1

Date:

FINAL EXAM 2020
(AI)

17k-3730
Sec C
Eisha Tir Raazia.

QUESTION # 01

(A)

Digit	A	B	
1	8	5	
7	4	8	
3	2	4	$\rightarrow C_1$
7	4	8	$\rightarrow C_2$
3	2	4	
0	3	3	

Assumption
C_1 = S/U policy.
C_2 = grade.

let's take digit 3 and 7 as random seed.

	A	B	(2,4)	(4,8)
1	8	5	6.08	5 ✓
7	4	8	4.47	0 ✓
3	2	4	0 ✓	4.47
7	4	8	4.47	0 ✓
3	2	4	0	4.47
0	3	3	1.4 ✓	5.09

euclidean distance
$\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$

$C_1 = (3,3,0)$ $(1,7,7) = C_2$

$C_1 = \left(\frac{2+2+3}{3}, \frac{4,4,3}{3}\right)$ $C_2 = \left(\frac{8+4+4}{3}, \frac{5+8+8}{3}\right)$

$= (2.33, 3.66)$ $= (5.33, 7)$

(2)

Date:

	A	B	C_1 (2.33, 3.66)	C_2 (5.33, 7)
1	8	5	5.82	3.336 ✓
7	4	8	4.65	1.664 ✓
3	2	4	0.47 ✓	4.48
7	4	8	4.65	1.664 ✓
3	2	4	0.47 ✓	4.48
0	3	3	0.94 ✓	4.629

$C_1 = \{3, 3, 0\}$ $C_2 = \{1, 7, 7\}$ → same clusters as ~~pred~~ previous iteration

S/U policy = 3, 3, 0
Letter grad = 1, 7, 7

(B)

		$D_2(1,1)$	$D_3(4,1)$
D_1	(2,1)	1	2
D_2	(1,1)	0	3
D_3	(4,1)	3	0
D_4	(1,2)	1	3.16
D_5	(2,2)	1.41	2.33
D_6	(4,2)	3.16	1

$C_1 = \{D_1, D_2, D_4, D_5\} = \frac{2+1+1+2}{4}, \frac{1+1+2+2}{4}$
$= (1.5, 1.5)$

$C_2 = \{D_3, D_6\} = \left(\frac{4+4}{2}, \frac{1+2}{2}\right)$
$= (4, 1.5)$

Iteration 2 :

		C_1 (1.5, 1.5)	C_2 (4, 1.5)
D_1	(2,1)	0.70	2.06
D_2	(1,1)	0.7	3.04
D_3	(4,1)	2.54	0.5
D_4	(1,2)	0.7	3.04
D_5	(2,2)	0.7	2.06
D_6	(4,2)	2.54	0.5

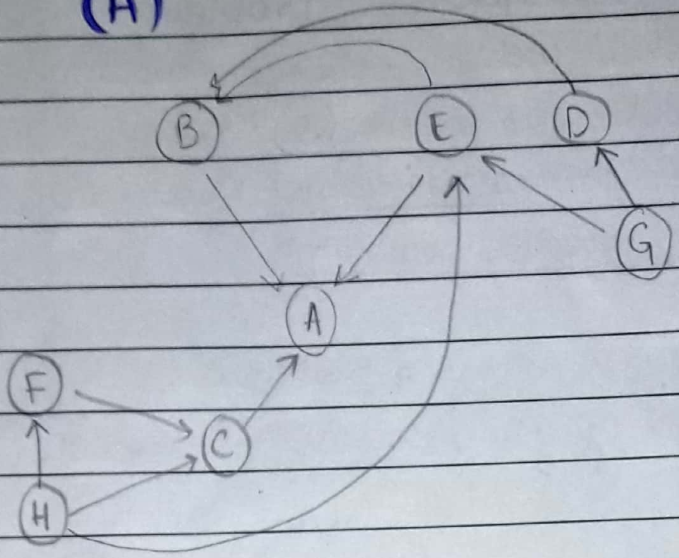
$C_1 = \{D_1, D_2, D_4, D_5\}$
$C_2 = \{D_3, D_6\}$
same as iteration 1

Ans.

Date:

QUESTION #02

(A)



(B)

$$P(OS|Pass \mid ITC = Pass) = \frac{P(OS \wedge ITC)}{P(ITC = Pass)}$$

$$= \frac{P(OS = Pass|DS) \times P(DS|OOP, ITC = Pass)}{P(ITC = pass)}$$

$P(ITC = Pass) = 0.6$

ITC	OS	DS	DOP	P(ITC)	P(DS=P | DOP,ITC=P)	P(OS=P | DS)	P(DOP)	
P	P	P	P	0.6	0.2	0.3	0.4	= 0.0144
P	P	P	F	0.6	0.3	0.3	0.6	= 0.0324
P	P	F	P	0.6	0.8	0.8	0.4	= 0.1536
P	P	F	F	0.6	0.7	0.8	0.6	= 0.2016
								0.402.

So $P(OS = Pass \mid ITC = Pass) = \frac{0.402}{0.6}$

$= 0.67$ Ans.

Date:

QUESTION #03
(A)

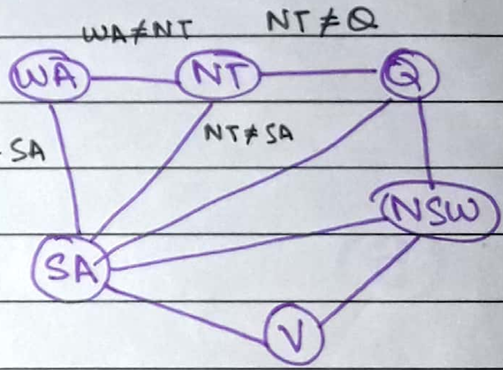
CSP (Constraint Satisfaction Problem)

It's a powerful problem solving paradigm which views problem as a set of variables to which we have to assign values that satisfy a no. of problem specific constraints.

★ Example 1 → graph colouring problem

Contraint graph : simplifies the search

i.e



Variables: {WA, NT, Q, NSW, V, SA}

Constraints: WA ≠ NT
NT ≠ Q
NT ≠ SA
SA ≠ WA

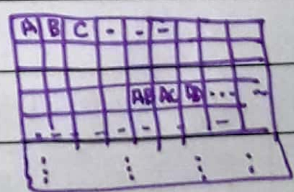
Domain : {R, G, B}

★ Example 2 → Sudoku

Variables → In case of 9x9 box, there would be 81 variables :

~~Constraints~~

~~Domain~~



Constraint : Row, column and specific grid should have unique domain values.

Domain : {1, 2, 3, ---, 9}

(5)

Date:

★ Example 3 : N-Queen problem.

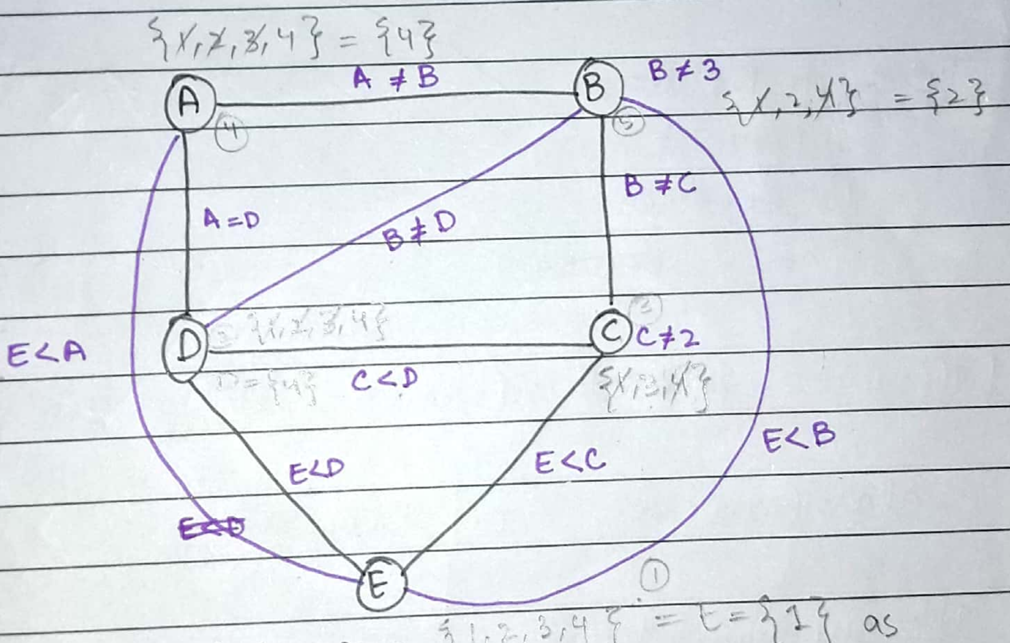
variables : $Q_1, Q_2, \dots Q_N$
(for queen 1–N).

Domain : 1–N^2 options. i.e: for 4 Queens we'll have 16 boxes as domain (for 4-queen: each queen has 1 row and 4 columns positions as it's domain)

Constraint : Queen isn't placed in diagonal to each other

→ for Q_i, Q_j $i \neq j$ (don't place Q_i and Q_j in same row and same column).

(B)



Initial domains : {1, 2, 3, 4} = E = {1} as

A = {1, 2, 3, 4}

B = {1, 2, 4} as B ≠ 3

C = {1, 3, 4} as C ≠ 2

E = D = {1, 2, 3, 4}

So

domain of E will be {1} as E < A, B, C, D

D = {4} as D > C, E so C will not have 4 and 1 and domain will shrink to {3}.

as, A = D so A = {4} and finally, B = {2}

sp

Date:

final domains :

$A = D = \{4\}$
$C = \{3\}$
$B = \{2\}$
$E = \{1\}$

QUESTION # 04

(A)

The field of AI through which ~~our~~ ~~machi~~ we help our algorithms to learn and improve from a large amount of data. Our algorithms also ~~to~~ improve by finding the different data patterns from data using different statistical techniques.

Types of ML are:
- Supervised
- Unsupervised
- Semi-Supervised

Difference b/w classification & clustering.

classification	Clustering
→ It's supervised learning technique whose goal is to classify the future observations.	→ It's unsupervised technique whose goal is to find similarities within a given dataset.
→ Labelled data as input	→ Unlabelled data as input.

Date:

(B)

Entropy of class label:

	covid (+)	not covid (-ve)
	3	07.

$$E = \sum -p_i(\log_2 p_i) = -\frac{3}{10}\log_2\left(\frac{3}{10}\right) - \frac{7}{10}\log_2\left(\frac{7}{10}\right)$$

$$= (0.52 + 0.36)$$

Entropy of class labels = 0.88.

Information gain of attributes:

IG = E(parent) − [weighted avg. × E(child)]

(i) $IG(\text{Headache}) = 0.88 - \left[\frac{5}{10} \times 0.96 + \frac{5}{10} \times 0.71\right]$

$= 0.04$

Yes	No
[2+ve, 3−ve]	[1+ve, 4−ve]
$\Rightarrow -\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\left(\log_2 \frac{2}{5}\right)$	$\Rightarrow -\frac{1}{5}\log_2\left(\frac{1}{5}\right) - \frac{4}{5}\log_2\left(\frac{4}{5}\right)$
$\Rightarrow 0.52 + 0.44$	$\Rightarrow 0.46 + 0.25$
$= 0.96$	$= 0.71$

(ii) $IG(\text{cough}) = 0.88 - \left[\frac{5}{10} \times 0.71 + \frac{5}{10} \times 0.96\right]$

$= 0.04$

Yes	No
[1+ve, 4−ve]	[2+ve, 3−ve]
$= -\frac{1}{5}\log_2\frac{1}{5} - \frac{4}{5}\log_2\frac{4}{5}$	$= -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\left(\frac{3}{5}\right)$
$= 0.71$	$= 0.96.$

sp

Date:

(iii) $IG(\text{fever}) = 0.88 - \left[\frac{3}{10} \times 0 + \frac{7}{10} \times 0\right]$

$= 0.88$

Yes	No
[3+ve, 0 -ve]	[0+ve, 7-ve]
= 0	= 0

(iv) $IG(\text{flu}) = 0.88 - \left[\frac{8}{10} \times 0.54 + \frac{2}{10} \times 0\right]$

$= 0.44$

Yes	No
[1+ve, 7-ve]	[2+ve, 0-ve]
$= -\frac{1}{8} \log_2 \frac{1}{8} - \frac{7}{8} \log_2\left(\frac{7}{8}\right)$	$= 0$
$= 0.37 + 0.16$	
$= 0.54$	

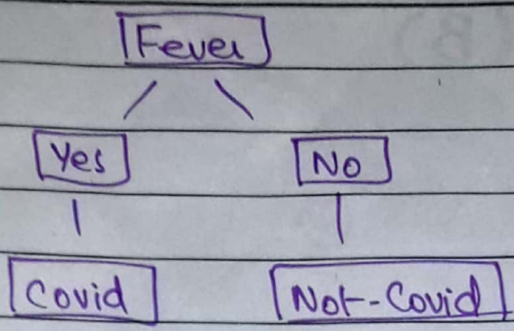
(v) $IG(\text{pain}) = 0.88 - \left[\frac{5}{10} \times 0.71 + \frac{5}{10} \times 0.96\right]$

$= 0.04$

Yes	No
[1+ve, 4-ve]	[2+ve, 3-ve]
$= -\frac{1}{5} \log_2\left(\frac{1}{5}\right) - \frac{4}{5} \log_2\left(\frac{4}{5}\right)$	$= -\frac{2}{5} \log_2\left(\frac{2}{5}\right) - \frac{3}{5} \log_2\left(\frac{3}{5}\right)$
$= 0.71$	$= 0.96$

The highest 'Information Gain' we get, is from 'fever' so, fever is the root node

sp



Dead end reached so we'll stop.

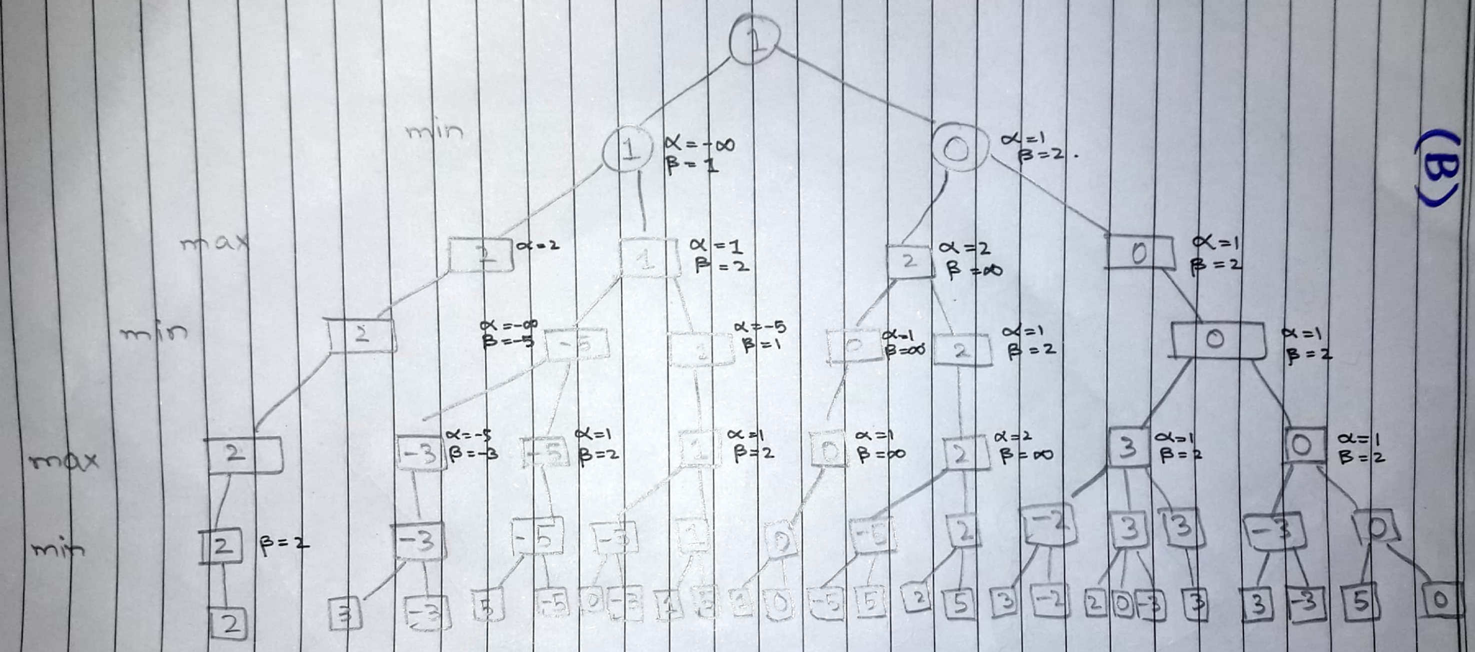
QUESTION # 05

(A)

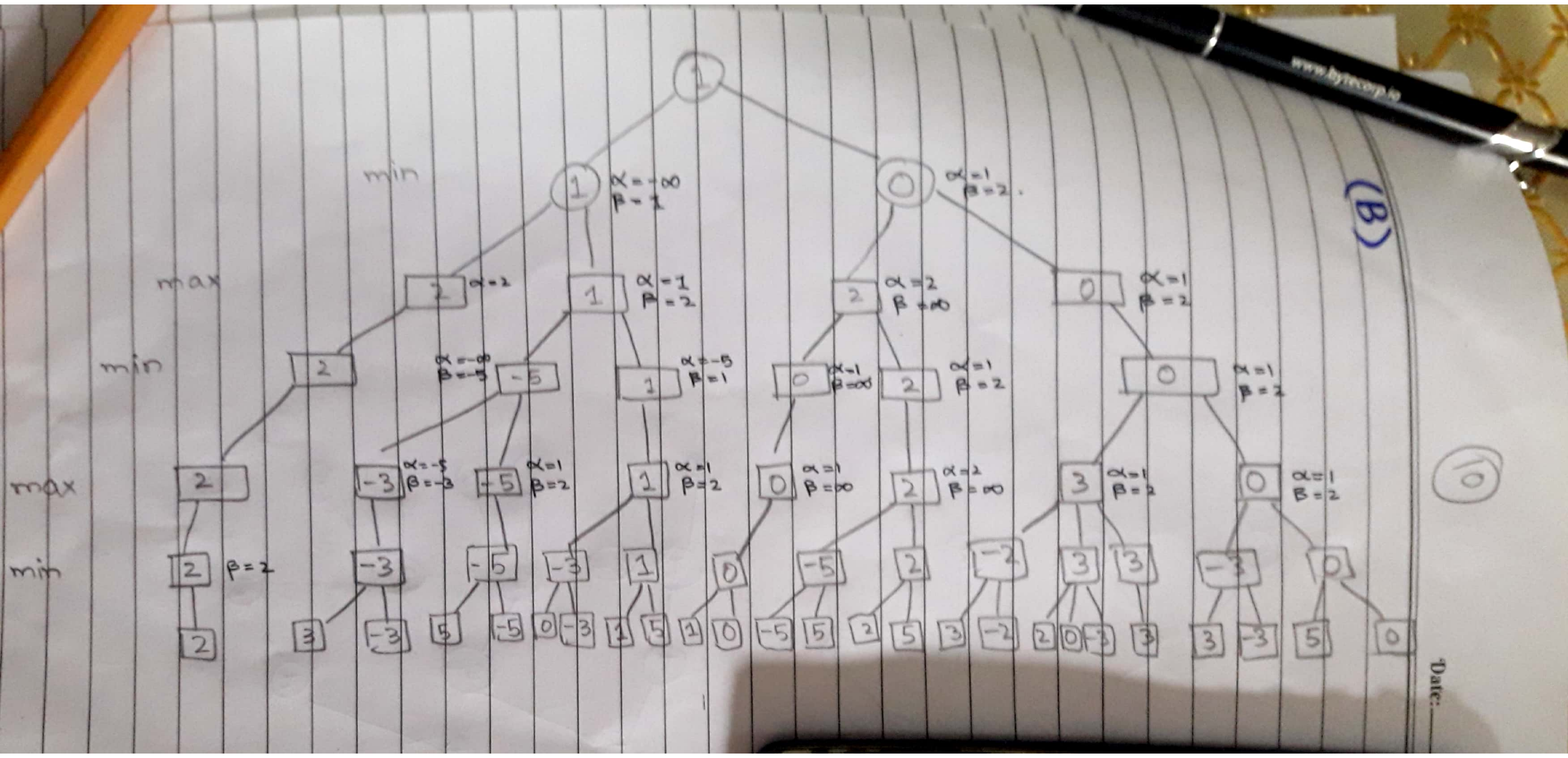
Yes, BFS can be used instead of minmax in a gameplay to find all the game states that can be obtained in upto x moves from some starting game state using a queue.

- Using BFS, all possible sequence of N moves can be easily figured out from starting position and situation after each sequence of N moves.
- In a BFS all reachable N moves can be easily found. The game states are reachable in N moves and will be at Nth-level of game state tree rooted at 's'(state). → As BFS do level wise traversal.

* BFS can be used but there are some drawbacks i.e:
  → we have to store all the nodes (level wise) and their children nodes until you reach terminating search depth - Consuming more memory as compared to minmax/DFS which require linear amount of memory.



(B)
10
min
max
min
max
min
α=-∞
β=1
α=1
β=2.
α=2
α=1
β=2
α=2
β=∞
α=1
β=2
α=-∞
β=-5
α=-5
β=1
α=1
β=∞
α=1
β=2
α=1
β=2
α=-5
β=-3
α=1
β=2
α=1
β=2
α=1
β=∞
α=2
β=∞
α=1
β=2
α=1
β=2
β=2


min
max
min
max
min
(B)
Date: