

CS1571
Fall 2019
9/25 In-Class Worksheet

Name: Simran Gidwani

Where were you sitting in class today: Center Middle

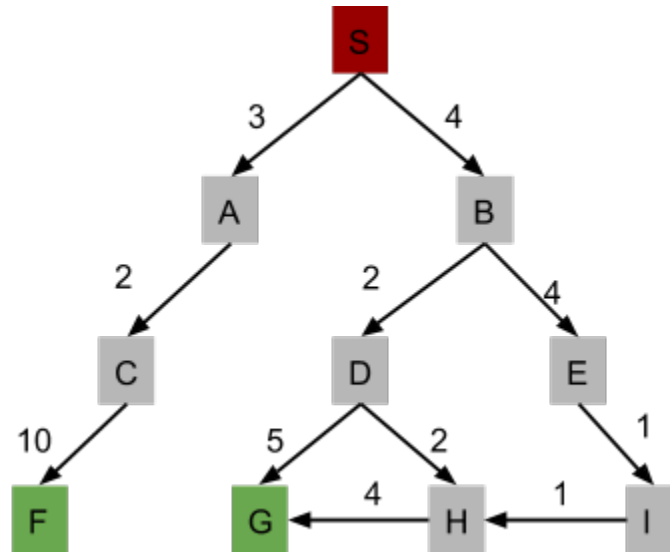
A. Iterative Deepening A*

1. Go to <https://algorithmsinsight.files.wordpress.com/2016/03/ida-star.gif?w=1326>

Based on this animation, explain the key points of how iterative deepening A* works. The values on the nodes represent the results of $f(n)$.

In each iteration, the algorithm checks to see if the child node exceeds the threshold value and if it does then it expands that node. At the end of each iteration, it stores only the minimum value from the expanded nodes to be the threshold for the next iteration and continues until reaching its goal.

2. What is the order in which the states are visited under IDA* for this graph?



You should assume the following values for the heuristic function:

state	S	A	B	C	D	E	F	G	H	I
$h(n)$	10	10	6	9	4	3	0	0	4	4

Iteration	Cut-off	Order of nodes expanded
1	10	S, B(f val = 10), D(6)
2	11	S, B, , D, G
3		
4		

B. Minimax

3. Given this game board, draw a search tree based on the Minimax algorithm (the pseudocode is included for your convenience). Based on the search tree, explain how X chooses its next move, and what it is

Minimax algorithm
<pre>function MINIMAX-DECISION(<i>state</i>) returns an action <i>v</i> ← MAX-VALUE(<i>state</i>) return the action in SUCCESSORS(<i>state</i>) with value <i>v</i></pre>
<pre>function MAX-VALUE(<i>state</i>) returns a utility value if TERMINAL-TEST(<i>state</i>) then return UTILITY(<i>state</i>) <i>v</i> ← $-\infty$ for <i>a, s</i> in SUCCESSORS(<i>state</i>) do <i>v</i> ← MAX(<i>v</i>, MIN-VALUE(<i>s</i>)) return <i>v</i></pre>
<pre>function MIN-VALUE(<i>state</i>) returns a utility value if TERMINAL-TEST(<i>state</i>) then return UTILITY(<i>state</i>) <i>v</i> ← ∞ for <i>a, s</i> in SUCCESSORS(<i>state</i>) do <i>v</i> ← MIN(<i>v</i>, MAX-VALUE(<i>s</i>)) return <i>v</i></pre>

		O
O	X	O
	X	