Name: Rushiraj Suwarnkar  
Roll no.: 381018  
PRN: 22310315  
———————————————————————————————

***Assignment 4: To Implement A\*Algorithm for an Application***———————————————————————————————

**Problem Statement:**  
Implement the A\* (A-star) search algorithm to solve a problem such as pathfinding in a grid, maze solving, or route planning. The algorithm should find the shortest path from the start node to the goal node using cost and heuristic functions.  
———————————————————————————————

**Objectives:**

* Understand the functioning of the A\* algorithm.
* Implement A\* to find optimal paths in a search space.
* Compare the efficiency of A\* with other search algorithms like BFS or DFS.  
  ———————————————————————————————

**Theory:**  
———————————————————————————————

**Methodology:**  
The A\* algorithm is a best-first search algorithm that finds the shortest path between nodes by combining the cost to reach a node (g(n)) and a heuristic estimate of the cost from that node to the goal (h(n)). The evaluation function is f(n) = g(n) + h(n). A\* efficiently guides the search using this function to explore promising paths first.  
———————————————————————————————

**Working Principle / Algorithm:**  
———————————————————————————————

*A Algorithm:*\*

1. Initialize the open list with the start node and the closed list as empty.
2. Repeat the following until the goal is reached or the open list is empty:
   * Choose the node n with the lowest f(n) from the open list.
   * If n is the goal, reconstruct the path and terminate.
   * Move n from the open list to the closed list.
   * For each neighbor of n:
     + If neighbor is in the closed list, skip.
     + Calculate g, h, and f values for the neighbor.
     + If neighbor is not in the open list, add it.
     + If neighbor is in the open list with higher f, update its values.  
       ———————————————————————————————

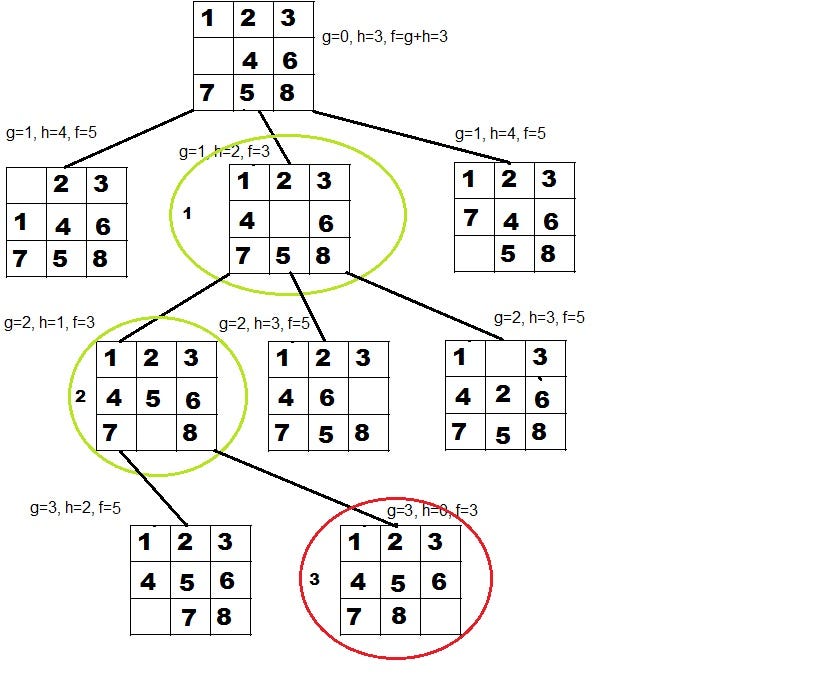
**Advantages:**

* Guarantees the shortest path if the heuristic is admissible.
* Efficiently reduces search space compared to uninformed searches.
* Can be applied to various pathfinding problems.  
  ———————————————————————————————

**Disadvantages / Limitations:**

* Requires memory to store all explored nodes.
* Performance depends on the quality of the heuristic.
* May be slow for very large search spaces.  
  ———————————————————————————————

**Diagram:**

  
———————————————————————————————

**Conclusion:**  
The A\* algorithm is a powerful and widely used search algorithm for pathfinding and optimization problems. By combining actual cost and heuristic estimates, it efficiently finds optimal paths while reducing unnecessary exploration.  
———————————————————————————————