

CS1571
Fall 2019
9/9 In-Class Worksheet

Name: Simran Gidwani

Where were you sitting in class today: Middle center

A. Implementation of Actions and Transition Models

Download search.py from CourseWeb. This is part of a codebase supplemental to the textbook that implements the algorithms described (<https://github.com/aimacode/aima-python>). Among other things, search.py provides an implementation of the EightPuzzle problem. Find the implementation, and answer the following questions about the code.

1. Is the list returned by *actions* dependent on the current state?

The list returned by actions depends on the current state that the problem is in because depending what spot on the board you're in you may have a limited number of actions so that you don't go off the edge of the board. It removes that option if you are on the edge.

2. What does *result* return and how is it calculated?

Result returns the state that is chosen from the last action.

3. Now, look at the **Node** class. Explain how it uses the *actions* and *result* methods to generate a list of child nodes.

The expand function uses the actions methods to display the nodes reachable from a current state. The child_node function uses the result method to determine the next action and creates the state.

B. Iterative Deepening DFS

4. Is idDFS complete?

It is complete since there is a cutoff on the depth so will always find a solution

5. Is it optimal?

Optimal because the cutoff would allow the algorithm to find the shortest path.

6. What is the time complexity?

$O(b^d)$

7. What is the space complexity?

I think the space complexity depends on the way that the tree is being stored at each iteration but I think it would be $O(b^d)$

C. Bidirectional Search

8. List two advantages of bidirectional search over DFS, and two disadvantages.