

1. Depth-First Search (DFS) Algorithm

The DFS algorithm uses a stack('list') to store the list of tuples containing the states visited and the path to reach them.

	Tiny maze	Medium maze	Big maze
Search nodes expanded	15	146	390
Time taken (in sec)	0.0	0.0	0.1
Total cost	10	130	210

2. Breadth-First Search (BFS) Algorithm

The BFS algorithm uses a queue('list') to store the list of tuples containing the states visited and the path to reach them.

	Tiny maze	Medium maze	Big maze
Search nodes expanded	15	269	620
Time taken (in sec)	0.0	0.0	0.1
Total cost	8	68	210

3. Uniform-Cost Search (UCS) Algorithm

The UCS algorithm uses a priority queue('list') that stores the states visited, path and cost to reach those states.

	Medium maze (search agent)	Med dotted maze (stayEastSearchAgent)	Big maze
Search nodes expanded	269	186	620
Time taken (in sec)	0.0	0.0	0.0
Total cost	68	1	210

4. A* Algorithm

A* algorithm uses a null heuristic and priority queue('list') that stores the states visited, path and cost to reach those states.

	Tiny maze	Medium maze	Big maze
Search nodes expanded	14	221	549
Time taken (in sec)	0.0	0.0	0.0
Total cost	8	68	210

5. Search Agent for CornersProblem

init() - The no. of corners to be visited is set to 4 since no corners are visited initially.

getStartState() - The start state and the corner positions are passed here.

getSuccessors() - As a corner is reached, it is removed from the list of corners to be visited.

isGoalState() – Returns true when the list of corners to be visited is empty, all the corners have been visited.

6. Heuristic for CornersProblem

In the CornersProblem, the Manhattan heuristic calculates the distance between the current state and the corners and returns the least distance between the current state and the corresponding corner. This helps in choosing the successor state with the least cost.

	Tiny maze	Medium maze	Big maze
Search nodes expanded	14	221	549
Time taken (in sec)	0.0	0.0	0.0
Total cost	8	68	210

7. Heuristic for FoodSearchProblem

In the FoodSearchProblem, the heuristic calculates the distance between the current state and the food co-ordinates and returns the co-ordinates with least distance from the current state. This helps in choosing the food closest to the current state.

	TrickySearch
Search nodes expanded	9551
Time taken (in sec)	12.6
Total cost	60