CS 312: Artificial Intelligence Laboratory

Lab 1 Report

Online Group - 10

Rudraksh Sattabhayya – 190030038

Sumedhsing Rajput – 190030042

Introduction:

The Aim of this Lab to achieve the task is to simulate breadth-first search, depth-first search, and DFID in the state space. The state-space consists of an m x n grid. The start state is (0,0). The goal state is the position of (*) in the grid. The Pacman is allowed to move UP, DOWN, LEFT and RIGHT (except for boundary). A comparison of the path length and the number of states explored between the different search methods and, also, between the orders in which neighbours are added, are performed.

Directions to Run Code

Command: 1: g++ Lab1_OnlineGroup10.cpp -o showOutput.exe

2:showOutput.exe

This will print output on terminal

1. Pseudo Code

1.1 GoalTest(state)

Returns true if the input state is goal and false otherwise.

Algorithm
1: **procedure** goalTest(state)
2: **if** state.value == “∗ “ **then**
3: **return** true
4: **return** false (=> state is not goal)

1.2 MoveGen(state)

The function takes a state as input and returns a set of states that are reachable from the input state in one step or basically, it returns neighbours of the state.

Algorithm
1: **procedure** moveGen(state)
2: *nextStates* ← () (=>initialize nextStates to empty set)
3: **for** neighbour n of *state* in order(DOWN,UP,RIGHT,LEFT) **do**
4: **if** n is not boundary **then**
5: *nextStates.append(n)*
6: **return** *nextStates* (nextStates are required moves generated)

2. Results

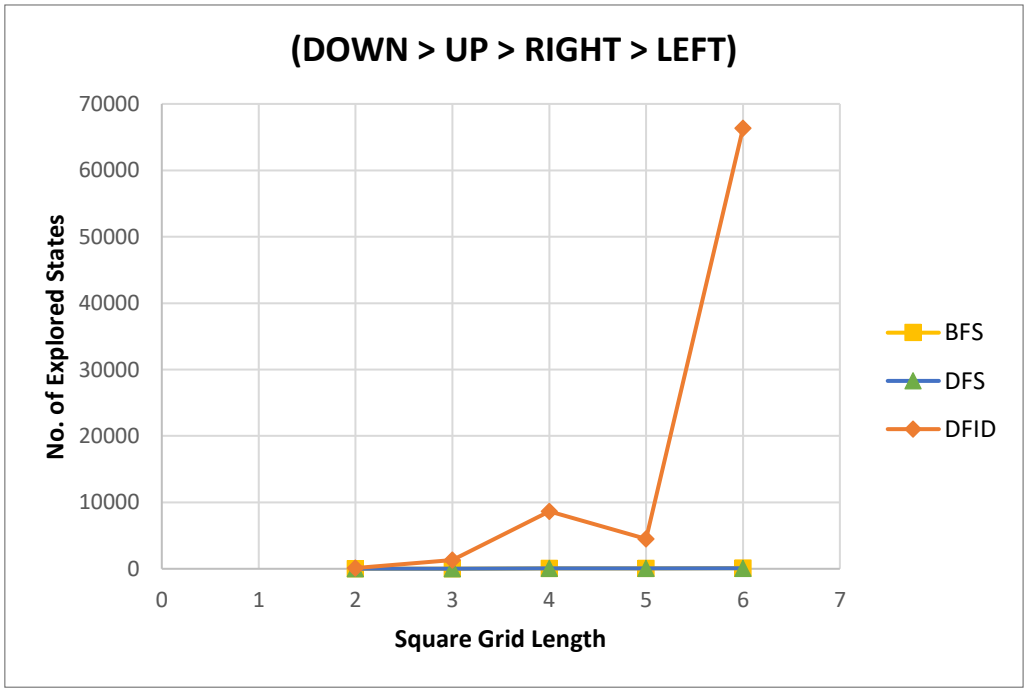
(*All the mazes used for making this report are in mazes.txt)

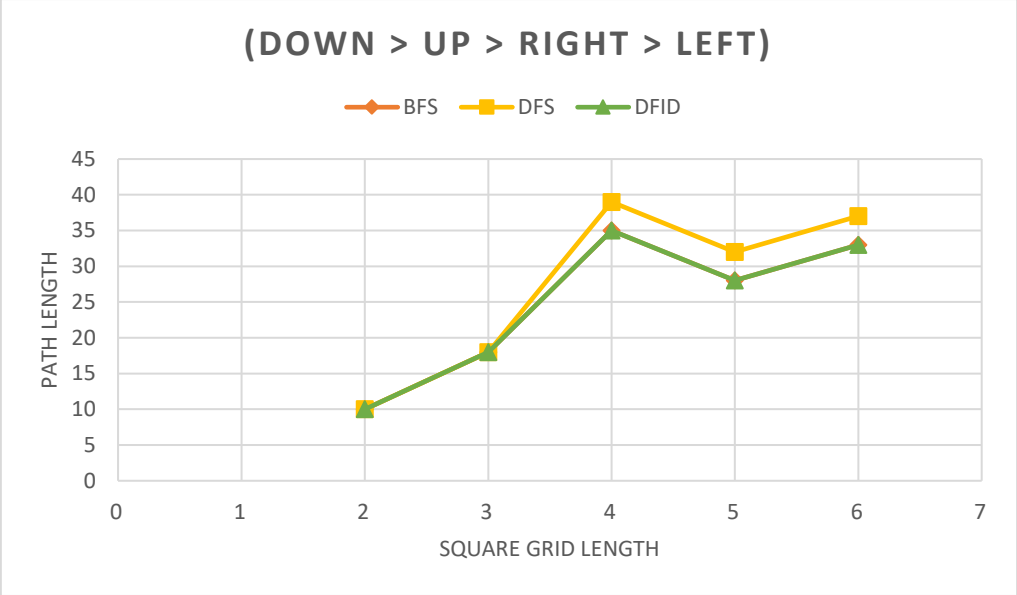
2.1 Tables

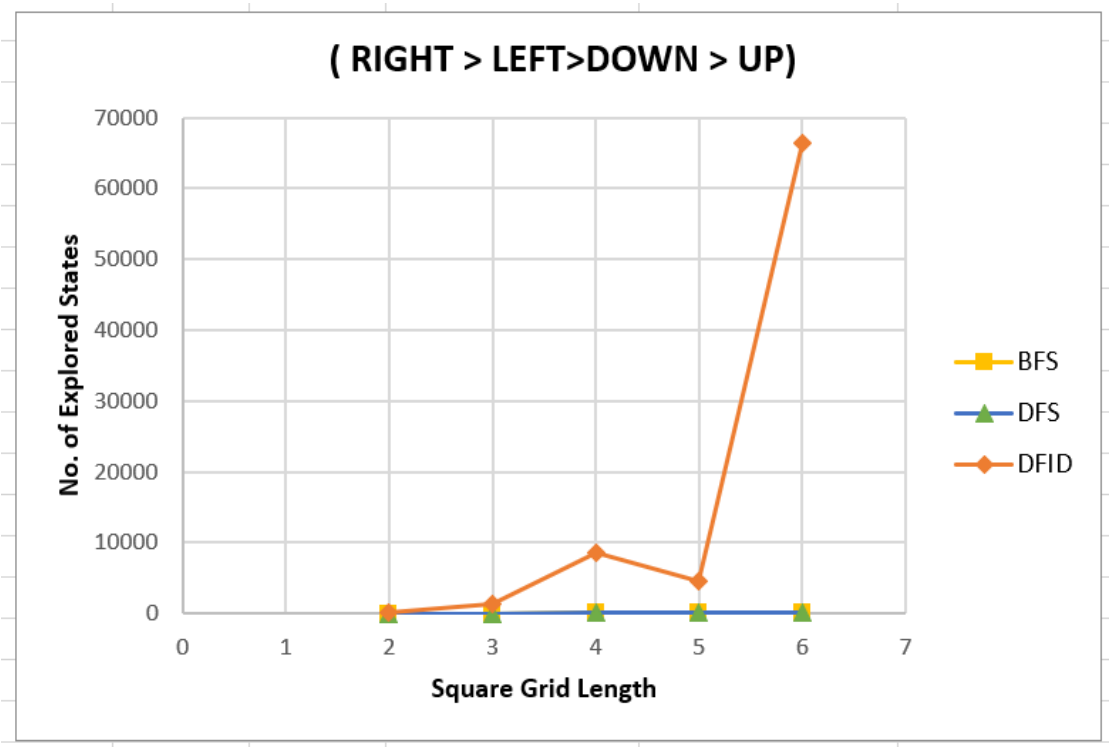
Algorithm	Statistics (DOWN > UP > RIGHT > LEFT)			
	No. horizontal cells	No. horizontal cells	No. states explored	Path length
BFS	2	2	14	10
DFS			14	10
DFID			110	10
BFS	3	3	26	18
DFS			22	18
DFID			1309	18
BFS	4	4	51	35
DFS			44	39
DFID			8624	35
BFS	5	5	60	28
DFS			41	32
DFID			4483	28
BFS	6	6	97	33
DFS			78	37
DFID			66346	33

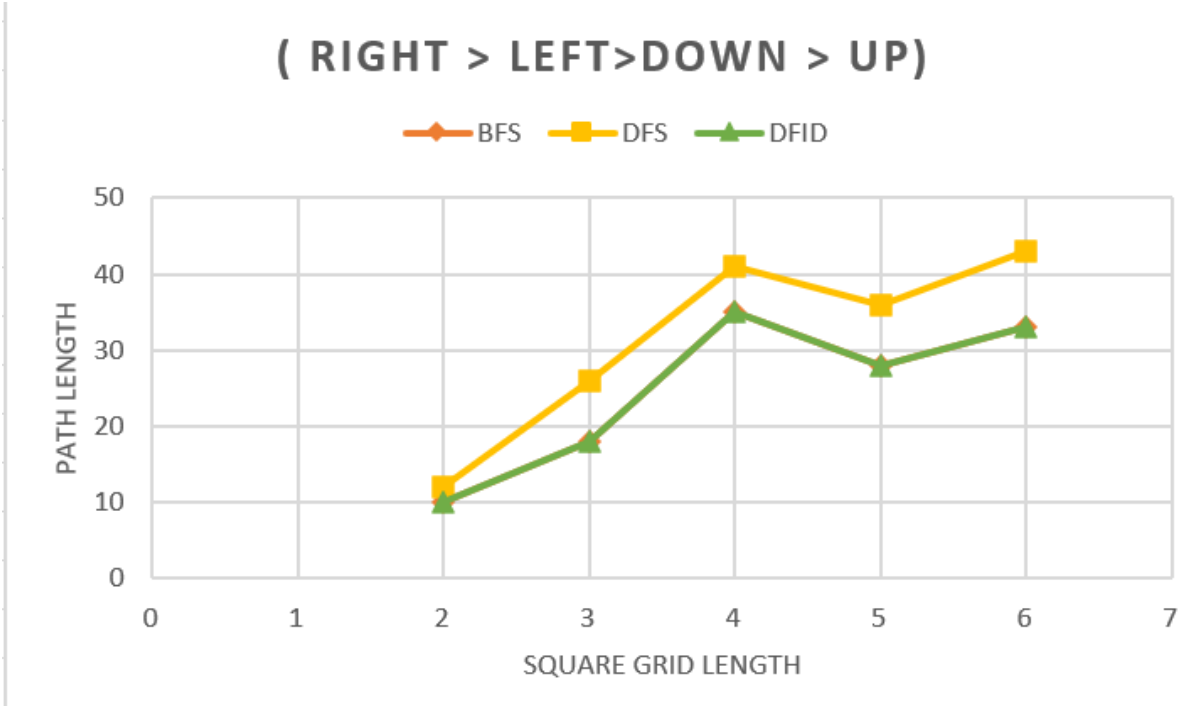
Algorithm	Statistics (RIGHT > LEFT>DOWN > UP)			
	No. horizontal cells	No. horizontal cells	No. states explored	Path length
BFS	2	2	14	10
DFS			14	12
DFID			82	10
BFS	3	3	26	18
DFS			26	26
DFID			1394	18
BFS	4	4	50	35
DFS			42	41
DFID			10178	35
BFS	5	5	62	28
DFS			72	36
DFID			4533	28
BFS	6	6	95	33
DFS			76	43
DFID			75107	33

2.2 Plots




(DOWN > UP > RIGHT > LEFT)
BFS
DFS
DFID
PATH LENGTH
45
40
35
30
25
20
15
10
5
0
0
1
2
3
4
5
6
7
SQUARE GRID LENGTH


(RIGHT > LEFT>DOWN > UP)
No. of Explored States
70000
60000
50000
40000
30000
20000
10000
0
0
1
2
3
4
5
6
7
Square Grid Length
BFS
DFS
DFID



3. Conclusion

The results of the dependence of the path length and number of states explored, as seen in the previous section, are summarized in the table below. For small inputs in DFID, we observe that the increase in the number of explored states is due to the small branching factor and high constant attached with the time complexity.

Algorithm	***Dependence on order of neighbours added***	
	No. States Explored	***Path Length***
BFS	*True*	*False*
DFS	*True*	*True*
DFID	*True*	*False*