Report (Assignment # 1)

Ali Haider 15f8096

Arbaz Sharif 15f8059

We have to implement the following searches on the given grid to find the goal

1. **Breadth first search:**

We use this with the help of queue data structure to implement it and find the goal with the time complexity of o(v^2) here v is the number of vertices.it takes the larger memory then the depth vise because it have to store the all nodes that are visited.

Optimality:

Optimal for finding the shortest distance, not in cost.

2. **Depth first search:**

We use this with the help of stack data structure to implement it and find the goal with the time complexity of O(V+E) here v is the number of vertices and e is number of edges .it takes the lesser memory then the breath vise because it doesn’t have to store the all nodes that are visited it just goes into depth of the graph that’s how it traverse the graph

Optimality: Not optimal

3. **Uniform Cost Search:**

Uniform Cost Search is the best algorithm so far for a search problem, It can solve any general graph for optimal cost. Uniform Cost Search as it sounds searches in branches which are more or less the same in cost. Uniform Cost Search again demands the use of a priority queue. It moves towards lesser cost .O(bc/m), where c is the cost of an optimal solution and m is the minimum edge cost.

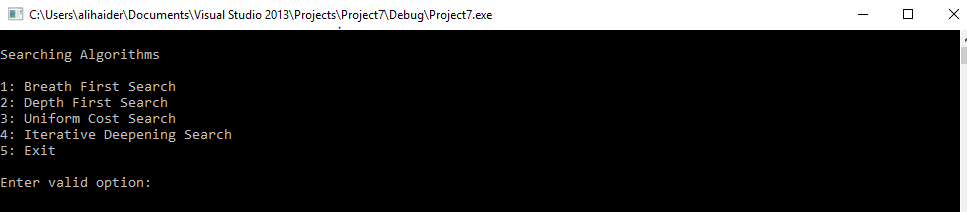
4. **Iterative deepening search:**

It performs multiple search phases, with each phase having a depth bound on the search. Thus, in terms of percentages, search time is not increased that much. The advantage of iterative deepening is that you can use a space-efficient search in each phase, such as Depth-First Search.

Best suited for a complete infinite tree

It uses a large number of space because it have to perform the both searches at same time.

**Instructions:**



When u run this console will appear u have to select a search which have to be perform after selecting it output will be shown