CS 480/580 | Introduction to Artificial Intelligence | 12/09/2021

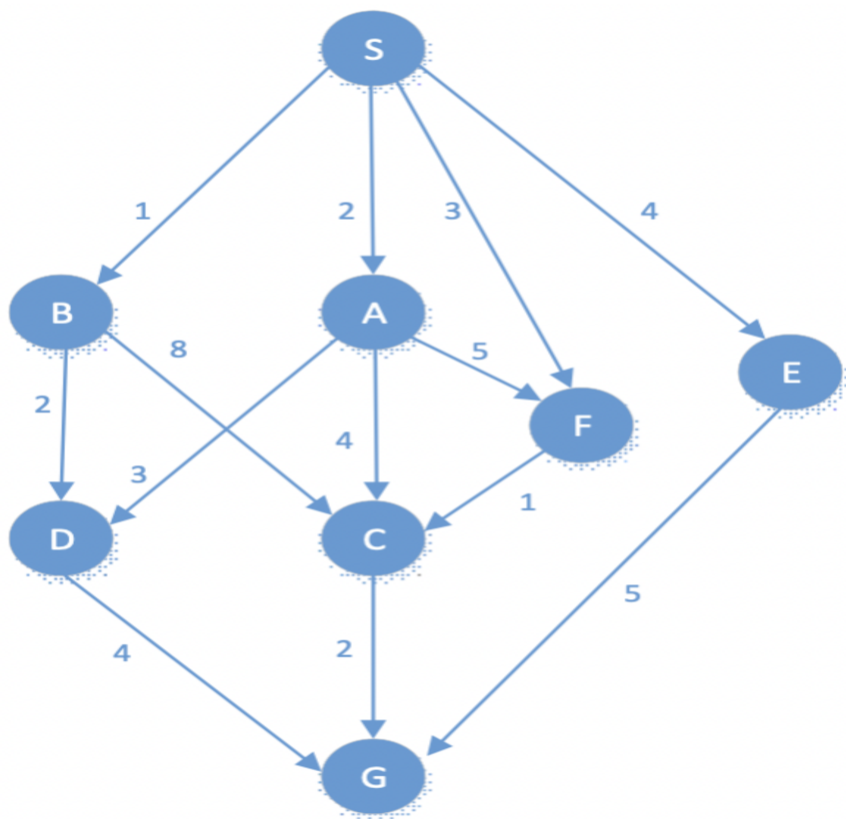
Introduction to Artificial Intelligence
Midterm Exam

Name: Nihaarika Aswatha Jagadish UIN: 01162729

Open book and notes. Please write your answers on the test sheet

Total: 100pts

1. Consider the search algorithm for the following graph with node S as the start node and node G as the goal node. The path cost value is showed next to the edge. The heuristic value is shown in the heuristic table. During search, if there are multiple nodes directly accessible from one node, the search always visits the one on the left first. If a new added node has the same cost as an existing one in the fringe, it is always put after the existing node. Also assume the algorithm do not eliminate duplicated nodes in the fringe (30 pts)



Heuristic

S	A	B	C	D	E	F	G
10	1	6	4	1	3	5	0

1) Simulate the behavior of Bread-First Search. Show the node visited and the nodes in the fringe at each step. Print out the final solution.

Answer:

Step 0:
Node Visited = null
Nodes in the fringe = (S)

Step 1:
Node Visited = S
Nodes in the fringe = (B,A,F,E)

Step 2:
Node Visited = B
Nodes in the fringe = (A,F,E,D,C)

Step 3:
Node Visited = A
Nodes in the fringe = (F,E,D,C,D,C,F)

Step 4:
Node Visited = F
Nodes in the fringe = (E,D,C,D,C,F,C)

Step 5:
Node Visited = E
Nodes in the fringe = (D,C,D,C,F,C,G)

Step 6:
Node visited = D
Nodes in the fringe = (C,D,C,F,C,G,G)

Step 7:
Node visited = C
Nodes in the fringe = (D,C,F,C,G,G,G)

Step 8:
Node visited = D
Nodes in the fringe = (C,F,C,G,G,G,G)

Step 9:
Node visited = C
Nodes in the fringe = (F,C,G,G,G,G,G)

Step 10:
Node visited = F
Nodes in the fringe = (C,G,G,G,G,G,C)

Step 11:
Node visited = C
Nodes in the fringe = (G,G,G,G,G,C,G)

Step 12:
Node visited = G
Goal state reached!

The goal state is reached in the 12th step.
The final solution is G → E → S

2) Simulate the behavior of Depth-First Search. Show the node visited and the nodes in the fringe at each step.

Answer:
Step 0:
Node visited = null
Nodes in the fringe = (S)

Step 1:
Node visited = S
Nodes in the fringe = (B,A,F,E)

Step 2:
Node visited = B
Nodes in the fringe = (D,C,A,F,E)

Step 3:
Node visited = D
Nodes in the fringe = (G, C,A,F,E)

Step 4:
Node visited = G
Goal state reached

The goal state is reach in the 4th step.
The final solution is G → D → B → S

3) Simulate the behavior of Uniform Cost Search. Show the node visited and the nodes in the fringe at each step.
Answer:

Step 0:
Node visited = null
Nodes in the fringe = [S(0)]

Step 1:
Node visited = S(0)
Nodes in the fringe = [B(1), A(2), F(3), E(4)]

Step 2:
Node visited = B(1)
Nodes in the fringe = [A(2), F(3), D(3), E(4), C(9)]

Step 3:
Node visited = A(2)
Nodes in the fringe = [F(3), D(3), E(4), D(5), C(6), F(7), C(9)]

Step 4:
Node visited = F(3)
Nodes in the fringe = [D(3), E(4), C(4), D(5), C(6), F(7), C(9)]

Step 5:
Node visited = D(3)
Nodes in the fringe = [E(4), C(4), D(5), C(6), F(7), G(7), C(9)]

Step 6:
Node visited = E(4)
Nodes in the fringe = [C(4), D(5), C(6), F(7), G(7), C(9), G(9)]

Step 7:
Node visited = C(4)
Nodes in the fringe = [D(5), C(6), G(6), F(7), G(7), C(9), G(9)]

Step 8:
Node visited = D(5)
Nodes in the fringe = [C(6), G(6), F(7), G(7), C(9), G(9), G(9)]

Step 9:
Node visited = C(6)
Nodes in the fringe = [G(6), F(7), G(7), G(7), G(8), C(9), G(9), G(9)]

Step 10:
Node visited = G(6)
Goal state reached

4) Simulate the behavior of Greedy Best-First Search. Show the node visited and the nodes in the fringe at each step.
Answer:

Step 0:
Node Visited = null
Nodes in the fringe = [S(10)]

Step 1:
Node Visited = S(10)
Nodes in the fringe = [A(1), E(3), F(5), B(6)]

Step 2:
Node Visited = A(1)
Nodes in the fringe = [D(1), E(3), C(4), F(5), F(5), B(6)]

Step 3:
Node Visited = D(1)
Nodes in the fringe = [G(0), E(3), C(4), F(5), F(5), B(6)]

Step 4:
Node Visited = G
Goal state reached!

5) Simulate the behavior of A* Search. Show the node visited and the nodes in the fringe at each step.

Answer:
Step 0:
Node visited = null
Nodes in the fringe = [S(10)]

Step 1:
Node visited = S(10)
Nodes in the fringe = [A(3), B(7),E(7), F(8)]

Step 2:
Node visited = A(3)
Nodes in the fringe = [D(6),B(7),E(7), F(8), C(10), F(12)]

Step 3:
Node visited = D(6)
Nodes in the fringe = [B(7),E(7),F(8), G(9), C(10), F(12)]

Step 4:
Node visited = B(7)
Nodes in the fringe = [D(4),E(7),F(8), G(9), C(10), F(12), C(13)]

Step 5:
Node visited = D(4)
Nodes in the fringe = [E(7),G(7),F(8), G(9), C(10), F(12), C(13)]

Step 6:
Node visited = E(7)
Nodes in the fringe = [G(7),F(8), G(9), G(9), C(10), F(12), C(13)]

Step 7:
Node visited = G(7)
Goal state reached!

6) Justify your answer on question 5). Does A* search provide the optimal solution? Why or why not?

Answer: Analysing all the solutions for the given question, we can conclude that A* does not provide the optimal solution in this case as there is another method,i.e, Depth First Search that reaches the solution state in the 4th step whereas A* reaches the goal state in the 7th step. Hence, we can consider DFS to be the optimal solution and not A* in this particular question.

2. (20 pts) Consider using first-order logic to describe the kinship. Assuming the following relations and functions are given.
Father(x, y): True if x is the father of y.
Mother(x, y): True if x is the mother of y.
Parent(x, y): True if x is the parent of y.
Male(x): True if x is male.
Sibling(x, y): True if x is the sibling of y.
Cousin(x, y): True if x is the first cousin of y.
Child(x, y): True if x is y's child.
Nephew(x, y): True if x is y's nephew.

1). One’s father is one’s male parent.
Answer:
$\forall$ f, c $\rightarrow$ Father(f, c) $\rightarrow$ Male(f) ^ Parent(f, c)

Where f stands for father and c stands for child.

2). Everybody has a father.
Answer:
$\forall$ c, $\exists$ f $\rightarrow$ Father(f, c)

Where f stands for father and c stands for child.

3). Whoever has a father, has a mother.
Answer:
[$\exists$ f $\rightarrow$ Father(f, c)] $\rightarrow$ [$\exists$ m $\rightarrow$ Mother(m, c)]

Where f stands for father, m stands for mother and c stands for child.

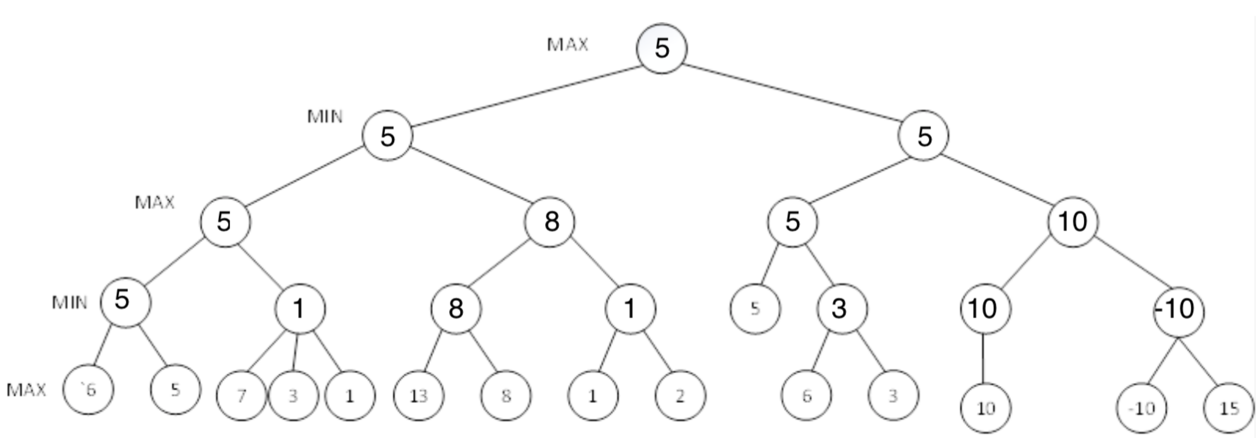
4). A nephew is a sibling’s male child.
Answer: $\forall$ a, b $\rightarrow$ [Nephew(a, b) $\longleftrightarrow$ $\exists$ c, [Sibling(b, c) ^ Child(a ,c) ^ Male(a)]]

5). A cousin is a child of a parent’s sibling.
Answer:
$\forall$ a, b Cousin(a, b) $\rightarrow$ $\exists$ p, ps Parent(p, b) ^ Sibling(ps, p) ^ Parent(ps, a)

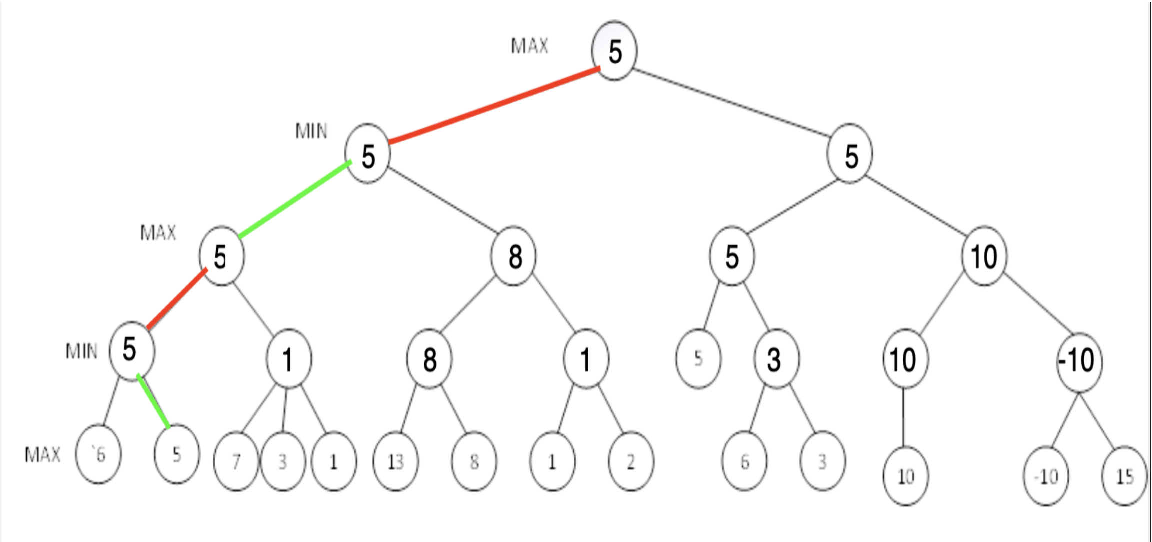
where p is parent, ps is sibling of parent, p is the parent of b and ps is the parent of a.

3. For the tree below representing a minimax game use the minimax algorithm with alpha-beta pruning to determine the optimal moves to get the highest possible score if MAX and MIN make perfect moves.

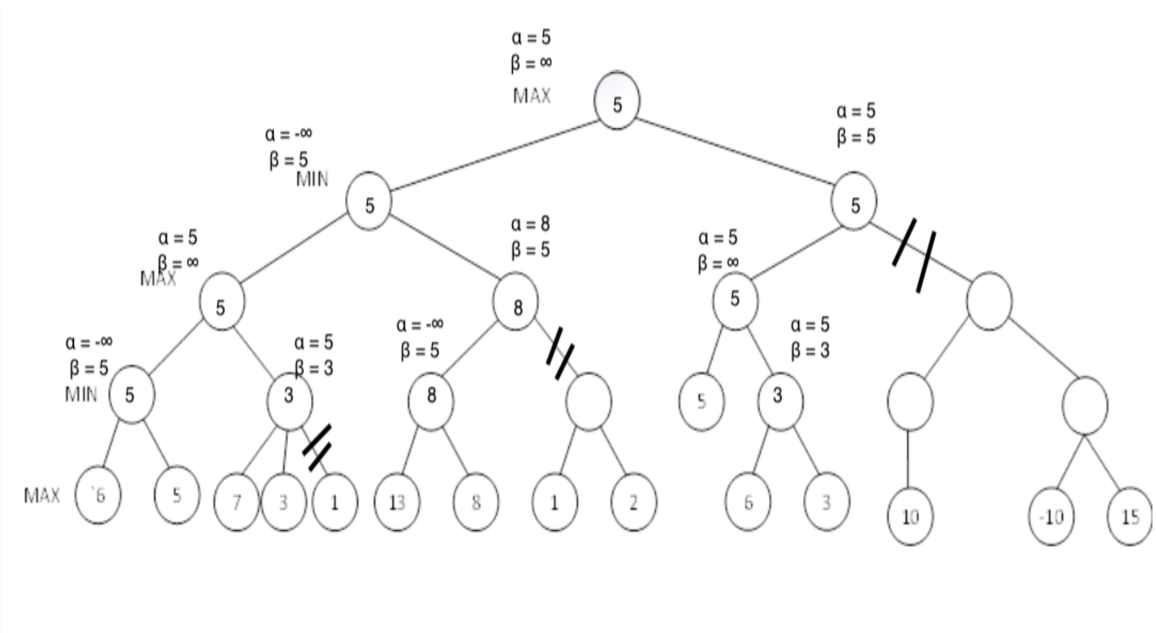
1) Show your work by labeling the nodes with the values of minimax algorithm. (5 pts)
Answer:



2) Highlight the optimal move on the game tree. (5 pts)
Answer:



3) Cross out all the pruned branches in the above game tree using the alpha-beta pruning algorithm. Show the alpha and beta values at every step. (10 pts)
Answer:



4. (10 pts) Let $\alpha = A \vee B$ and KB = { $A \vee C \vee D, A \vee \neg C \vee D$ }

Is it the case that KB $\models \alpha$? Justify your answer with your method.

Answer:

A	B	C	D	A v C v D	A v !C v D	KB	α
T	T	T	T	T	T	T	T
T	T	T	F	T	T	T	T
T	T	F	T	T	T	T	T
T	T	F	F	T	T	T	T
T	F	T	T	T	T	T	T
T	F	T	F	T	T	T	T
T	F	F	T	T	T	T	T
T	F	F	F	T	T	T	T
F	T	T	T	T	T	T	T
F	T	T	F	T	F	F	T
F	T	F	T	T	T	T	T
F	T	F	F	F	T	F	T
F	F	T	T	T	T	T	F
F	F	T	F	T	F	F	F
F	F	F	T	T	T	T	F
F	F	F	F	F	T	F	F

From the above table we can conclude that not all the rows in KB column are same as the values in α column. Thus KB$\models$α is not true in all cases.

5. (20 pts) Assuming that you build a neural network for Spam detection and you generate the following confusion matrix:

	Actual Spam (Positive)	Actual Non-Spam (Negative)
Predicted Spam (Positive)	9	1
Predicted Non-Spam (Negative)	52	729

Answer:
From the above table we know:
TP = 9
TN = 729
FP = 1
FN = 52

1) Calculate the Precision.
Precision = TP/(TP+FP)
= 9 /(9+1)
= 9/ 10
= 0.9

2) Calculate the Recall.
Recall = TP/(TP+FN)
= 9 /(9+52)
= 9 / 61
= 0.1475

3) Calculate the Accuracy.
Accuracy = Correct predictions / Total Prodictions
= (TP + TN)/(TP+TN+FP+FN)
= (9+729)/(9+1+52+729)
= 738/791
=0.933

4) Is your neural network a good Spam detector? Pleae justify your answer.
Answer:
From the above calculations, we can notice that the accuracy is about 93% which means that out of 100 emails, almost 93 emails will be classified correctly as spam or not spam. A 93% accuracy may be good or bad depending on the user. Let's say an user receives only 10 emails each day then the chances of him getting a span email in his inbox is 0.7 which is not even a single email. So in that case it can be considered a good spam detector. But in cases where the user receives 100 emails each day then the number of emails that are spam and received in his inbox are 7 which is actually a big number and may frustrate the user. So in such a case, this is not a good spam detector.

Another measure for the goodness of the model is the precision which is 0.9. Which means that for every 100 emails 10 emails which are not spams are also considered as spam emails and are moved to the spam folder and the user misses the email. So, it is not really a good spam detector. The other good measure is to calculate the F-measure which can be calculated as (2*Recall*Precision)/(Recall + Precision). From the formula we get the F Measure to be 0.2527 which is not a good score. Thus, from all this we can conclude that it is not a good spam detector.