Mengjie Wang, Jun Zhou

September 16, 2018

CS420

Professor Tao

Project Report

	Depth-First Search			Breadth-First Search		
	#nodes explored	Solution length	Is it optimal?	#nodes explored	Solution length	Is it optimal?
tinyMaze	15	10	No	15	8	Yes
medium Maze	146	130	No	268	68	Yes
bigMaze	390	210	Yes	620	210	Yes

b)

1) The DFS will firstly explore the root and then explore a single path to the leaf thoroughly if the goal is not found then explore another path. For BFS, it will explore all the successors of the root and then try to explore the successors of the successors and continue on.
2) To find the solution, DFS will expand fewer nodes than BFS, which means it will be faster in execution and takes less time.
3) If there is only one path that can lead to the destination, as in the bigMaze, DFS can always find the optimal solution. In this case, DFS is better than BFS as it expands fewer nodes.
4) BFS requires more memory than DFS as it will have to remember every node it expands. However, DFS only have to remember only one path from the root.

5) BFS is always optimal while DFS is not

In the tinyMaze, there are two paths leads to the food, the DFS chose the one that is not optimal because it gets expanded first and BFS will always return the best solution. The node expand is 15 in both BFS and DFS as both of them expanded the whole maze.

In the mediumMaze, there are multiple paths to the food and BFS will always return the optimal one, but it needs to expand 268 of nodes to get the solution, which can take a long time and user plenty of memory. For DFS, although it does not return the optimal solution, it expanded fewer nodes and therefore faster than DFS.

In the bigMaze, there is only one path to the food. Since both BFS and DFS are complete which means they will find the solution if exists, they both will find this path to the food. In that case, DFS will have better performance as it is faster and take less memory than BFS.