's current position)  initial\_state(2, 2). % Assuming the robot starts at position (2, 2).  % Define the goal state (target location)  goal\_state(5, 5). % Assuming the goal is to reach position (5, 5).  % Define the means or actions (robot's movement commands)  move(up, 0, -1).  move(down, 0, 1).  move(left, -1, 0).  move(right, 1, 0).  % Define the MEA algorithm  mea(State, State, []).  mea(CurrentState, GoalState, [Action | Actions]) :-  move(Action, DX, DY),  NewX is CurrentStateX + DX,  NewY is CurrentStateY + DY,  mea((NewX, NewY), GoalState, Actions).  % Entry point to solve the traversal problem using MEA  solve\_traversal :-  initial\_state(CurrentStateX, CurrentStateY),  goal\_state(GoalStateX, GoalStateY),  mea((CurrentStateX, CurrentStateY), (GoalStateX, GoalStateY), Actions),  write('Solution Actions: '), nl,  print\_actions(Actions).  % Predicate to print the sequence of actions  print\_actions([]).  print\_actions([Action | Rest]) :-  write(Action), nl,  print\_actions(Rest).  % Start the solver  :- solve\_traversal.  In this example, we specify the robot's starting point, destination, and possible moves. The MEA algorithm is implemented by the mea predicate. It computes a series of actions (movements) to accomplish the objective given the current state (the robot's position) and the goal state (the target location).  The starting position, the desired state, and the possible moves can all be changed to fit your unique robot traversal scenario. This is a condensed example of how MEA can be used in Prolog to solve such an issue. The grid world may be more complicated and require consideration of extra constraints in real-life circumstances.  The Robot (traversal) problem was solved using means-ends analysis (MEA). MEA breaks down the problem into subproblems, identifies operators to reduce the differences between states, and repeats until the goal state is reached. This approach provides a systematic solution to the robot navigation problem. |