University of Asia Pacific

Admit Card

Final-Term Examination of Spring, 2020

Financial Clearance	PAID

Registration No : 17101007

Student Name : Mahnaz Rafia Islam

Program : Bachelor of Science in Computer Science and Engineering

SI.NO.	COURSE CODE	COURSE TITLE	CR.HR.	EXAM. SCHEDULE
1	CSE 400	Project / Thesis	3.00	
2	CSE 401	Mathematics for computer Science	3.00	
3	CSE 403	Artificial Intelligence and Expert Systems	3.00	
4	CSE 404	Artificial Intelligence and Expert Systems Lab	1.50	
5	CSE 405	Operating Systems	3.00	
6	CSE 406	Operating Systems Lab	1.50	
7	CSE 407	ICTLaw, Policy and Ethics	2.00	
8	CSE 410	Software Development	1.50	
9	CSE 427	Topics of Current Interest	3.00	

Total Credit: 21.50

1. Examinees are not allowed to enter the examination hall after 30 minutes of commencement of examination for mid semester examinations and 60 minutes for semester final examinations.

2. No examinees shall be allowed to submit their answer scripts before 50% of the allocated time of examination has elapsed.

3. No examinees would be allowed to go to washroom within the first 60 minutes of final examinations.

**4. No student will be allowed to carry any books, bags, extra paper or cellular phone or objectionable items/incriminating paper in the examination hall.
Violators will be subjects to disciplinary action.**

This is a system generated Admit Card. No signature is required.

Admit Card Generation Time: 26-Oct-2020 05:55 PM

University of Asia Pacific
Department of Computer Science and Engineering

Final Term Examination: Spring-2020

Name: Mahnaz Rafia Isam			Registration No: 17101007
Roll No: 07	Year: 4th	Semester: 1st	Course Code: CSE 403
Course Title: Artificial Intelligence and Expert Systems			Date: 05.11.2020

Answer to the question no: 1

Here,

My Id is 17101007

$T_1 = (07 \% 3) + 2 = 1 + 2 = 3$

$T_2 = -7$

$T_3 = -10$

$T_4 = (07 \% 4) + 1 = 3 + 1 = 4$

$T_5 = T_1 + 5 = 3 + 5 = 8$

$T_6 = -(T4 + 1) = -(4 + 1) = -5$

In this game tree,

□ → Max

○ → Min

△ → Terminal state.

The min-max algorithm is the best for this problem and the required condition for pruning

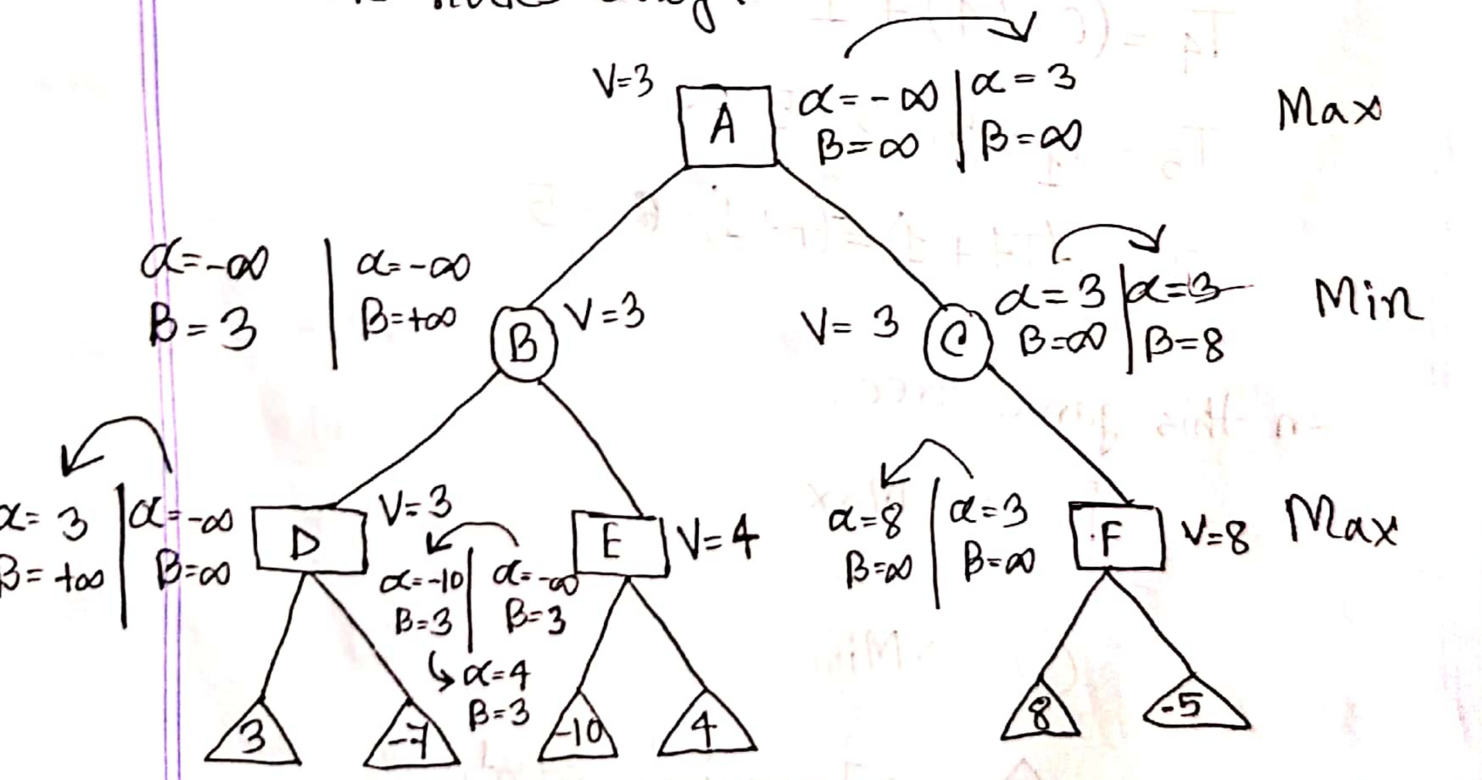
Alpha-Beta Pruning Condition: $\alpha \geqslant \beta$

→ The Max player only update value of alpha.

→ The Min will update beta.

→ In backtracking, the node values will pass to upper nodes insted of value of alpha-beta.

→ The alpha beta value will be passed to the child nodes only.



Step-1 Max will start first move from node A where $\alpha = -\infty$, $\beta = \infty$, these value of α and β passed to node B where again $\alpha = -\infty$ and $\beta = +\infty$ and node B passes the same value to its child node D.

Step-2: At node D, it's turn, so the value of α will be compared with firstly 3, then -7 and then $\max(3,-7) = 3$ will be the value of α at node D and node value will also be 3.

Step-3: Now, backtrack to node B. Here, $\beta = +\alpha$ will compare with the available subsequent nodes value i.e $\min(\infty, 3) = 3$, so at node B the value of $\alpha = -\infty$ and $\beta = 3$

Step-4: Now next successor is Node E and the value of $\alpha = -\infty$ and $\beta = 3$ will also be passed to node E.

At node E, it's max turn and current value of α will be compared with left child (-10). so max

$(-\infty,-10)=-10$. so at E node $\alpha=-10$ and $\beta=3$. Then compare with right child 4, so $\max(-10,4)$ so, at E node $\alpha=4$ and $\beta=3$. Here, $\alpha \geq \beta$ but there is no successor of E to be pruned.

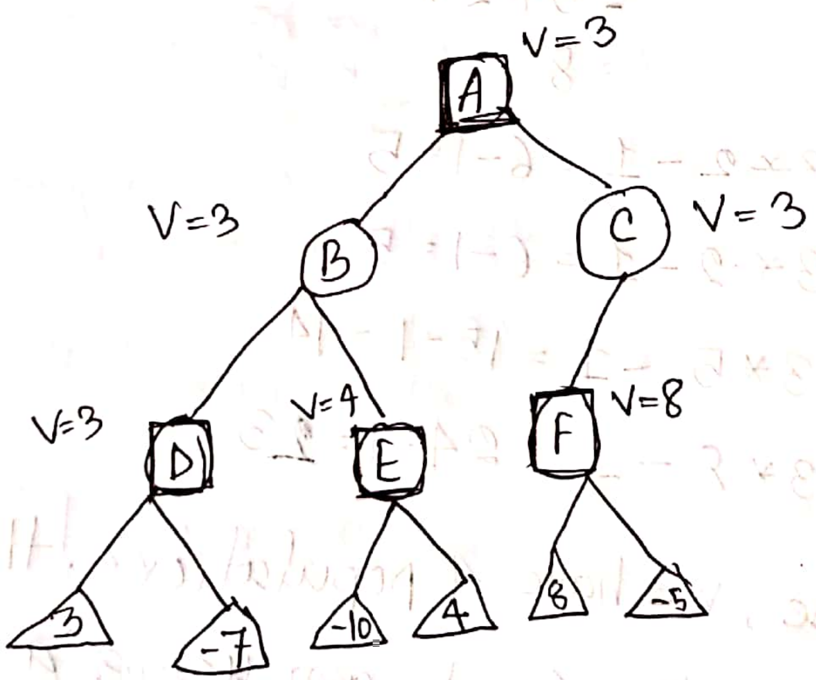
Step-5: Now again backtrack node B to to node A. the value of α will be change ~~max(-∞,~~ $\min(-\infty,3)=3$ and $\beta=$ ~~+∞~~ $3, \alpha=-\infty$. these two value now passes to right successor of A that is to node C.

At node C $\alpha=3$ and $\beta=+\infty$ and the values will be passed to node F.

Step 6: At node F again the value of α will be compared with left child (8) and $\max(8,3)=8$. Here ~~the node value will also be 8~~ then compared with right child -5. Still $\max(8,-5)=8$. Still α remains 8. and the node value is also 8.

Step-7: Now, F returns the value 8 to node C. At C ~~$\alpha=8$ and $\beta=+\infty$~~. $\alpha=3$ and $\beta=8$.

Step-8: Now, C return the value of 3 to A.
Here, A is max so max $(3,3)=3$.
The final game tree is showing the nodes which are computed and there is nothing to be never computed.
Thus, the optimal value for the maximizer is 3 for this example.



Answer to the question no: 2

Given, $f(x) = 3x-1$, Here, 4 population.

My Age is 22.

Population $A = 2$

$B = 2$

$C = (2+2)+1 = 5$

$D = C + Min(A, B) + 1$

$= 5 + Min(2,2) + 1$

$= 5+2+1$

$= 8$

$\therefore f(A) = 3*2-1 = 6-1 = 5$

$f(B) = 3*2-1 = 6-1 = 5$

$f(C) = 3*5-1 = 15-1 = 14$

$f(D) = 3*8-1 = 24-1 = 23$

In this case, we have 4 populations, the range of x is 2~10. So I am using a 4-bit representation.

$f(x) = 3x-1$: Selection

String No.	Initial Population	x Value	Fitness $f(x)=3x-1$	Probability (i)	Expected Count	Actual Count
1(A)	0010	2	5	0.11	0.44	0
2(B)	0010	2	5	0.11	0.44	1 (Taking Ceiling)
3(C)	0101	5	14	0.29	1.16	1
4(D)	1000	8	23	0.49	1.96	2
		Sum →	47	1	4.00	4
		Average →	11.75	0.25	1.00	1
		Max →	23	0.49	1.96	2

Here, I am taking Ceilling value for the Actual Count of B to make sure that the total no. of population is 4.

Calculation of Probability: $\left(\frac{\text{fitness Value}}{\text{Sum of fitness}}\right)$

$P(A) = \frac{5}{47} = 0.11$

$P(B) = \frac{5}{47} = 0.11$

$P(C) = \frac{14}{47} = 0.29$

$P(D) = \frac{23}{47} = 0.49$

Calculation of Expected Count: (Probability $\times$ Total no. of population)

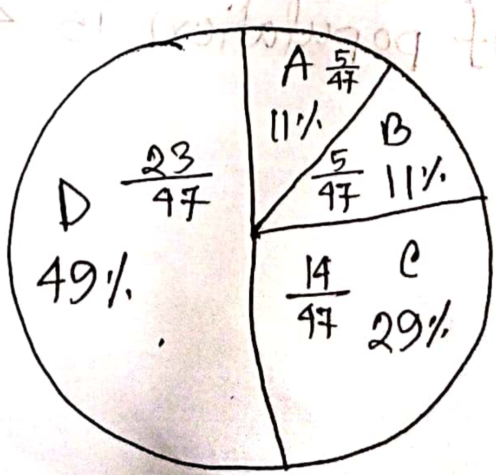
$0.11 \times 4 = 0.44$

$0.11 \times 4 = 0.44$

$0.29 \times 4 = 1.16$

$0.49 \times 4 = 1.96$

Roulette Wheel:



$f(x) = 3x-1$: Crossover:

String No.	Mating Pool	Crossover point	Offspring after crossover	x value	Fitness $f(x) = 3x-1$
2(B)	001|0	3	0010	2	5
4(D)	100|0	3	1000	8	23
3(C)	01|01	2	0100	4	11
4(D)	10|00	2	1001	9	26
				Sum →	65
				Average →	16.25
				Max →	26

$f(x) = 3x-1$: Mutation:

String No.	Offspring after crossover	Offspring after mutation	x Value	Fitness $f(x) = 3x-1$
2(B)	$\boxed{0}010$	$\boxed{1}010$	10	29
4(D)	1000	1000	8	23
3(C)	$01\boxed{0}0$	$01\boxed{1}0$	6	17
4(D)	1001	1001	9	26
			Sum →	95
			Average →	23.75
			Max →	29

Answer to the question no: 3(a)

The probability of raining on sunday = 0.7
" " not raining on sunday (sunny) = 1-0.7 = 0.3

Here,
Last two digit of my id = 07

$B = 07/100 = 0.07$

$A = 1-B = 1-0.07 = 0.93$

~~$C = \sqrt{07}/10 = 0.26$~~

~~$D = 1-C = 1-0.26 = 0.74$~~

$C = \sqrt{27}/10 = 0.52$

$D = 1-C = 1-0.52 = 0.48$

Sunday	Next day	Probability Distribution
rain	rain	0.93
rain	sun	0.07
sun	rain	0.52
sun	sun	0.48

Let's assume,

X_1 = sunday
X_2 = monday
X_3 = tuesday

$$P(X_2=rain) = P(X_2=rain \mid X_1=sun) \cdot P(X_1=sun) + P(X_2=rain \mid X_1=rain) \cdot P(X_1=rain)$$

$$= 0.52 \times 0.3 + 0.93 \times 0.7$$

$$= 0.156 + 0.651 = 0.807$$

$$\therefore P(X_2=sun) = 1 - P(X_2=rain)$$

$$= 1 - 0.807$$

$$= 0.193$$

$$P(X_3=rain) = P(X_3=rain \mid X_2=sun) \cdot P(X_2=sun) + P(X_3=rain \mid X_2=rain) \cdot P(X_2=rain)$$

$$= 0.52 \times 0.193 + 0.93 \times 0.807$$

$$= 0.10036 + 0.75051$$

$$= 0.85087$$

So the probability of raining on Tuesday is 0.85.

Answer to the question no: 3 (b)

Here,

query variable → cavity.

evidence variable → ¬toothache.

hidden variable → catch.

Last two digit of my id → 07

$A = 07/100 = 0.07$

$B = 1 - A = 1 - 0.07 = 0.93$

$C = 0.576$

$D = 0.144$

$$P(\text{Cavity} \mid \neg \text{toothache}) = \frac{P(\text{Cavity} \land \neg \text{toothache})}{P(\neg \text{toothache})}$$

$$= \frac{A+B}{A+B+C+D}$$

$$= \frac{0.07 + 0.93}{0.07 + 0.93 + 0.576 + 0.144}$$

$$= \frac{1}{1.72}$$

$$= 0.58$$

Answer to the question no: 4(a)

Perceptron is the simplest form of a neural network. It consists of a single neuron with adjustable synaptic weights and a hard limiter activation function. The weights of neurons in a neural network can increases or decreases the signal strength that it send over synapse. It can adjusts its weight in need to meet the desired output and that's why the weights in a neural network are called 'adjustable synaptic weights'. The main goal is to match the predicted output with the desired output.

Answer to the question no: 4(b)

Given, $W_{13} = 0.3$

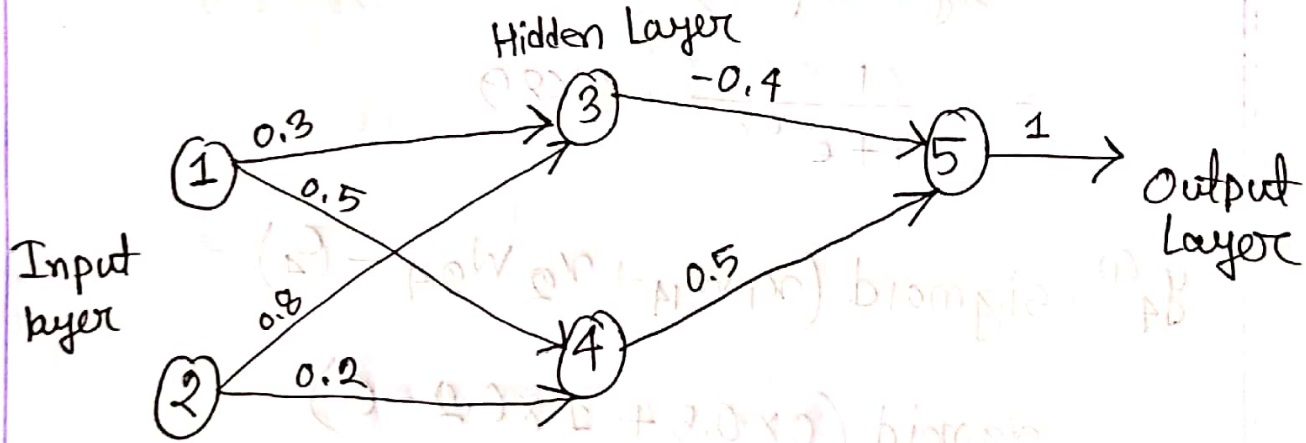
$W_{23} = (07\%\,2) - 0.2 = 1 - 0.2 = 0.8$

$W_{14} = (07\%\,3) - 0.5 = 1 - 0.5 = 0.5$

$W_{24} = 0.2$

$W_{35} = -0.4$

$W_{45} = 0.5$



$X = [0, 1]$, $Y = [1]$ $\therefore X_1 = 0$, $X_2 = 1$

Given, $\theta_3 = \theta_4 = \theta_5 = 0$

learning rate $\alpha = 0.1$

(i)

① Let us at first find out the output at hidden layer—

$$y_3^{(1)} = sigmoid(x_1 w_{13} + x_2 w_{23} - \theta_3)$$

$$= sigmoid(0 \times 0.3 + 1 \times 0.8 - 0)$$

$$= sigmoid(0.8)$$

$$= \frac{1}{1+e^{-0.8}} = 0.689$$

$$y_4^{(1)} = sigmoid(x_1 w_{14} + x_2 w_{24} - \theta_4)$$

$$= sigmoid(0 \times 0.5 + 1 \times 0.2 - 0)$$

$$= sigmoid(0.2)$$

$$= \frac{1}{1+e^{-0.2}} = 0.549$$

(ii)

~~Ac~~ Predicted output at output layer:

$$y_5 = \text{sigmoid}(y_3 W_{35} + y_4 W_{45} - \theta_5)$$

$$= \text{sigmoid}(0.689 \times (-0.4) + 0.549 \times 0.5 - 0)$$

$$= \text{sigmoid}(-0.2756 + \overset{0.2745}{\cancel{0.2895}} - 0)$$

$$\cancel{= \text{sigmoid}(0.0139)} = \text{sigmoid}(-0.0011)$$

$$\cancel{= \frac{1}{1+e^{-0.0139}}} = \frac{1}{1+e^{0.0011}}$$

$$\cancel{= 0.503} \qquad = 0.499$$

(iii)

At first calculating error at output layer:

$$e_5 = y_{d5} - y_5 = 1 - \overset{0.499}{\cancel{0.503}} = \cancel{0.497} = 0.501$$

Now calculating error gradients–

$$\delta_5 = y_5 [1 - y_5] \times e_5$$

$$\cancel{= 0.501 [1 - 0.501]} = 0.499[1 - 0.499] \times 0.501$$

$$= 0.125$$

Now updating weights between output & hidden layer:

$$\Delta W_{35} = \alpha \times y_3 \times \delta_5$$

$$= 0.1 \times 0.689 \times 0.125$$

$$= 0.0086$$

$$\Delta W_{45} = \alpha \times y_4 \times \delta_5$$

$$= 0.1 \times 0.549 \times 0.125$$

$$= 0.0068$$

Therefore,

$$W_{35}(2) = W_{35}(1) + \Delta W_{35}$$

$$= -0.4 + 0.0086 = -0.3914$$

$$W_{45}(2) = W_{45}(1) + \Delta W_{45}$$

$$= 0.5 + 0.0068$$

$$= 0.5068$$

This is the updated weights of output layer.