<u>Student Name: Pratik Deshpande</u> <u>Student NetID:PRD190001</u>

University of Texas at Dallas
Department of Computer Science
CS 6364.001 – Artificial Intelligence
Fall 2020
Instructor: Dr. Sanda Harabagiu

Instructions: Do not communicate with anyone in any shape or form. This is an independent exam. Do not delete any problem formulation, just attach your answer in the space provided. If the problem is deleted and you send only the answer, you shall receive ZERO points.
Copy and paste the Mid-Term Exam into a Word document, enter your answers (either by typing in Word, or by inserting a VERY CLEAR picture of your hand-written solution) and transform the file of the exam into a PDF format. If we cannot clearly read the picture, you will get ZERO for that answer! Make sure that you insert EACH answer immediately after EACH question. Failure to do so will result in ZERO points for the entire exam! Submit the PDF file with the name **MidTerM_Exam_netID.pdf**, where netID is your unique netid provided by UTD. If you submit your exam in any other format your will receive ZERO points. The Midterm shall be submitted in eLearning <u>before the deadline</u>. No late submissions shall be graded! Any cheating attempt will determine the ENTIRE grade of the mid-term to become ZERO.

<u>Problem 1 (50 points)</u>

Proteins have an amino acid "alphabet" of 11 elements: AM1, AM2, ..., AM11*. Amino acids are chemically linked together to form protein chains. Between amino acids there are chemical links of different strengths. Suppose you examine under microscope a sample of a protein that belongs to an alien species, having only 11 amino acids. You want to generate an optimal path between* AM1 *and* AM2 *using the A* search algorithm. You are given the strengths of the chemical links in the sample as a graph representation:*

Oracle distance to AM2		The Graph			
AM1	160	AM11	----- AM4	::::	50
AM3	100	AM11	----- AM10	::::	150
AM4	200	AM11	----- AM9	::::	15
AM5	120	AM4	----- AM7	::::	40
AM6	80	AM7	----- AM8	::::	180
AM7	250	AM7	----- AM6	::::	110
AM8	40	AM9	----- AM8	::::	70
AM9	60	AM10	----- AM2	::::	30
AM10	25	AM8	----- AM2	::::	45
AM11	100	AM10	----- AM3	::::	80
*******************************		AM3	----- AM5	::::	50
*******************************		AM5	----- AM1	::::	40
*******************************		AM1	----- AM6	::::	70
*******************************		AM6	----- AM8	::::	20
*******************************		AM1	----- AM4	::::	350
*******************************					

An oracle also gives you the heuristic distance values to AM2 *from each other amino acid in the sample. This heuristic is consistent. Specify if you will use TREE-SEARCH or GRAPH-SEARCH. Motivate your decision.* **(5 points)**

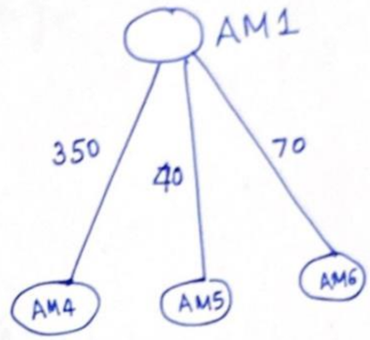
Provide the path of amino acids from AM1 *to* AM2 *as well as the cost of obtaining it. it. Describe <u>at each step of the search</u> (1) what amino acids you have on the search frontier; (2) the current list of explored amino acids; (3) the current path from AM1 to the current amino acid and the cost of that path. Show the successors of each current node, show how you compute all the evaluation functions and which node you select for the next step.* **(45 points)**

Solution:

The use of A* graph-search would be optimal here as the heuristic, “Oracle-distance” that is provided is consistent.

A* Graph Search:-

Step 1:-



Current Node: AM1

Is AM1 Goal?: NO

Children: {AM4, AM6, AM5}

Evaluation Function:-

AM4 = 350 + 200 = 550

AM6 = 70 + 80 = 150

AM5 = 40 + 120 = 160

Frontier = { AM6 AM5 AM4 / 150 160 550 }

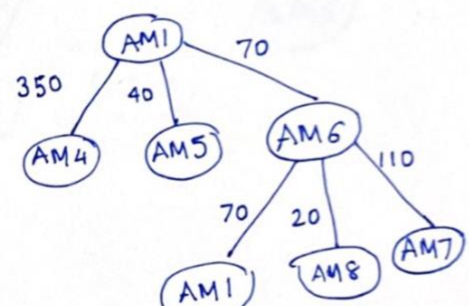
AM6	AM5	AM4
150	160	550

Explored = {AM1}

Next Node = {AM6}

Path cost: 0

Step 2:-



Current Node: AM6

Is AM6 Goal? NO

Children: {AM1, AM8, AM7}

Evaluation Function:-

AM1 = 70 + 70 + 160 = 300

AM8 = 70 + 20 + 40 = 130

AM7 = 70 + 110 + 250 = 430

Frontier =

AM8	AM5	AM7	AM4
130	160	430	550

*As AM1 in explored we dont insert it in frontier.

Explored = {AM1 AM6}

Next Node = AM8

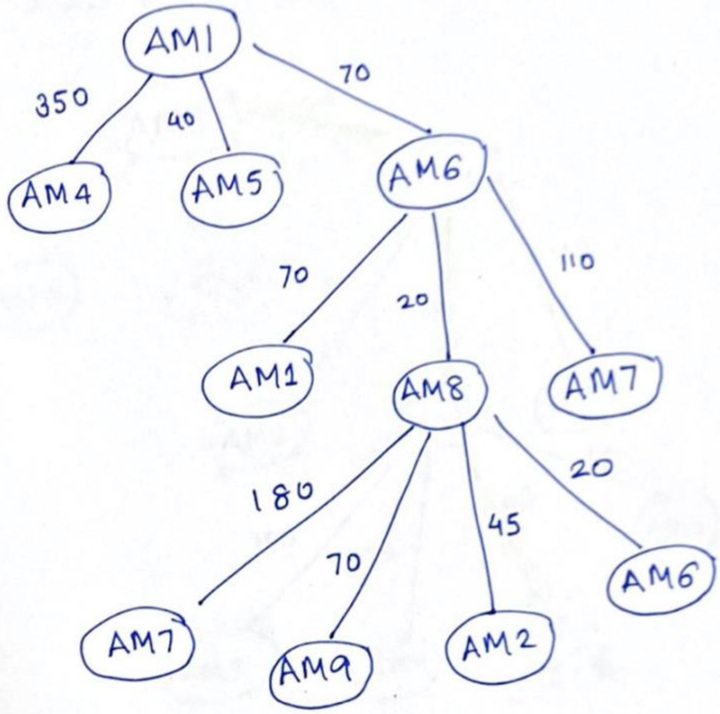
Path cost: 0 + 70 = 70

Step 3:-

Current Node: AM8

Is AM8 a Goal Node? NO

Children: {AM7, AM9, AM2, AM6}



Evaluation Function:-

AM7 = 90 + 180 + 250 = 520

AM9 = 90 + 70 + 60 = 220

AM2 = 90 + 45 + 0 = 135

AM6 = 90 + 20 + 80 = 190

Frontier =	AM2	AM5	AM9	AM7	AM4
	135	160	220	430	550

Explored = {AM1 AM6 AM8}

Next Node : AM2 PathCost 70 + 20 = 90

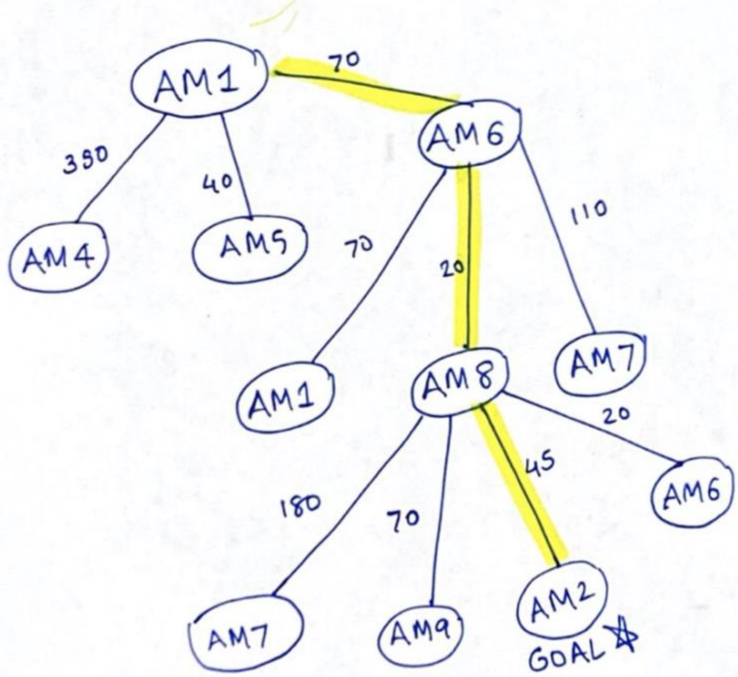
Step 4:-

Current Node : AM2

Is AM2 a Goal Node? Yes.

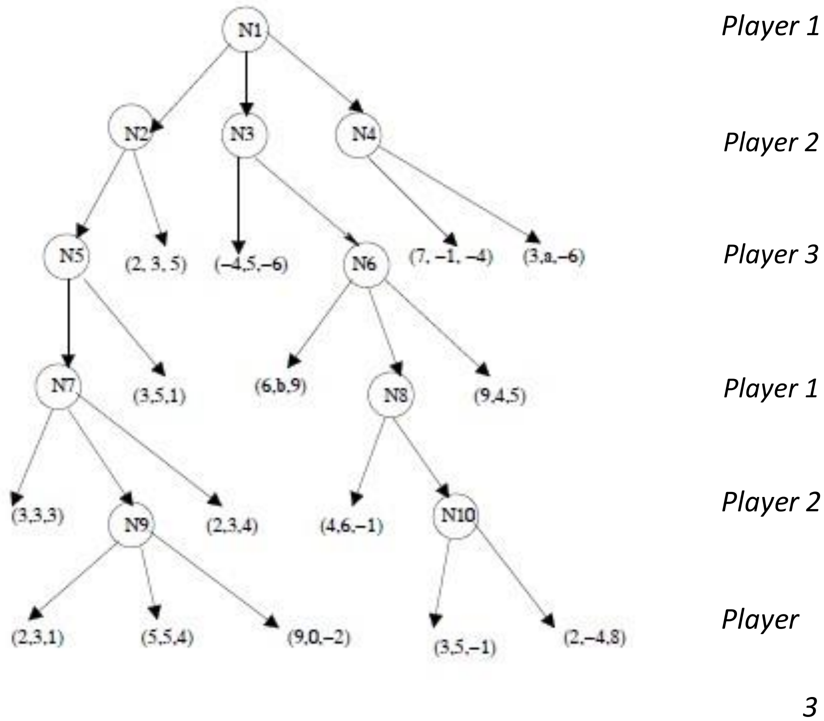
Return AM1 → AM6 → AM8 → AM2

Path cost: 135 (90 + 45)



Problem 2 (50 points)

(a) (**15 points**) Given the following game tree, find the possible values of variables *a* and *b* such that the minimax values in node N1 are (7, −1, −4). Also compute the minimax values at nodes: N2, N3, N4, N5, N6, N7, N8, N9 and N10.



Solution:

Lets start computing values for Minimax.

N10 = (3, 5, 1)

N9 = (5, 5, 4)

N8 = (4, 6, -1)

N7 = (5, 5, 4)

N5 = (5, 5, 4)

N6 = (6, b, 9)

N2 = (5, 5, 4)

*We know that we want the Minimax value at node N1 to be (7, -1, -4)
For N1 to have the above value (7, -1, -4) needs to be the value of N4. which means $a < -1$.
Only when a is less than -1 ~~the~~ N4 will pick (7, -1, -4) as it is trying to Maximize score of player 2.
$\therefore$ N4 = (7, -1, -4) & $a < -1$.

*At N3, player 2 is going to pick Max{(-4, 5, 6), (6, b, 9)}
We want the value at N1 to be (7, -1, -4), which we have at N4. As N1 will pick the Max between N2, N3, N4, 7 has to be the highest value for player 1 in between N2, N3, N4. Thus we can see that (7 > -4) & (7 > 6) thus whatever value b takes. N1 will pick (7, -1, -4) as 7 would be the highest.

Thus if $b > 5$

$N3 = (6, b, 9)$

and if $b < 5$,

$N3 = (-4, 5, -6)$

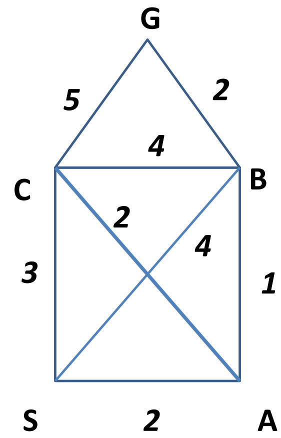
* $N1 = \text{Max}\{(5,5,4), (6,b,9), (7,-1,-4)\}$

or

$= \text{Max}\{(5,5,4), (-4,5,-6), (7,-1,-4)\}$

$= \underline{(7,-1,-4)}$ in both cases.

(b) (**15 points**) An agent stating in state *S* should reach the goal state *G*. If the possible states the agent can reach are *A, B, C* or *G*, as depicted bellow:

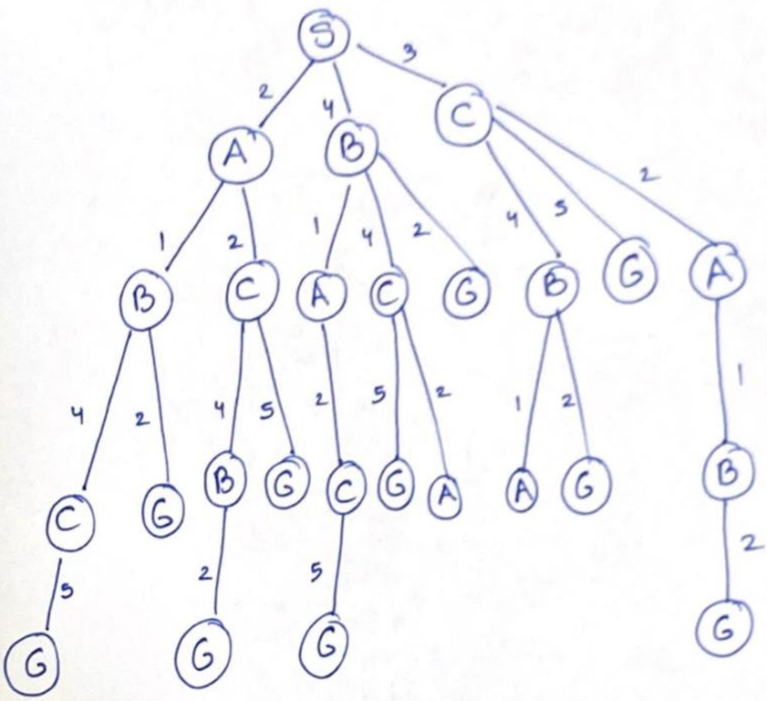


And as shown in the figure: cost(S→A) = 2; cost(S→B) = 4; cost(S→C) = 3; cost(A→B) = 1; cost(A→C) = 2; cost(B→C) = 4; cost(B→G) = 2; cost(C→G) = 5; you are asked to:

(a) draw the search tree that allows the agent to travel from *S* to *G*, knowing that the agent *cannot ever visit S again*, and cannot visit any state more than once. Show in the search tree all the ways in which the agent can get from the state *S* to the goal state *G*; (**5 points**) How many ways of getting to the goal state *G* from *S* are there? (**2 points**) *HINT:* Any solution path starts in *S* and ends in *G* but does not have to visit all other nodes! However, it cannot visit more than one any node!

(b) What is the least costly and the costliest way for the agent to get from state *S* to state *G*? Show the least costly path (**2 points**) and specify how much it costs (**2 points**). Show the costliest path (**2 points**) and specify how much it costs (**2 points**) and show how you have computed the costs.

Solution:



Total ways to reach goal = 10

1) S → A → B → C → G = 2+1+4+5 = 12

2) S → A → C → B → G = 2 + 2 + 4 + 2 = 10

3) S → A → B → G = 2+1+2 = 5

4) S → A → C → G = 2 + 2 + 5 = 9

5) S → B → A → C → G = 4 + 1 + 2 + 5 = 12

6) S → B → C → G = 4 + 4 + 5 = 13

7) S → B → G = 4 + 2 = 6

8) S → C → B → G = 3 + 4 + 2 = 9

9) S → C → G = 3 + 5 = 8

10) S → C → A → B → G = 3 + 2 + 1 + 2 = 8.

Least costliest path : $S \to A \to B \to G = 5$

$2 + 1 + 2 = 5$

Most costliest path : $S \to B \to C \to G = 13$

$4 + 4 + 5 = 13$

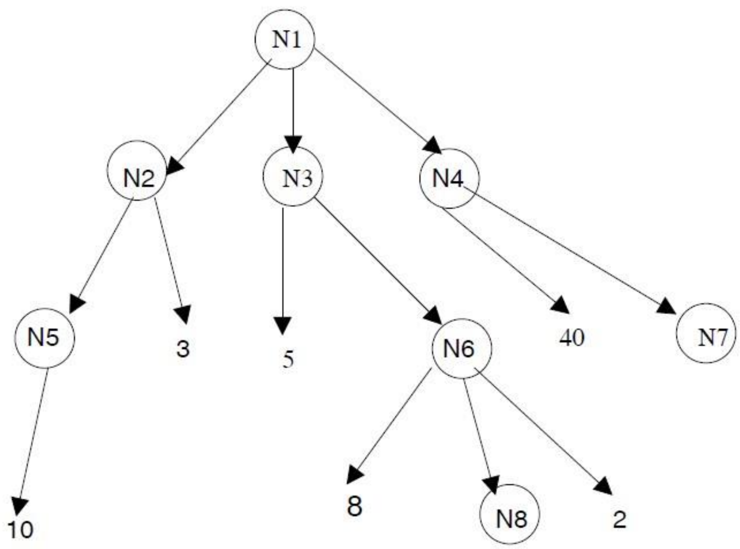
(c) (**20 points**) Given the game tree below, compute the value of alpha and beta at following nodes, if the order is the same as in depth-first search:

(1) alpha and beta at node N3 before and after visiting the terminal node with utility 5. Also show the values of alpha and beta in N3 after visiting N6 (Hint: Show also the values of alpha and beta at all nodes visited before you reached N3.); **(3 points)**

(2) alpha and beta at node N6; **(2 points)**

(3) alpha and beta at node N1 after visiting N7, if the node N7 has a child node with an utility value *x* (after you visited all nodes illustrated in the Figure) **(5 points)**

(4) should the game tree be pruned? If yes, how? **(10 points)**



MAX

MIN

MAX

MIN

Node	α	β	Value	Function
N1	$-\infty$	$+\infty$	$-\infty$	MAX VALUE.
N2	$-\infty$	$+\infty$	$+\infty$	MIN VALUE.
N5	$-\infty$	$+\infty$	$-\infty$	MAX VALUE.
10	10	10	10	
N5	10	$+\infty$	10	MAX VALUE.
N2	$-\infty$	10	10	MIN VALUE
3	3	3	3	
N2	$-\infty$	3	3	MIN VALUE.
N1	3	$+\infty$	3	MAX VALUE.
N3	3	$+\infty$	$+\infty$	MIN VALUE
5	5	5	5	
N3	3	5	5	MIN VALUE
N6	3	5	$-\infty$	MAX ~~MIN~~ VALUE.
8	8	8	8	
N6	3	5	8	MAX VAL. (PRUNE N8 & 2) Beta
N3	3	5	5	MIN VAL
N1	5	$+\infty$	5	MAX VAL.
N4	5	$+\infty$	$+\infty$	MIN VAL.
40	40	40	40	
N4	5	40	40	MIN VAL.

Node	α	β	value	function
N7	5	40	$-\infty$	MAX VALUE.

CASE 1: $x < 5$

x	x	x	x	
N7	x	40	x	MAX VALUE
N4	5	x	x	MIN VALUE
N1	5	5	5	MAX VALUE

CASE 2: $5 < x < 40$

x	x	x	x.	
N7	x	40	x	MAX VALUE
N4	5	x	x	MIN VALUE
N1	x	x	x	MAX VALUE

CASE 3: $40 < x$.

x	x	x	x	
N7	x	40	x	MAX VALUE
N4	5	40	40	MIN VALUE
N1	40	40	40	MAX VALUE.

Pruning

Yes. N8 & 2 will be pruned and it will be Beta Pruning As 8>5.