CS 440: Intro to AI **Rutgers: Fall 2021**

Midterm Exam

November 15, 2021

Name: Ryan Coslove (rmc326)

Problem 1.

Solution.

1a.)

BFS: A, B, C, D, E, F, G, H, Z

BFS	Step	List
	0	A
	1	B, C, D, E
	2	C, D, E, F
	3	D, E, F, G
	4	E, F, G, H
	5	F, G, H, Z
		Found Goal State - Z

DFS: A, B, C, D, E, F, G, H, Z

DFS	Step	List
	0	A
	1	B, C, D, E
	2	C, D, E, F
	3	D, E, F, G
	4	E, F, G, H
	5	F, G, H, Z
		Found Goal State - Z

IDS: A, B, F, I, G, C, A, D, H, J, K, Z

IDS	Step	List
	0	A
	1	B
	2	F
	3	I, L
	4	I, G,
	5	G, C, J
	6	C, A
	7	D
	8	H
	9	J, K
	10	K
	11	Z
		Found Goal State - Z

A*: A, B, F, L, Z

A*	Step	List
	0	A
	1	B, C, D, E
	2	F
	3	L, I
	4	Z
		Found Goal State – Z

1b.)

A* did find the optimal path. It took the least amount of steps/iterations to find Z because from A, BCDE all had values of 4, so alphabetically we choose B. Then we go to F. The lowest heuristic value from F was L. Then L leads to Z. BFS and DFS took the same amount of steps, each using uniform cost and alphabet as the tiebreaker. IDS took the most steps and was the least optimal path.

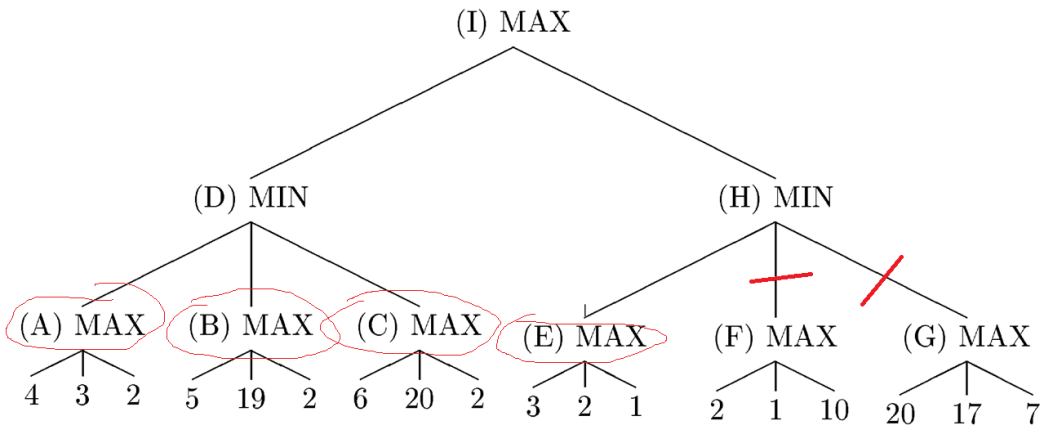
1c.)

Yes all methods found a solution using graph-search. All successfuly found goal node Z. A* was the most optimal, IDS was the least optimal.

Problem 2.

Solution.

2a.)



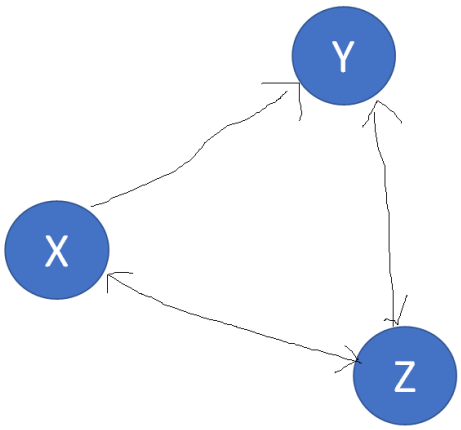
2b.) MAX will move to node D. This is because the MAX of ABC are 4, 19, and 20, respectively. This will return 4 as the MIN for D. The MAX of EFG are 3, 10, 20, respectively. This will return 3 as the MIN of H. So the MAX will choose node D because $4 > 3$ and $D > H$.

2c.) The MAX of C is 20. The MAX of G is also 20. So for (D) MIN = 20 and (H) MIN = 20. Given that both MAX values are equal, it does not matter if MAX goes to D or H as its first move.

Problem 3.

Solution.

3a.)



3b.)

Domains	Agenda	Arc
X = {1,…,10} -> {6, 7, 8, 9, 10} -> {9, 10}	X > Y	X > Y
Y = {5,…,15} -> {5, 6, 7}	Y + Z = 12	Y < X
Z = {5,…, 20} -> {5, 6, 7}	X + Z = 16	Y + Z = 12
X > Y		X + Z = 16
Y + Z = 12		
X + Z = 16		

Problem 4.

Solution.

Most Constrained Variable: In the search tree, we alternate between choosing variables and choosing values for the variables. At the stage where we choose a variable, we break the search and backtrack if we find one variable that cannot be satisfied. Finding only one such variable is sufficient to say that something was wrong earlier, and we go up in the search to try other assignments. Therefore, we want to fail quickly, which will save us the trouble of trying many variables before finding the one that fails.

Least Constraining Value: Once we choose a variable, we have to try all the possible values before we can say that it failed. Therefore, we will be only choosing time with values that fail (since we will still have to check the remaining values). But if we succeed, the search stops and we don't have to try the remaining values.
