Farmer_Fox

Welcome to ItrBFS Initial State: LEFT: boat & farmer fox chicken grain RIGHT: len(OPEN)=1; len(CLOSED)=0; COUNT = 0 OPEN is now: LEFT: fox grain RIGHT: boat & farmer chicken len(OPEN)=1; len(CLOSED)=1; COUNT = 1 OPEN is now: LEFT: fox grain boat & farmer RIGHT: chicken len(OPEN)=1; len(CLOSED)=2; COUNT = 2 OPEN is now: LEFT: grain RIGHT: chicken boat & farmer fox , LEFT: fox RIGHT: chicken boat & farmer grain len(OPEN)=2; len(CLOSED)=3; COUNT = 3 OPEN is now: LEFT: fox RIGHT: chicken boat & farmer grain , LEFT: grain boat & farmer chicken RIGHT: fox len(OPEN)=2; len(CLOSED)=4; COUNT = 4 OPEN is now: LEFT: grain boat & farmer chicken RIGHT: fox , LEFT: fox boat & farmer chicken RIGHT: grain len(OPEN)=2; len(CLOSED)=5; COUNT = 5 OPEN is now: LEFT: fox boat & farmer chicken RIGHT: grain , LEFT: chicken RIGHT: fox boat & farmer grain len(OPEN)=2; len(CLOSED)=6; COUNT = 6 OPEN is now: LEFT: chicken RIGHT: fox boat & farmer grain len(OPEN)=1; len(CLOSED)=7; COUNT = 7 OPEN is now: LEFT: chicken boat & farmer RIGHT: fox grain len(OPEN)=1; len(CLOSED)=8; COUNT = 8 OPEN is now: LEFT: RIGHT: fox grain boat & farmer chicken len(OPEN)=1; len(CLOSED)=9; COUNT = 9 Congratulations on successfully helping the	Welcome to ItrDFS Initial State: LEFT: boat & farmer fox chicken grain RIGHT: len(OPEN)=1; len(CLOSED)=0; COUNT = 0 OPEN is now: LEFT: fox grain RIGHT: boat & farmer chicken len(OPEN)=1; len(CLOSED)=1; COUNT = 1 OPEN is now: LEFT: fox grain boat & farmer RIGHT: chicken len(OPEN)=1; len(CLOSED)=2; COUNT = 2 OPEN is now: LEFT: grain RIGHT: chicken boat & farmer fox , LEFT: fox RIGHT: chicken boat & farmer grain len(OPEN)=2; len(CLOSED)=3; COUNT = 3 OPEN is now: LEFT: grain boat & farmer chicken RIGHT: fox , LEFT: fox RIGHT: chicken boat & farmer grain len(OPEN)=2; len(CLOSED)=4; COUNT = 4 OPEN is now: LEFT: chicken RIGHT: fox boat & farmer grain , LEFT: fox RIGHT: chicken boat & farmer grain len(OPEN)=2; len(CLOSED)=5; COUNT = 5 OPEN is now: LEFT: chicken boat & farmer RIGHT: fox grain , LEFT: chicken boat & farmer fox RIGHT: grain , LEFT: fox RIGHT: chicken boat & farmer grain len(OPEN)=3; len(CLOSED)=6; COUNT = 6 OPEN is now: LEFT: RIGHT: fox grain boat & farmer chicken , LEFT: chicken boat & farmer fox RIGHT: grain , LEFT: fox RIGHT: chicken boat & farmer grain len(OPEN)=3; len(CLOSED)=7; COUNT = 7 Congratulations on successfully helping the farmer with the task Solution path: LEFT: boat & farmer fox chicken grain

farmer with the task Solution path: LEFT: boat & farmer fox chicken grain RIGHT: LEFT: fox grain RIGHT: boat & farmer chicken LEFT: fox grain boat & farmer RIGHT: chicken LEFT: grain RIGHT: chicken boat & farmer fox LEFT: grain boat & farmer chicken RIGHT: fox LEFT: chicken RIGHT: fox boat & farmer grain LEFT: chicken boat & farmer RIGHT: fox grain LEFT: RIGHT: fox grain boat & farmer chicken Length of solution path found: 7 edges 9 states expanded. MAX_OPEN_LENGTH = 2	RIGHT: LEFT: fox grain RIGHT: boat & farmer chicken LEFT: fox grain boat & farmer RIGHT: chicken LEFT: grain RIGHT: chicken boat & farmer fox LEFT: grain boat & farmer chicken RIGHT: fox LEFT: chicken RIGHT: fox boat & farmer grain LEFT: chicken boat & farmer RIGHT: fox grain LEFT: RIGHT: fox grain boat & farmer chicken Length of solution path found: 7 edges 7 states expanded. MAX_OPEN_LENGTH = 3
Length of Path(BFS):7	Length of Path(DFS):7
Nodes expanded(BFS):9	Nodes expanded(DFS):7

Missionaries

Welcome to ItrBFS Initial State: M on left:3 C on left:3 M on right:0 C on right:0 boat is on the left. len(OPEN)=1; len(CLOSED)=0; COUNT = 0 OPEN is now: M on left:0 C on left:3 M on right:3 C on right:0 boat is on the right. , M on left:2 C on left:2 M on right:1 C on right:1 boat is on the right. len(OPEN)=2; len(CLOSED)=1; COUNT = 1 OPEN is now: M on left:2 C on left:2 M on right:1 C on right:1 boat is on the right. len(OPEN)=1; len(CLOSED)=2; COUNT = 2 OPEN is now: M on left:3 C on left:2 M on right:0 C on right:1 boat is on the left. len(OPEN)=1; len(CLOSED)=3; COUNT = 3	Welcome to ItrDFS Initial State: M on left:3 C on left:3 M on right:0 C on right:0 boat is on the left. len(OPEN)=1; len(CLOSED)=0; COUNT = 0 OPEN is now: M on left:0 C on left:3 M on right:3 C on right:0 boat is on the right. , M on left:2 C on left:2 M on right:1 C on right:1 boat is on the right. len(OPEN)=2; len(CLOSED)=1; COUNT = 1 OPEN is now: M on left:2 C on left:2 M on right:1 C on right:1 boat is on the right. len(OPEN)=1; len(CLOSED)=2; COUNT = 2 OPEN is now: M on left:3 C on left:2 M on right:0 C on right:1 boat is on the left. len(OPEN)=1; len(CLOSED)=3; COUNT = 3

OPEN is now: M on left:0 C on left:2 M on right:3 C on right:1 boat is on the right. , M on left:1 C on left:1 M on right:2 C on right:2 boat is on the right. len(OPEN)=2; len(CLOSED)=4; COUNT = 4 OPEN is now: M on left:1 C on left:1 M on right:2 C on right:2 boat is on the right. , M on left:2 C on left:2 M on right:1 C on right:1 boat is on the left. len(OPEN)=2; len(CLOSED)=5; COUNT = 5 OPEN is now: M on left:2 C on left:2 M on right:1 C on right:1 boat is on the left. , M on left:3 C on left:1 M on right:0 C on right:2 boat is on the left. len(OPEN)=2; len(CLOSED)=6; COUNT = 6 OPEN is now: M on left:3	OPEN is now: M on left:0 C on left:2 M on right:3 C on right:1 boat is on the right. , M on left:1 C on left:1 M on right:2 C on right:2 boat is on the right. len(OPEN)=2; len(CLOSED)=4; COUNT = 4 OPEN is now: M on left:2 C on left:2 M on right:1 C on right:1 boat is on the left. , M on left:1 C on left:1 M on right:2 C on right:2 boat is on the right. len(OPEN)=2; len(CLOSED)=5; COUNT = 5 OPEN is now: M on left:1 C on left:1 M on right:2 C on right:2 boat is on the right. , M on left:0 C on left:1 M on right:3 C on right:2 boat is on the right. len(OPEN)=2; len(CLOSED)=6; COUNT = 6 OPEN is now: M on left:3

C on left:1 M on right:0 C on right:2 boat is on the left. ,	C on left:1 M on right:0 C on right:2 boat is on the left. ,
M on left:0 C on left:1 M on right:3 C on right:2 boat is on the right.	M on left:0 C on left:1 M on right:3 C on right:2 boat is on the right.
len(OPEN)=2; len(CLOSED)=7; COUNT = 7 OPEN is now: M on left:0 C on left:1 M on right:3 C on right:2 boat is on the right.	len(OPEN)=2; len(CLOSED)=7; COUNT = 7 OPEN is now: M on left:0 C on left:1 M on right:3 C on right:2 boat is on the right.
len(OPEN)=1; len(CLOSED)=8; COUNT = 8 OPEN is now: M on left:1 C on left:1 M on right:2 C on right:2 boat is on the left.	len(OPEN)=1; len(CLOSED)=8; COUNT = 8 OPEN is now: M on left:1 C on left:1 M on right:2 C on right:2 boat is on the left.
len(OPEN)=1; len(CLOSED)=9; COUNT = 9 OPEN is now: M on left:0 C on left:0 M on right:3 C on right:3 boat is on the right.	len(OPEN)=1; len(CLOSED)=9; COUNT = 9 OPEN is now: M on left:0 C on left:0 M on right:3 C on right:3 boat is on the right.
len(OPEN)=1; len(CLOSED)=10; COUNT = 10 Congratulations on successfully guiding the missionaries and cannibals across the river! Solution path: M on left:3 C on left:3 M on right:0	len(OPEN)=1; len(CLOSED)=10; COUNT = 10 Congratulations on successfully guiding the missionaries and cannibals across the river! Solution path: M on left:3 C on left:3 M on right:0

C on right:0 boat is on the left. M on left:2 C on left:2 M on right:1 C on right:1 boat is on the right. M on left:3 C on left:2 M on right:0 C on right:1 boat is on the left. M on left:0 C on left:2 M on right:3 C on right:1 boat is on the right. M on left:2 C on left:2 M on right:1 C on right:1 boat is on the left. M on left:0 C on left:1 M on right:3 C on right:2 boat is on the right. M on left:1 C on left:1 M on right:2 C on right:2 boat is on the left.	C on right:0 boat is on the left. M on left:2 C on left:2 M on right:1 C on right:1 boat is on the right. M on left:3 C on left:2 M on right:0 C on right:1 boat is on the left. M on left:0 C on left:2 M on right:3 C on right:1 boat is on the right. M on left:2 C on left:2 M on right:1 C on right:1 boat is on the left. M on left:1 C on left:1 M on right:2 C on right:2 boat is on the right. M on left:3 C on left:1 M on right:0 C on right:2 boat is on the left.

M on left:0 C on left:0 M on right:3 C on right:3 boat is on the right. Length of solution path found: 7 edges 10 states expanded. MAX_OPEN_LENGTH = 2	M on left:0 C on left:1 M on right:3 C on right:2 boat is on the right. M on left:1 C on left:1 M on right:2 C on right:2 boat is on the left. M on left:0 C on left:0 M on right:3 C on right:3 boat is on the right. Length of solution path found: 9 edges 10 states expanded. MAX_OPEN_LENGTH = 2
Length of Path(BFS): 7	Length of Path(DFS): 9
Nodes expanded(BFS): 10	Nodes expanded(DFS): 10

TowersOfHanoi

Welcome to ItrBFS Initial State: [[4, 3, 2, 1] ,[] ,[]] len(OPEN)=1; len(CLOSED)=0; COUNT = 0 OPEN is now: [[4, 3, 2] ,[1] ,[]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=2; len(CLOSED)=1; COUNT = 1 OPEN is now: [[4, 3, 2] ,[] ,[1]], [[4, 3] ,[1] ,[2]] len(OPEN)=2; len(CLOSED)=2; COUNT = 2 OPEN is now: [[4, 3] ,[1] ,[2]], [[4, 3] ,[2] ,[1]] len(OPEN)=2; len(CLOSED)=3; COUNT = 3 OPEN is now: [[4, 3] ,[2] ,[1]], [[4, 3, 1] ,[] ,[2]], [[4, 3] ,[] ,[2, 1]] len(OPEN)=3; len(CLOSED)=4; COUNT = 4 OPEN is now: [[4, 3, 1] ,[] ,[2]], [[4, 3] ,[] ,[2, 1]], [[4, 3, 1] ,[2] ,[]], [[4, 3] ,[2, 1] ,[]] len(OPEN)=4; len(CLOSED)=5; COUNT = 5 OPEN is now: [[4, 3] ,[] ,[2, 1]], [[4, 3, 1] ,[2] ,[]], [[4, 3] ,[2, 1] ,[]], [[4, 3, 1] ,[2] ,[]] len(OPEN)=4; len(CLOSED)=6; COUNT = 6 OPEN is now: [[4, 3, 1] ,[2] ,[]], [[4, 3] ,[2, 1] ,[]], [[4, 3, 1] ,[2] ,[]], [[4] ,[3] ,[2, 1]] len(OPEN)=4; len(CLOSED)=7; COUNT = 7 OPEN is now: [[4, 3] ,[2, 1] ,[]], [[4, 3, 1] ,[2] ,[]], [[4] ,[3] ,[2, 1]] len(OPEN)=3; len(CLOSED)=8; COUNT = 8 OPEN is now: [[4, 3, 1] ,[2] ,[]], [[4] ,[3] ,[2, 1]], [[4] ,[2, 1] ,[3]] len(OPEN)=3; len(CLOSED)=9; COUNT = 9 OPEN is now: [[4] ,[3] ,[2, 1]], [[4] ,[2, 1] ,[3]] len(OPEN)=2; len(CLOSED)=10; COUNT = 10 OPEN is now: [[4] ,[2, 1] ,[3]], [[4, 1] ,[3] ,[2]], [[4] ,[3, 1] ,[2]] len(OPEN)=3; len(CLOSED)=11; COUNT = 11 OPEN is now: [[4, 1] ,[3] ,[2]], [[4] ,[3, 1] ,[2]], [[4, 1] ,[2] ,[3]], [[4] ,[2] ,[3, 1]] len(OPEN)=4; len(CLOSED)=12; COUNT = 12 OPEN is now: [[4] ,[3, 1] ,[2]], [[4, 1] ,[2] ,[3]], [[4] ,[2] ,[3, 1]], [[4, 1] ,[3, 2] ,[]]	Welcome to ItrDFS Initial State: [[4, 3, 2, 1] ,[] ,[]] len(OPEN)=1; len(CLOSED)=0; COUNT = 0 OPEN is now: [[4, 3, 2] ,[1] ,[]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=2; len(CLOSED)=1; COUNT = 1 OPEN is now: [[4, 3] ,[1] ,[2]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=2; len(CLOSED)=2; COUNT = 2 OPEN is now: [[4, 3, 1] ,[] ,[2]], [[4, 3] ,[] ,[2, 1]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=3; len(CLOSED)=3; COUNT = 3 OPEN is now: [[4, 3] ,[] ,[2, 1]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=3; len(CLOSED)=4; COUNT = 4 OPEN is now: [[4] ,[3] ,[2, 1]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=3; len(CLOSED)=5; COUNT = 5 OPEN is now: [[4, 1] ,[3] ,[2]], [[4] ,[3, 1] ,[2]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=4; len(CLOSED)=6; COUNT = 6 OPEN is now: [[4] ,[3, 1] ,[2]], [[4, 1] ,[3, 2] ,[]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=4; len(CLOSED)=7; COUNT = 7 OPEN is now: [[4, 2] ,[3, 1] ,[]], [[4, 1] ,[3, 2] ,[]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=4; len(CLOSED)=8; COUNT = 8 OPEN is now: [[4, 2, 1] ,[3] ,[]], [[4, 2] ,[3] ,[1]], [[4, 1] ,[3, 2] ,[]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=5; len(CLOSED)=9; COUNT = 9 OPEN is now: [[4, 2] ,[3] ,[1]], [[4, 2, 1] ,[] ,[3]], [[4, 1] ,[3, 2] ,[]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=5; len(CLOSED)=10; COUNT = 10 OPEN is now: [[4] ,[3, 2] ,[1]], [[4, 2, 1] ,[] ,[3]], [[4, 1] ,[3, 2] ,[]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]] len(OPEN)=5; len(CLOSED)=11; COUNT = 11 OPEN is now: [[4, 1] ,[3, 2] ,[]], [[4] ,[3, 2,

len(OPEN)=4; len(CLOSED)=13; COUNT = 13
OPEN is now: [[4, 1] ,[2] ,[3]], [[4] ,[2] ,[3, 1]], [[4, 1] ,[3, 2] ,[]], [[4, 2] ,[3, 1] ,[]]
len(OPEN)=4; len(CLOSED)=14; COUNT = 14
OPEN is now: [[4] ,[2] ,[3, 1]], [[4, 1] ,[3, 2] ,[]], [[4, 2] ,[3, 1] ,[]], [[4, 1] ,[] ,[3, 2]]
len(OPEN)=4; len(CLOSED)=15; COUNT = 15
OPEN is now: [[4, 1] ,[3, 2] ,[]], [[4, 2] ,[3, 1] ,[]], [[4, 1] ,[] ,[3, 2]], [[4, 2] ,[] ,[3, 1]]
len(OPEN)=4; len(CLOSED)=16; COUNT = 16
OPEN is now: [[4, 2] ,[3, 1] ,[]], [[4, 1] ,[] ,[3, 2]], [[4, 2] ,[] ,[3, 1]], [[4] ,[3, 2, 1] ,[]], [[4] ,[3, 2] ,[1]]
len(OPEN)=5; len(CLOSED)=17; COUNT = 17
OPEN is now: [[4, 1] ,[] ,[3, 2]], [[4, 2] ,[] ,[3, 1]], [[4] ,[3, 2, 1] ,[]], [[4] ,[3, 2] ,[1]], [[4, 2, 1] ,[3] ,[]], [[4, 2] ,[3] ,[1]], [[4] ,[1] ,[3, 2]], [[4] ,[]
,[3, 2, 1]]
len(OPEN)=7; len(CLOSED)=19; COUNT = 19
OPEN is now: [[4] ,[3, 2, 1] ,[]], [[4] ,[3, 2] ,[1]], [[4, 2, 1] ,[3] ,[]], [[4, 2] ,[3] ,[1]], [[4] ,[1] ,[3, 2]], [[4] ,[] ,[3, 2, 1]], [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]]
len(OPEN)=8; len(CLOSED)=20; COUNT = 20
OPEN is now: [[4] ,[3, 2] ,[1]], [[4, 2, 1] ,[3] ,[]], [[4, 2] ,[3] ,[1]], [[4] ,[1] ,[3, 2]], [[4] ,[] ,[3, 2, 1]], [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]], [[] ,[3, 2, 1] ,[4]]
len(OPEN)=8; len(CLOSED)=21; COUNT = 21
OPEN is now: [[4, 2, 1] ,[3] ,[]], [[4, 2] ,[3] ,[1]], [[4] ,[1] ,[3, 2]], [[4] ,[] ,[3, 2, 1]], [[4, 2, 1] ,[]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=5; len(CLOSED)=12; COUNT = 12
OPEN is now: [[4] ,[3, 2, 1] ,[]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=4; len(CLOSED)=13; COUNT = 13
OPEN is now: [[] ,[3, 2, 1] ,[4]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=4; len(CLOSED)=14; COUNT = 14
OPEN is now: [[1] ,[3, 2] ,[4]], [[] ,[3, 2] ,[4, 1]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=5; len(CLOSED)=15; COUNT = 15
OPEN is now: [[] ,[3, 2] ,[4, 1]], [[1] ,[3] ,[4, 2]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=5; len(CLOSED)=16; COUNT = 16
OPEN is now: [[2] ,[3] ,[4, 1]], [[1] ,[3] ,[4, 2]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=5; len(CLOSED)=17; COUNT = 17
OPEN is now: [[2, 1] ,[3] ,[4]], [[2] ,[3, 1] ,[4]], [[1] ,[3] ,[4, 2]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=18; COUNT = 18
OPEN is now: [[2] ,[3, 1] ,[4]], [[2, 1] ,[] ,[4, 3]], [[1] ,[3] ,[4, 2]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=19; COUNT = 19
OPEN is now: [[] ,[3, 1] ,[4, 2]], [[2, 1] ,[] ,[4, 3]], [[1] ,[3] ,[4, 2]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=20; COUNT = 20
OPEN is now: [[1] ,[3] ,[4, 2]], [[] ,[3] ,[4, 2, 1]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3,

1] ,[] ,[3]], [[4, 2] ,[1] ,[3]], [[] ,[3,
2, 1] ,[4]]
len(OPEN)=7; len(CLOSED)=22; COUNT = 22
OPEN is now: [[4, 2] ,[3] ,[1]], [[4] ,[1] ,[3, 2]], [[4] ,[] ,[3, 2, 1]], [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]], [[] ,[3, 2, 1] ,[4]], [[4, 2, 1] ,[] ,[3]]
len(OPEN)=7; len(CLOSED)=23; COUNT = 23
OPEN is now: [[4] ,[1] ,[3, 2]], [[4] ,[] ,[3, 2, 1]], [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]], [[] ,[3, 2, 1] ,[4]], [[4, 2, 1] ,[] ,[3]]
len(OPEN)=6; len(CLOSED)=24; COUNT = 24
OPEN is now: [[4] ,[] ,[3, 2, 1]], [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]], [[] ,[3, 2, 1] ,[4]], [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]]
len(OPEN)=6; len(CLOSED)=25; COUNT = 25
OPEN is now: [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]], [[] ,[3, 2, 1] ,[4]], [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]], [[] ,[4] ,[3, 2, 1]]
len(OPEN)=6; len(CLOSED)=26; COUNT = 26
OPEN is now: [[4, 2] ,[1] ,[3]], [[] ,[3, 2, 1] ,[4]], [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]], [[] ,[4] ,[3, 2, 1]]
len(OPEN)=5; len(CLOSED)=27; COUNT = 27
OPEN is now: [[] ,[3, 2, 1] ,[4]], [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]], [[] ,[4] ,[3, 2, 1]]
len(OPEN)=4; len(CLOSED)=28; COUNT = 28
OPEN is now: [[4, 2, 1] ,[] ,[3]], [[4, 2] ,[1] ,[3]], [[] ,[4] ,[3, 2, 1]], [[1] ,[3, 2] ,[4]], [[] ,[3, 2] ,[4, 1]]
len(OPEN)=5; len(CLOSED)=29; COUNT = 29
OPEN is now: [[4, 2] ,[1] ,[3]], [[] ,[4] ,[3, 2, 1]], [[1] ,[3, 2] ,[4]], [[] ,[3, 2] ,[4, 1]]
len(OPEN)=4; len(CLOSED)=30; COUNT = 30
OPEN is now: [[] ,[4] ,[3, 2, 1]], [[1] ,[3, 2] ,[4]], [[] ,[3, 2] ,[4, 1]]

1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=21; COUNT = 21
OPEN is now: [[] ,[3] ,[4, 2, 1]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=5; len(CLOSED)=22; COUNT = 22
OPEN is now: [[3] ,[] ,[4, 2, 1]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=5; len(CLOSED)=23; COUNT = 23
OPEN is now: [[3, 1] ,[] ,[4, 2]], [[3] ,[1] ,[4, 2]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=24; COUNT = 24
OPEN is now: [[3] ,[1] ,[4, 2]], [[3, 1] ,[2] ,[4]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=25; COUNT = 25
OPEN is now: [[3, 2] ,[1] ,[4]], [[3, 1] ,[2] ,[4]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=26; COUNT = 26
OPEN is now: [[3, 2, 1] ,[] ,[4]], [[3, 2] ,[] ,[4, 1]], [[3, 1] ,[2] ,[4]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4,
3, 2] ,[] ,[1]]
len(OPEN)=7; len(CLOSED)=27; COUNT = 27
OPEN is now: [[3, 2] ,[] ,[4, 1]], [[3, 2, 1] ,[4] ,[]], [[3, 1] ,[2] ,[4]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4,
3, 2] ,[] ,[1]]
len(OPEN)=7; len(CLOSED)=28; COUNT = 28
OPEN is now: [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[4] ,[]], [[3, 1] ,[2] ,[4]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=7; len(CLOSED)=29; COUNT =

len(OPEN)=3; len(CLOSED)=31; COUNT = 31
OPEN is now: [[1] ,[3, 2] ,[4]], [[] ,[3, 2] ,[4, 1]], [[1] ,[4] ,[3, 2]], [[] ,[4, 1] ,[3, 2]]
len(OPEN)=4; len(CLOSED)=32; COUNT = 32
OPEN is now: [[] ,[3, 2] ,[4, 1]], [[1] ,[4] ,[3, 2]], [[] ,[4, 1] ,[3, 2]], [[1] ,[3] ,[4, 2]]
len(OPEN)=4; len(CLOSED)=33; COUNT = 33
OPEN is now: [[1] ,[4] ,[3, 2]], [[] ,[4, 1] ,[3, 2]], [[1] ,[3] ,[4, 2]], [[2] ,[3] ,[4, 1]]
len(OPEN)=4; len(CLOSED)=34; COUNT = 34
OPEN is now: [[] ,[4, 1] ,[3, 2]], [[1] ,[3] ,[4, 2]], [[2] ,[3] ,[4, 1]], [[1] ,[4, 2] ,[3]]
len(OPEN)=4; len(CLOSED)=35; COUNT = 35
OPEN is now: [[1] ,[3] ,[4, 2]], [[2] ,[3] ,[4, 1]], [[1] ,[4, 2] ,[3]], [[2] ,[4, 1] ,[3]]
len(OPEN)=4; len(CLOSED)=36; COUNT = 36
OPEN is now: [[2] ,[3] ,[4, 1]], [[1] ,[4, 2] ,[3]], [[2] ,[4, 1] ,[3]], [[] ,[3, 1] ,[4, 2]], [[] ,[3] ,[4, 2, 1]]
len(OPEN)=5; len(CLOSED)=37; COUNT = 37
OPEN is now: [[1] ,[4, 2] ,[3]], [[2] ,[4, 1] ,[3]], [[] ,[3, 1] ,[4, 2]], [[] ,[3] ,[4, 2, 1]], [[2, 1] ,[3] ,[4]], [[2] ,[3, 1] ,[4]]
len(OPEN)=6; len(CLOSED)=38; COUNT = 38
OPEN is now: [[2] ,[4, 1] ,[3]], [[] ,[3, 1] ,[4, 2]], [[] ,[3] ,[4, 2, 1]], [[2, 1] ,[3] ,[4]], [[2] ,[3, 1] ,[4]], [[] ,[4, 2, 1] ,[3]], [[] ,[4,
2] ,[3, 1]]
len(OPEN)=7; len(CLOSED)=39; COUNT = 39
OPEN is now: [[] ,[3, 1] ,[4, 2]], [[] ,[3] ,[4, 2, 1]], [[2, 1] ,[3] ,[4]], [[2] ,[3, 1] ,[4]], [[] ,[4, 2, 1] ,[3]], [[] ,[4, 2] ,[3, 1]], [[2, 1] ,[4] ,[3]], [[2] ,[4] ,[3, 1]]
len(OPEN)=8; len(CLOSED)=40; COUNT = 40
29
OPEN is now: [[3, 1] ,[2] ,[4]], [[3] ,[2, 1] ,[4]], [[3, 2, 1] ,[4] ,[]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=7; len(CLOSED)=30; COUNT = 30
OPEN is now: [[3] ,[2, 1] ,[4]], [[3, 2, 1] ,[4] ,[]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=31; COUNT = 31
OPEN is now: [[] ,[2, 1] ,[4, 3]], [[3, 2, 1] ,[4] ,[]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=32; COUNT = 32
OPEN is now: [[1] ,[2] ,[4, 3]], [[] ,[2] ,[4, 3, 1]], [[3, 2, 1] ,[4] ,[]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4,
3, 2] ,[] ,[1]]
len(OPEN)=7; len(CLOSED)=33; COUNT = 33
OPEN is now: [[] ,[2] ,[4, 3, 1]], [[1] ,[] ,[4, 3, 2]], [[3, 2, 1] ,[4] ,[]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=7; len(CLOSED)=34; COUNT = 34
OPEN is now: [[2] ,[] ,[4, 3, 1]], [[1] ,[] ,[4, 3, 2]], [[3, 2, 1] ,[4] ,[]], [[2, 1] ,[] ,[4, 3]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=7; len(CLOSED)=35; COUNT = 35
OPEN is now: [[2, 1] ,[] ,[4, 3]], [[2] ,[1] ,[4, 3]], [[1] ,[] ,[4, 3, 2]], [[3, 2, 1] ,[4] ,[]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4,
3, 2] ,[] ,[1]]
len(OPEN)=7; len(CLOSED)=36; COUNT = 36
OPEN is now: [[2] ,[1] ,[4, 3]], [[1] ,[] ,[4, 3, 2]], [[3, 2, 1] ,[4] ,[]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=37; COUNT = 37
OPEN is now: [[] ,[1] ,[4, 3, 2]], [[1] ,[] ,[4, 3,

OPEN is now: [[] ,[3] ,[4, 2, 1]], [[2, 1] ,[3] ,[4]], [[2] ,[3, 1] ,[4]], [[] ,[4, 2, 1] ,[3]], [[] ,[4, 2] ,[3, 1]], [[2, 1] ,[4] ,[3]], [[2] ,[4] ,[3, 1]], [[2] ,[3, 1] ,[4]]
len(OPEN)=8; len(CLOSED)=41; COUNT = 41
OPEN is now: [[2, 1] ,[3] ,[4]], [[2] ,[3, 1] ,[4]], [[] ,[4, 2, 1] ,[3]], [[] ,[4, 2] ,[3, 1]], [[2, 1] ,[4] ,[3]], [[2] ,[4] ,[3, 1]], [[2] ,[3, 1] ,[4]], [[3] ,[] ,[4, 2, 1]]
len(OPEN)=8; len(CLOSED)=42; COUNT = 42
OPEN is now: [[2] ,[3, 1] ,[4]], [[] ,[4, 2, 1] ,[3]], [[] ,[4, 2] ,[3, 1]], [[2, 1] ,[4] ,[3]], [[2] ,[4] ,[3, 1]], [[2] ,[3, 1] ,[4]], [[3] ,[] ,[4, 2, 1]], [[2, 1] ,[] ,[4, 3]]
len(OPEN)=8; len(CLOSED)=43; COUNT = 43
OPEN is now: [[] ,[4, 2, 1] ,[3]], [[] ,[4, 2] ,[3, 1]], [[2, 1] ,[4] ,[3]], [[2] ,[4] ,[3, 1]], [[2] ,[3, 1] ,[4]], [[3] ,[] ,[4, 2, 1]], [[2, 1] ,[] ,[4, 3]]
len(OPEN)=7; len(CLOSED)=44; COUNT = 44
OPEN is now: [[] ,[4, 2] ,[3, 1]], [[2, 1] ,[4] ,[3]], [[2] ,[4] ,[3, 1]], [[2] ,[3, 1] ,[4]], [[3] ,[] ,[4, 2, 1]], [[2, 1] ,[] ,[4, 3]], [[3] ,[4, 2, 1] ,[]]
len(OPEN)=7; len(CLOSED)=45; COUNT = 45
OPEN is now: [[2, 1] ,[4] ,[3]], [[2] ,[4] ,[3, 1]], [[2] ,[3, 1] ,[4]], [[3] ,[] ,[4, 2, 1]], [[2, 1] ,[] ,[4, 3]], [[3] ,[4, 2, 1] ,[]]
len(OPEN)=6; len(CLOSED)=46; COUNT = 46
OPEN is now: [[2] ,[4] ,[3, 1]], [[2] ,[3, 1] ,[4]], [[3] ,[] ,[4, 2, 1]], [[2, 1] ,[] ,[4, 3]], [[3] ,[4, 2, 1] ,[]], [[2, 1] ,[4, 3] ,[]]
len(OPEN)=6; len(CLOSED)=47; COUNT = 47
OPEN is now: [[2] ,[3, 1] ,[4]], [[3] ,[] ,[4, 2, 1]], [[2, 1] ,[] ,[4, 3]], [[3] ,[4, 2, 1] ,[]], [[2, 1] ,[4, 3] ,[]]
len(OPEN)=5; len(CLOSED)=48; COUNT = 48
OPEN is now: [[3] ,[] ,[4, 2, 1]], [[2, 1] ,[] ,[4, 2]], [[3, 2, 1] ,[4] ,[]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=38; COUNT = 38
OPEN is now: [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]], [[3, 2, 1] ,[4] ,[]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=6; len(CLOSED)=39; COUNT = 39
OPEN is now: [[] ,[] ,[4, 3, 2, 1]], [[3, 2, 1] ,[4] ,[]], [[4, 2, 1] ,[] ,[3]], [[4, 3, 1] ,[2] ,[]], [[4, 3, 2] ,[] ,[1]]
len(OPEN)=5; len(CLOSED)=40; COUNT = 40
The Tower Transport is Triumphant!
Solution path:
[[4, 3, 2, 1] ,[] ,[]]
[[4, 3, 2] ,[1] ,[]]
[[4, 3] ,[1] ,[2]]
[[4, 3, 1] ,[] ,[2]]
[[4, 3] ,[] ,[2, 1]]
[[4] ,[3] ,[2, 1]]
[[4, 1] ,[3] ,[2]]
[[4] ,[3, 1] ,[2]]
[[4, 2] ,[3, 1] ,[]]
[[4, 2, 1] ,[3] ,[]]
[[4, 2] ,[3] ,[1]]
[[4] ,[3, 2] ,[1]]
[[4, 1] ,[3, 2] ,[]]
[[4] ,[3, 2, 1] ,[]]
[[] ,[3, 2, 1] ,[4]]
[[1] ,[3, 2] ,[4]]
[[] ,[3, 2] ,[4, 1]]
[[2] ,[3] ,[4, 1]]
[[2, 1] ,[3] ,[4]]
[[2] ,[3, 1] ,[4]]
[[] ,[3, 1] ,[4, 2]]
[[1] ,[3] ,[4, 2]]
[[] ,[3] ,[4, 2, 1]]
[[3] ,[] ,[4, 2, 1]]
[[3, 1] ,[] ,[4, 2]]
[[3] ,[1] ,[4, 2]]
[[3, 2] ,[1] ,[4]]
[[3, 2, 1] ,[] ,[4]]

3]], [[3] ,[4, 2, 1] ,[]], [[2, 1] ,[4, 3] ,[]] len(OPEN)=4; len(CLOSED)=49; COUNT = 49 OPEN is now: [[2, 1] ,[] ,[4, 3]], [[3] ,[4, 2, 1] ,[]], [[2, 1] ,[4, 3] ,[]], [[3, 1] ,[] ,[4, 2]], [[3] ,[1] ,[4, 2]] len(OPEN)=5; len(CLOSED)=50; COUNT = 50 OPEN is now: [[3] ,[4, 2, 1] ,[]], [[2, 1] ,[4, 3] ,[]], [[3, 1] ,[] ,[4, 2]], [[3] ,[1] ,[4, 2]], [[2] ,[1] ,[4, 3]], [[2] ,[] ,[4, 3, 1]] len(OPEN)=6; len(CLOSED)=51; COUNT = 51 OPEN is now: [[2, 1] ,[4, 3] ,[]], [[3, 1] ,[] ,[4, 2]], [[3] ,[1] ,[4, 2]], [[2] ,[1] ,[4, 3]], [[2] ,[] ,[4, 3, 1]], [[3, 1] ,[4, 2] ,[]], [[3] ,[4, 2] ,[1]] len(OPEN)=7; len(CLOSED)=52; COUNT = 52 OPEN is now: [[3, 1] ,[] ,[4, 2]], [[3] ,[1] ,[4, 2]], [[2] ,[1] ,[4, 3]], [[2] ,[] ,[4, 3, 1]], [[3, 1] ,[4, 2] ,[]], [[3] ,[4, 2] ,[1]], [[2] ,[4, 3, 1] ,[]], [[2] ,[4, 3] ,[1]] len(OPEN)=8; len(CLOSED)=53; COUNT = 53 OPEN is now: [[3] ,[1] ,[4, 2]], [[2] ,[1] ,[4, 3]], [[2] ,[] ,[4, 3, 1]], [[3, 1] ,[4, 2] ,[]], [[3] ,[4, 2] ,[1]], [[2] ,[4, 3, 1] ,[]], [[2] ,[4, 3] ,[1]], [[3, 1] ,[2] ,[4]] len(OPEN)=8; len(CLOSED)=54; COUNT = 54 OPEN is now: [[2] ,[1] ,[4, 3]], [[2] ,[] ,[4, 3, 1]], [[3, 1] ,[4, 2] ,[]], [[3] ,[4, 2] ,[1]], [[2] ,[4, 3, 1] ,[]], [[2] ,[4, 3] ,[1]], [[3, 1] ,[2] ,[4]], [[3, 2] ,[1] ,[4]] len(OPEN)=8; len(CLOSED)=55; COUNT = 55 OPEN is now: [[2] ,[] ,[4, 3, 1]], [[3, 1] ,[4, 2] ,[]], [[3] ,[4, 2] ,[1]], [[2] ,[4, 3, 1] ,[]], [[2] ,[4, 3] ,[1]], [[3, 1] ,[2] ,[4]], [[3, 2] ,[1] ,[4]], [[] ,[1] ,[4, 3, 2]] len(OPEN)=8; len(CLOSED)=56; COUNT = 56 OPEN is now: [[3, 1] ,[4, 2] ,[]], [[3] ,[4, 2] ,[1]], [[2] ,[4, 3, 1] ,[]], [[2] ,[4, 3] ,[1]], [[3, 1] ,[2] ,[4]],	[[3, 2] ,[] ,[4, 1]] [[3] ,[2] ,[4, 1]] [[3, 1] ,[2] ,[4]] [[3] ,[2, 1] ,[4]] [[] ,[2, 1] ,[4, 3]] [[1] ,[2] ,[4, 3]] [[] ,[2] ,[4, 3, 1]] [[2] ,[] ,[4, 3, 1]] [[2, 1] ,[] ,[4, 3]] [[2] ,[1] ,[4, 3]] [[] ,[1] ,[4, 3, 2]] [[1] ,[] ,[4, 3, 2]] [[] ,[] ,[4, 3, 2, 1]] Length of solution path found: 40 edges 40 states expanded. MAX_OPEN_LENGTH = 7

[[3, 2] ,[1] ,[4]], [[] ,[1] ,[4, 3, 2]], [[] ,[2] ,[4, 3, 1]] len(OPEN)=8; len(CLOSED)=57; COUNT = 57 OPEN is now: [[3] ,[4, 2] ,[1]], [[2] ,[4, 3, 1] ,[]], [[2] ,[4, 3] ,[1]], [[3, 1] ,[2] ,[4]], [[3, 2] ,[1] ,[4]], [[] ,[1] ,[4, 3, 2]], [[] ,[2] ,[4, 3, 1]], [[3, 1] ,[4] ,[2]] len(OPEN)=8; len(CLOSED)=58; COUNT = 58 OPEN is now: [[2] ,[4, 3, 1] ,[]], [[2] ,[4, 3] ,[1]], [[3, 1] ,[2] ,[4]], [[3, 2] ,[1] ,[4]], [[] ,[1] ,[4, 3, 2]], [[] ,[2] ,[4, 3, 1]], [[3, 1] ,[4] ,[2]], [[3, 2] ,[4] ,[1]] len(OPEN)=8; len(CLOSED)=59; COUNT = 59 OPEN is now: [[2] ,[4, 3] ,[1]], [[3, 1] ,[2] ,[4]], [[3, 2] ,[1] ,[4]], [[] ,[1] ,[4, 3, 2]], [[] ,[2] ,[4, 3, 1]], [[3, 1] ,[4] ,[2]], [[3, 2] ,[4] ,[1]], [[] ,[4, 3, 1] ,[2]] len(OPEN)=8; len(CLOSED)=60; COUNT = 60 OPEN is now: [[3, 1] ,[2] ,[4]], [[3, 2] ,[1] ,[4]], [[] ,[1] ,[4, 3, 2]], [[] ,[2] ,[4, 3, 1]], [[3, 1] ,[4] ,[2]], [[3, 2] ,[4] ,[1]], [[] ,[4, 3, 1] ,[2]], [[] ,[4, 3, 2] ,[1]] len(OPEN)=8; len(CLOSED)=61; COUNT = 61 OPEN is now: [[3, 2] ,[1] ,[4]], [[] ,[1] ,[4, 3, 2]], [[] ,[2] ,[4, 3, 1]], [[3, 1] ,[4] ,[2]], [[3, 2] ,[4] ,[1]], [[] ,[4, 3, 1] ,[2]], [[] ,[4, 3, 2] ,[1]], [[3] ,[2, 1] ,[4]], [[3] ,[2] ,[4, 1]] len(OPEN)=9; len(CLOSED)=62; COUNT = 62 OPEN is now: [[] ,[1] ,[4, 3, 2]], [[] ,[2] ,[4, 3, 1]], [[3, 1] ,[4] ,[2]], [[3, 2] ,[4] ,[1]], [[] ,[4, 3, 1] ,[2]], [[] ,[4, 3, 2] ,[1]], [[3] ,[2, 1] ,[4]], [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[] ,[4]], [[3, 2] ,[] ,[4, 1]] len(OPEN)=10; len(CLOSED)=63; COUNT = 63 OPEN is now: [[] ,[2] ,[4, 3, 1]], [[3, 1] ,[4] ,[2]], [[3, 2] ,[4] ,[1]], [[] ,[4, 3, 1] ,[2]], [[] ,[4, 3, 2] ,[1]], [[3] ,[2, 1] ,[4]], [[3] ,[2] ,[4, 1]], [[3, 2,	

1] ,[] ,[4]], [[3, 2] ,[] ,[4, 1]], [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]] len(OPEN)=11; len(CLOSED)=64; COUNT = 64 OPEN is now: [[3, 1] ,[4] ,[2]], [[3, 2] ,[4] ,[1]], [[] ,[4, 3, 1] ,[2]], [[] ,[4, 3, 2] ,[1]], [[3] ,[2, 1] ,[4]], [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[] ,[4]], [[3, 2] ,[] ,[4, 1]], [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]], [[1] ,[2] ,[4, 3]], [[] ,[2, 1] ,[4, 3]] len(OPEN)=12; len(CLOSED)=65; COUNT = 65 OPEN is now: [[3, 2] ,[4] ,[1]], [[] ,[4, 3, 1] ,[2]], [[] ,[4, 3, 2] ,[1]], [[3] ,[2, 1] ,[4]], [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[] ,[4]], [[3, 2] ,[] ,[4, 1]], [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]], [[1] ,[2] ,[4, 3]], [[] ,[2, 1] ,[4, 3]], [[3] ,[4, 1] ,[2]], [[3] ,[4] ,[2, 1]] len(OPEN)=13; len(CLOSED)=66; COUNT = 66 OPEN is now: [[] ,[4, 3, 1] ,[2]], [[] ,[4, 3, 2] ,[1]], [[3] ,[2, 1] ,[4]], [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[] ,[4]], [[3, 2] ,[] ,[4, 1]], [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]], [[1] ,[2] ,[4, 3]], [[] ,[2, 1] ,[4, 3]], [[3] ,[4, 1] ,[2]], [[3] ,[4] ,[2, 1]], [[3, 2, 1] ,[4] ,[]], [[3, 2] ,[4, 1] ,[]] len(OPEN)=14; len(CLOSED)=67; COUNT = 67 OPEN is now: [[] ,[4, 3, 2] ,[1]], [[3] ,[2, 1] ,[4]], [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[] ,[4]], [[3, 2] ,[] ,[4, 1]], [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]], [[1] ,[2] ,[4, 3]], [[] ,[2, 1] ,[4, 3]], [[3] ,[4, 1] ,[2]], [[3] ,[4] ,[2, 1]], [[3, 2, 1] ,[4] ,[]], [[3, 2] ,[4, 1] ,[]], [[1] ,[4, 3] ,[2]], [[] ,[4, 3] ,[2, 1]] len(OPEN)=15; len(CLOSED)=68; COUNT = 68 OPEN is now: [[3] ,[2, 1] ,[4]], [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[] ,[4]], [[3, 2] ,[] ,[4, 1]], [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]], [[1] ,[2] ,[4, 3]], [[] ,[2, 1] ,[4, 3]], [[3] ,[4, 1] ,[2]], [[3] ,[4] ,[2, 1]], [[3, 2, 1] ,[4] ,[]], [[3, 2] ,[4, 1] ,[]], [[1] ,[4, 3] ,[2]], [[] ,[4, 3] ,[2, 1]], [[1] ,[4, 3, 2] ,[]], [[] ,[4, 3, 2, 1] ,[]]	

len(OPEN)=16; len(CLOSED)=69; COUNT = 69 OPEN is now: [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[] ,[4]], [[3, 2] ,[] ,[4, 1]], [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]], [[1] ,[2] ,[4, 3]], [[] ,[2, 1] ,[4, 3]], [[3] ,[4, 1] ,[2]], [[3] ,[4] ,[2, 1]], [[3, 2, 1] ,[4] ,[]], [[3, 2] ,[4, 1] ,[]], [[1] ,[4, 3] ,[2]], [[] ,[4, 3] ,[2, 1]], [[1] ,[4, 3, 2] ,[]], [[] ,[4, 3, 2, 1] ,[]], [[3] ,[2] ,[4, 1]] len(OPEN)=16; len(CLOSED)=70; COUNT = 70 OPEN is now: [[3, 2, 1] ,[] ,[4]], [[3, 2] ,[] ,[4, 1]], [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]], [[1] ,[2] ,[4, 3]], [[] ,[2, 1] ,[4, 3]], [[3] ,[4, 1] ,[2]], [[3] ,[4] ,[2, 1]], [[3, 2, 1] ,[4] ,[]], [[3, 2] ,[4, 1] ,[]], [[1] ,[4, 3] ,[2]], [[] ,[4, 3] ,[2, 1]], [[1] ,[4, 3, 2] ,[]], [[] ,[4, 3, 2, 1] ,[]], [[3] ,[2] ,[4, 1]] len(OPEN)=15; len(CLOSED)=71; COUNT = 71 OPEN is now: [[3, 2] ,[] ,[4, 1]], [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]], [[1] ,[2] ,[4, 3]], [[] ,[2, 1] ,[4, 3]], [[3] ,[4, 1] ,[2]], [[3] ,[4] ,[2, 1]], [[3, 2, 1] ,[4] ,[]], [[3, 2] ,[4, 1] ,[]], [[1] ,[4, 3] ,[2]], [[] ,[4, 3] ,[2, 1]], [[1] ,[4, 3, 2] ,[]], [[] ,[4, 3, 2, 1] ,[]], [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[4] ,[]] len(OPEN)=15; len(CLOSED)=72; COUNT = 72 OPEN is now: [[1] ,[] ,[4, 3, 2]], [[] ,[] ,[4, 3, 2, 1]], [[1] ,[2] ,[4, 3]], [[] ,[2, 1] ,[4, 3]], [[3] ,[4, 1] ,[2]], [[3] ,[4] ,[2, 1]], [[3, 2, 1] ,[4] ,[]], [[3, 2] ,[4, 1] ,[]], [[1] ,[4, 3] ,[2]], [[] ,[4, 3] ,[2, 1]], [[1] ,[4, 3, 2] ,[]], [[] ,[4, 3, 2, 1] ,[]], [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[4] ,[]] len(OPEN)=14; len(CLOSED)=73; COUNT = 73 OPEN is now: [[] ,[] ,[4, 3, 2, 1]], [[1] ,[2] ,[4, 3]], [[] ,[2, 1] ,[4, 3]], [[3] ,[4, 1] ,[2]], [[3] ,[4] ,[2, 1]], [[3, 2, 1] ,[4] ,[]], [[3, 2] ,[4, 1] ,[]], [[1] ,[4, 3] ,[2]], [[] ,[4, 3] ,[2, 1]], [[1] ,[4, 3, 2] ,[]], [[] ,[4, 3, 2, 1] ,[]], [[3] ,[2] ,[4, 1]], [[3, 2, 1] ,[4] ,[]], [[1] ,[2] ,[4, 3]] len(OPEN)=14; len(CLOSED)=74; COUNT = 74	

The Tower Transport is Triumphant! Solution path: [[4, 3, 2, 1] ,[] ,[]] [[4, 3, 2] ,[1] ,[]] [[4, 3] ,[1] ,[2]] [[4, 3] ,[] ,[2, 1]] [[4] ,[3] ,[2, 1]] [[4, 1] ,[3] ,[2]] [[4, 1] ,[3, 2] ,[]] [[4] ,[3, 2, 1] ,[]] [[] ,[3, 2, 1] ,[4]] [[] ,[3, 2] ,[4, 1]] [[2] ,[3] ,[4, 1]] [[2, 1] ,[3] ,[4]] [[2, 1] ,[] ,[4, 3]] [[2] ,[1] ,[4, 3]] [[] ,[1] ,[4, 3, 2]] [[] ,[] ,[4, 3, 2, 1]] Length of solution path found: 15 edges 74 states expanded. MAX_OPEN_LENGTH = 16	
Length of Path(BFS): 15	Length of Path(DFS): 40
Nodes expanded(BFS): 74	Nodes expanded(DFS): 40