

my):1  
ty):0

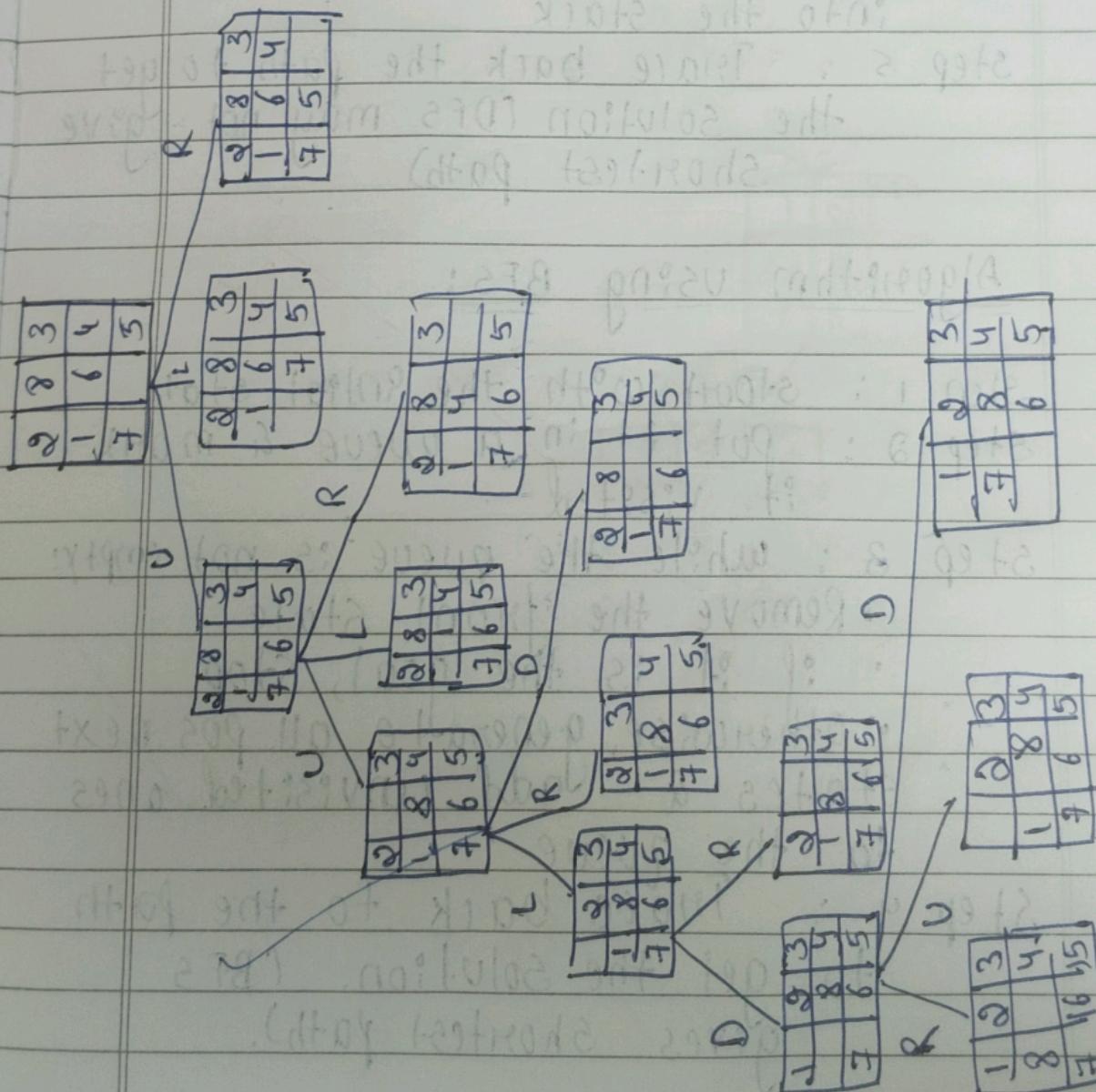
Using BFS solve 8 puzzle without heuristic

2	8	3	1	2	3
1	6	4	8	4	
7	5		7	6	5

initial      goal

direction:

up, up, left, down, Right



Algorithm using DFS:

Step 1 : start with the initial state of the puzzle

Step 2 : check if the puzzle is solvable by counting inversions. if not stop

Step 3 : push it onto a stack & mark it visited

Step 4 : while the stack is not empty:

- pop the top state

- if it is the goal, stop

- otherwise, generate all pos next states & push unvisited ones into the stack

Step 5 : Trace back the path to get the solution (DFS may not give shortest path)

Algorithm using BFS:

Step 1 : start with the initial state

Step 2 : put it in a queue & mark it visited

Step 3 : while the queue is not empty:

- Remove the front state

- if it is the goal, stop.

- otherwise, generate all pos next states & add unvisited ones to the queue

Step 4 : Trace back to the path to get the solution. (BFS gives shortest path).

DFS (b)

2	8	3
1	6	4
7		5

↓ down

9	8	3
1		4
7	6	5

↓ down

2	8	3
1	8	4
7	6	5

right

	2	3
1	8	4
7	6	5

up

1	2	3
	8	4
7	6	5

left

1	2	3
8		4
7	6	5

Output for BFS

Output for DFS

solution found in 5 moves: soln found in 5 moves.

2	8	3
1	6	4
7	0	5

2	8	3
1	0	4
7	6	5

2	0	3
1	8	4
7	6	5

0	2	3
1	8	4
7	6	5

1	2	3
0	8	4
7	6	5

1	2	3
8	0	4
7	6	5

goal

2	8	3
1	6	4
7	0	5

2	8	3
1	0	4
7	6	5

2	0	3
1	8	4
7	6	5

0	2	3
1	8	4
7	6	5

1	2	3
0	8	4
7	6	5

1	2	3
8	0	4
7	6	5

goal.

Solution found in 5 moves:

(2, 8, 3)

(1, 6, 4)

(7, 0, 5)

(2, 8, 3)

(1, 0, 4)

(7, 6, 5)

(2, 0, 3)

(1, 8, 4)

(7, 6, 5)

(0, 2, 3)

(1, 8, 4)

(7, 6, 5)

(1, 2, 3)

(0, 8, 4)

(7, 6, 5)

(1, 2, 3)

(8, 0, 4)

(7, 6, 5)

Srushti N  
1BM24CS424

Solution found in 5 moves:

(2, 8, 3)

(1, 6, 4)

(7, 0, 5)

(2, 8, 3)

(1, 0, 4)

(7, 6, 5)

(2, 0, 3)

(1, 8, 4)

(7, 6, 5)

(0, 2, 3)

(1, 8, 4)

(7, 6, 5)

(1, 2, 3)

(0, 8, 4)

(7, 6, 5)

(1, 2, 3)

(8, 0, 4)

(7, 6, 5)

Srushti N  
1BM24CS424

DFS

5 moves

initial

## Iterative Deepening Search (IDS) or Iterative -ve Deepening Depth First Search (IDDFS)

1	0	3
4	0	5
6	7	8

initial

1	3	3
4	5	0
6	7	8

goal

sol'n found in depth 1

Right →

1	0	∞
2	4	+
-	5	0

Left →

3	6	∞
2	5	+
-	0	0

Algorithm :

Step 1 : Set depth limit = 0

Step 2 : Perform Depth-limited DFS from the initial state up to the current depth limit

Step 3 : If the goal is found during the path DLS, stop & return the

Step 4 : If not found, increase the depth limit by 1

Step 5 : Repeat steps

2-4 until the goal is found or max depth is reached.

down →

3	6	∞
2	5	+
-	0	0

up →

3	6	∞
0	2	+
-	5	0

Limit

Output:

searching with depth limit = 0

searching with depth limit = 1

Solution found in 1 moves.

Step 0:

1 2 3

4 0 5

6 7 8

Step 1:

1 2 3

4 5 0 6 7 8

6 7 8

```
Searching with depth limit = 0  
Searching with depth limit = 1  
Solution found in 1 moves!
```

Step 0:

```
1 2 3  
4 0 5  
6 7 8
```

Step 1:

```
1 2 3  
4 5 0  
6 7 8
```

Srushti N  
1BM24CS424