Aaron Hong (ahong02)

Iterative DFS

Problem	Solution path	Path length	# of nodes expanded
Humans, Robots, Ferry	H on left:3 R on left:3 H on right:0 R on right:0 ferry is on the left. H on left:2 R on left:2 H on right:1 R on right:1 ferry is on the right. H on left:3 R on left:2 H on right:0 R on right:1 ferry is on the left. H on left:0 R on left:2 H on right:3 R on right:1 ferry is on the right. H on left:2 R on left:2 H on right:1 R on right:1 ferry is on the left. H on left:1 R on left:1 H on right:2 R on right:2 ferry is on the right. H on left:3 R on left:1 H on right:0 R on right:2 ferry is on the left. H on left:0 R on left:1 H on right:3	9	10

	R on right:2 ferry is on the right. H on left:1 R on left:1 H on right:2 R on right:2 ferry is on the left. H on left:0 R on left:0 H on right:3 R on right:3 ferry is on the right.		
Farmer, Fox, Chicken, Grain	farmer is on the left fox is on the left chicken is on the left grain is on the left farmer is on the right fox is on the left chicken is on the right grain is on the left farmer is on the left fox is on the left chicken is on the right grain is on the left farmer is on the right fox is on the left chicken is on the right grain is on the right farmer is on the left fox is on the left chicken is on the left grain is on the right farmer is on the right fox is on the right chicken is on the left grain is on the right farmer is on the left fox is on the right chicken is on the left grain is on the right	7	7

	farmer is on the right fox is on the right chicken is on the right grain is on the right		
Towers of Hanoi (4)	[[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, 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]]	40	40

Iterative BFS

Problem	Solution path	Path length	# of nodes expanded
Humans, Robots, Ferry	H on left:3 R on left:3 H on right:0 R on right:0 ferry is on the left. H on left:2 R on left:2 H on right:1 R on right:1 ferry is on the right. H on left:3 R on left:2 H on right:0 R on right:1 ferry is on the left. H on left:0 R on left:2 H on right:3 R on right:1 ferry is on the right. H on left:2 R on left:2 H on right:1 R on right:1 ferry is on the left. H on left:0 R on left:1 H on right:3 R on right:2 ferry is on the right. H on left:1 R on left:1 H on right:2 R on right:2 ferry is on the left. H on left:0 R on left:0 H on right:3	7	10

	R on right:3 ferry is on the right.		
Farmer, Fox, Chicken, Grain	farmer is on the left fox is on the left chicken is on the left grain is on the left farmer is on the right fox is on the left chicken is on the right grain is on the left farmer is on the left fox is on the left chicken is on the right grain is on the left farmer is on the right fox is on the left chicken is on the right grain is on the right farmer is on the left fox is on the left chicken is on the left grain is on the right farmer is on the right fox is on the right chicken is on the left grain is on the right farmer is on the left fox is on the right chicken is on the left grain is on the right farmer is on the right fox is on the right chicken is on the right grain is on the right	7	9

Towers of Hanoi (4)	[[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]]	15	70

In the Towers of Hanoi problem, the breadth-first search algorithm had a larger maximum open list than the depth-first search. This is because the breadth-first search considered every possibility branching from the initial state, leading to many more states needing to be considered. The depth-first search mainly considered only one branch at each level and as a result ventured much deeper into the tree than the breadth-first search.

This led to the breadth-first search yielding a much shorter solution path than the depth-first search. This is because the depth-first search algorithm returns the first solution path that it encounters, as opposed to the breadth-first search, which returns the shortest solution path it encounters. In the context of the problem itself, it seems there are many solution paths for Towers of Hanoi.