**Artificial Intelligence – BSCS – Fall 2022 FinalTerm Exam**

**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Reg.No. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Allowed Time: 2.5 hours Total Marks: 120**

**Question#1.** Following is the training dataset with attributes **Type, Color, Condition, and Refurbished.** Apply naïve Bayes algorithm to compute the prediction of the Origin for the following instance: **X = {Sports, Blue, New, Yes}**  **[ 20 ]**

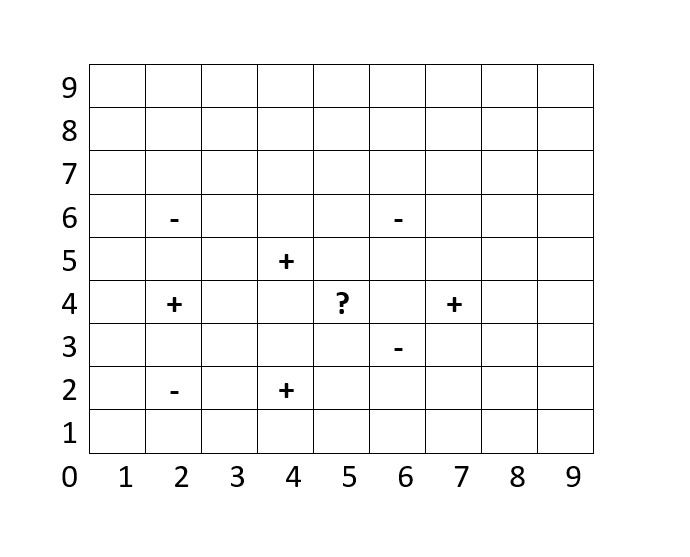
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type** | **Color** | **Condition** | **Refurbished** | **Origin** |
| Sports | Red | New | Yes | Domestic |
| SUV | Blue | Used | Yes | Imported |
| SUV | Blue | New | No | Domestic |
| Sports | Red | Used | Yes | Imported |
| SUV | Blue | New | No | Imported |
| Sports | Red | Used | Yes | Domestic |
| SUV | Blue | Used | No | Domestic |
| SUV | Red | New | No | Imported |
| Sports | Blue | Used | Yes | Imported |
| Sports | Red | Used | No | Domestic |

**Question#2.** Following are the product details collected from a supermarket in which **Size** and **Cost** of the product has been considered as **feature vectors**. It is required to distribute the products in to **TWO (2)** clusters. By using **K-Means clustering algorithm**, perform 2 iterations and identify of each cluster. **[ 20 ]**

**(Note:** **P2** and **P8** are the initial Clusters)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **P1** | **P2** | **P3** | **P4** | **P5** | **P6** | **P7** | **P8** | **P9** | **P10** |
| **Size** | **4** | **2** | **9** | **3** | **8** | **4** | **3** | **10** | **2** | **9** |
| **Price** | **22** | **27** | **40** | **25** | **37** | **16** | **26** | **39** | **25** | **40** |

**Question#3.** Consider the following dataset given with two classes i.e. positive (+1) and negative (-1). Data points as shown in the figure are:  **[ 10 + 10 = 20 ]**



**Negative:** (2, 2) (2, 6) (6, 3) (6, 6)

**Positive:** (2, 4) (4, 2) (4, 5) (7, 4)

An unknown point is located at **(5, 4).**

1. What is the prediction 1-Nearest Neighbor Classifier at point **(5,4).**
2. What is the prediction 3-Nearest Neighbor Classifier at point **(5,4).**

**Question#4.** Consider the following input and activation function. [**5 + 10 + 5 = 20]**

|  |  |  |
| --- | --- | --- |
| **Inputs** | | **Target** |
| **X1** | **X2** | **(t)** |
| 1 | 1 | -1 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 0 | 0 | 1 |

We initialize the parameters as: **w1** = 0.5, **w2** = 0.4 and **b** = 0.5, **learning rate (α)** = 0.25

1. Design the Single perception for the above dataset
2. Perform the first 2 epoch for training the perception. Write the learned weights clearly at end of each epoch.
3. What are limitations of the perception? Justify your answer

**Question#5**. The initial states **S0, S1, S2, S3** of **8-Queen problem** is given. Evaluate the **fitness function** of each state (no. of non-Attacking pairs). Select the **two pairs** for reproduction in accordance with their fitness ratio (**arrange the fitness values in ascending order**). For each pair to be mated, apply **two points crossovers** after 3nd and 6th queen. Apply **mutation by swapping extreme bits**. Write the new chromosomes. **[20]**

S1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | **Q4** |  |  |  |  |
|  |  |  |  |  | **Q6** |  |  |
|  |  |  |  |  |  |  |  |
| **Q1** |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  | **Q5** |  |  |  |
|  |  | **Q3** |  |  |  |  | **Q8** |
|  | **Q2** |  |  |  |  | **Q7** |  |

S0

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Q5** |  |  |  |
| **Q1** |  |  |  |  |  |  |  |
|  |  | **Q3** |  |  |  | **Q7** |  |
|  | **Q2** |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | **Q4** |  |  |  | **Q8** |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  | **Q6** |  |  |

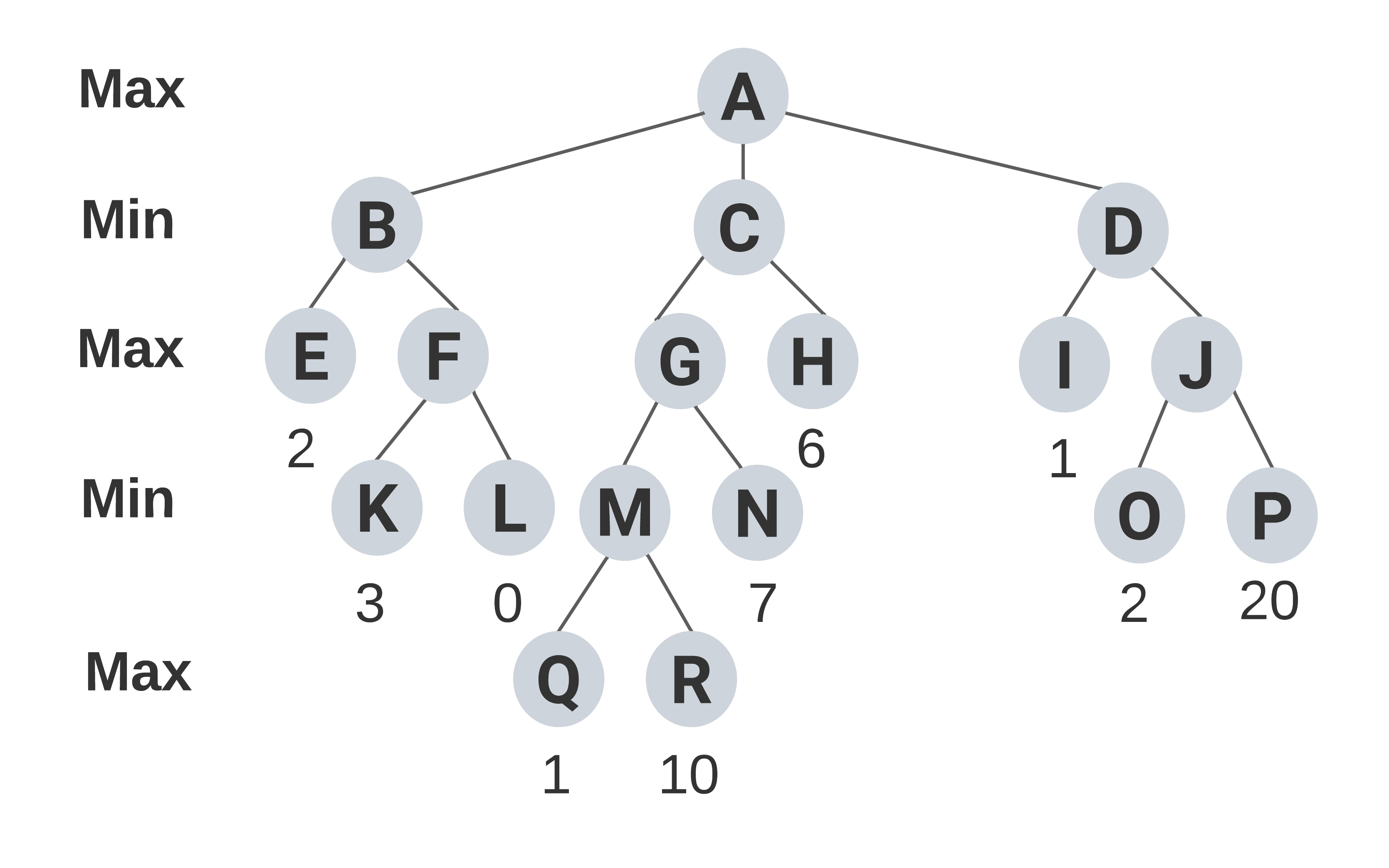
S3

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  | **Q8** |
|  |  | **Q3** |  |  |  |  |  |
|  |  |  |  |  |  | **Q7** |  |
| **Q1** | **Q2** |  |  |  |  |  |  |
|  |  |  | **Q4** | **Q5** |  |  |  |
|  |  |  |  |  | **Q6** |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

S2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Q2** |  | **Q4** |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  | **Q6** |  |  |
|  |  |  |  |  |  |  | **Q8** |
|  |  | **Q3** |  | **Q5** |  |  |  |
|  |  |  |  |  |  |  |  |
| **Q1** |  |  |  |  |  |  |  |
|  |  |  |  |  |  | **Q7** |  |

**Question#6.** Apply minimax search to find the best move for the player. Then apply alpha beta pruning and specify which branch is **alpha-cut** and which is **beta-cut**. **[10 + 10 = 20]**



**[Good Luck]**