USN □□□□□□□□□□□□□

RV University, Bengaluru

School of Computer Science and Engineering

B.Tech (Hons.)

Retest Examination – Answer keys

Academic Year 2024-2025

Course: IoT and Edge Computing		**Course Code: CS3100**	**Semester: V**
Date: 22 Nov 24	**Duration: 120 minutes**	**Max Marks: 30**	

Sl. No.	Questions	Marks	L1-L6	CO

1.	Give the output printed by the code given below.
	import numpy as np
	npStrDef = np.dtype([('name', 'S5'), ('id', 'i4'), ('score', 'f8')])
	npData = np.array([(b'Alice', 42, 95.5)], dtype=npStrDef)
	print('Item Size: ', npData.itemsize)
	print('Array Size: ', npData.nbytes)
	print('Value: ', npData)
		2	L4	2

2.	With a neat diagram explain how messages are transmitted in AMQP using publish subscriber model.	2	L4	3

3.	Consider the following C code, which is executed on a 64-bit system. Assume that the variable data is located at the memory address 0x7FFFC1234560.
	int main() {
	    float data = 3.14;
	    float* fPtr;
	    fPtr = &data;
	    fPtr++;
	    data += 1.0;
	    printf("The values in fPtr = %p and data = %.2f\n", fPtr, data);
	    return 0;
	}
	a) What will be the output of the printf statement?
	b) Explain how the value of fPtr is determined.
		2	L3	2

4.	In the below given code
	output_tensor = np.array([0.1, 0.7, 0.2]) # Sample output
	probabilities = np.exp(output_tensor) / np.sum(np.exp(output_tensor))
	        top_prediction = np.argmax(probabilities)
	Explain how the softmax probabilities are calculated and why np.argmax is used to select the top prediction.
		2	L2	2

L1- Remember, L2- Understand, L3 – Apply, L4- Analyse, L5 – Evaluate, L6 – Create

5.	Examine the relationship between input size and the number of weights in a fully connected layer.	2	L2	2
6.	In which scenarios would you use Binary Cross-Entropy versus Categorical Cross-Entropy for a car classification model?	2	L3	2
7.	What is overfitting and underfitting in a network in terms of model size?	2	L3	2
8.	How is MAE calculated, and when would you prefer to use MAE over MSE?	2	L2	2
9.	Briefly describe the stages of TFLite Model Conversion.	2	L2	2

10	The function initfResult sets the fResult array to zero before each prediction. Explain why this is necessary. What would happen if this step were skipped?
	void initfResult(float *fResult)
	{
	for(int i = 0; i < NUMBER_OF_OUTPUTS; i++)
	{
	fResult[i] = 0.0f;
	}
	}
		2	L3	3

11	The input features (price, maint, doors, etc.) of car model are encoded as one-hot vectors in the input arrays (input1, input2, etc.). Explain why one-hot encoding is used here and describe an alternative encoding method that could be applied to these features.	2	L3	2
12	Briefly describe the benefits of MQTT protocol.	2	L2	3

Part B

Instructions: Answer any two questions. The best two answers will be considered for evaluation.

Sl. No.	**Questions**	**Marks**	**L1-L6**	**CO**
1.	Draw the layered architecture of IoT and give a short description on each of the layers mentioning their responsibilities.	3	L2	1
2.	With an example Illustrate How does the field number affect data retrieval from ThingSpeak on ESP32?	3	L2	4
3.	Explain the purpose of Serial.begin(115200); in the setup function. Why is this important for IoT devices like the ESP32?	3	L2	3

L1- Remember, L2- Understand, L3 – Apply, L4- Analyse, L5 – Evaluate, L6 – Create

Course Outcomes

1. Choose a suitable wireless protocol based on the problem domain of an IoT product
2. Understand NumPy, TensorFlow framework and the need for TFLite for Edge Devices
3. Identify an IoT solution based on the features supported by ESP32 and FreeRTOS
4. Demonstrate integration of cloud computing platforms with ESP32
5. Demonstrate the need for AI accelerators for IoT in the ML domain

Marks Distribution										
L1	L2	L3	L4	L5	L6	CO1	CO2	CO3	CO4	CO5
0	19	10	4	0	0	3	18	9	3	0

****

Answers

Q1.
Item Size: 17
Array Size: 17
Value: [(b'Alice', 42, 95.5)]

Q2. In the **Publish-Subscribe (Pub-Sub)** messaging pattern, multiple subscribers can receive messages broadcast by a publisher. AMQP facilitates this using **exchanges** and **queues**.
Diagram-1 marks

Q3. The values in fPtr = 0x7FFFC1234564 and data = 4.14
Since fPtr is a float*, incrementing it (fPtr++) moves it by 4 bytes.

Q4. In the above code last two lines calculates the softmax probability and np.argmax(probabilities) finds the index of the maximum value in the probabilities array, which corresponds to the class with the highest probability.

Q5. In a fully connected (dense) layer of a neural network, each neuron in the layer has a weight associated with every input it receives. Therefore, the number of weights in a fully connected layer is directly related to the input size.

Q6. Binary Cross-Entropy:

- Used when there are **two classes** (e.g., "Car" vs. "Not a Car").
- Used in **multi-label classification**, where each label is treated independently (e.g., identifying whether a car is a "Sedan" and/or "Blue").

Categorical Cross-Entropy:

- Used when there are **more than two mutually exclusive classes** (e.g., classifying cars into brands like Toyota, Ford, or Tesla).

Q7. Overfitting and underfitting in the context of neural networks are heavily influenced by model size, which refers to the complexity of the model, specifically the number of parameters (e.g., weights and biases), layers, and neurons.
When the model is too large that what is required, overfitting happens. Similarly when it is too simpler in term of its size, it results in underfitting.

L1- Remember, L2- Understand, L3 – Apply, L4- Analyse, L5 – Evaluate, L6 – Create

Q8. Mean Absolute Error (MAE) is a metric used to evaluate the accuracy of a model's predictions. It measures the average of the absolute differences between the predicted values and the actual values.

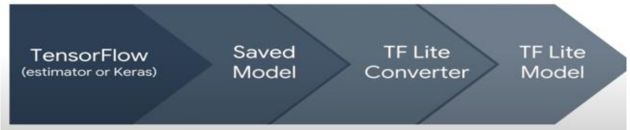
The formula to calculate MAE is:

$$MAE=1/n\sum|y_i-y_i\hat{}|$$

MAE is particularly useful when:

Interpretability is important: MAE gives an error in the same unit as the original data, making it easy to interpret. For example, if you're predicting house prices, MAE tells you the average absolute difference in prices, making the error directly interpretable.

Q9.



Q10. The initfResult function resets the fResult array to zero before each prediction, ensuring that each prediction starts with a clean state. This is important because fResult holds the model's output probabilities (or classification scores) for each output category, and initializing these values to zero prevents leftover values from previous predictions from affecting the new result.

If this step were skipped, the array would contain residual values from the previous prediction, which could lead to incorrect or misleading outputs. The model's output for the current input might be incorrectly combined with the prior results, making it difficult to interpret or validate the prediction's accuracy.

Q11. One-hot encoding is used here to represent categorical data in a way that the machine learning model can process efficiently. In this example, each categorical feature (like price, maint, doors, etc.) has multiple possible values, which are represented as binary arrays where each possible value corresponds to a specific position in the array. If a particular category is active, its position in the array is set to 1, and all other positions are set to 0.Alternative Encoding Method is Ordinal encoding or Integer encoding

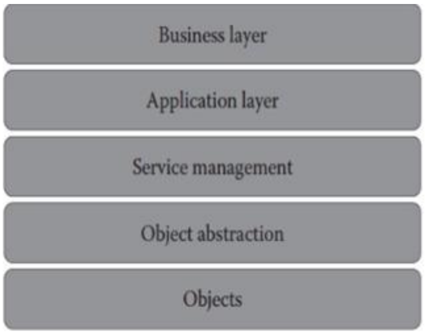
Q12.

1. **Lightweight and efficient**: MQTT minimizes the resources required by clients and network bandwidth.
2. **Bidirectional communication**: MQTT facilitates communication between devices and servers, supporting publishing and subscribing. It also allows broadcasting messages to groups of devices.
3. **Scalability**: MQTT can scale to support millions of devices or "things" in an IoT or IIoT ecosystem.

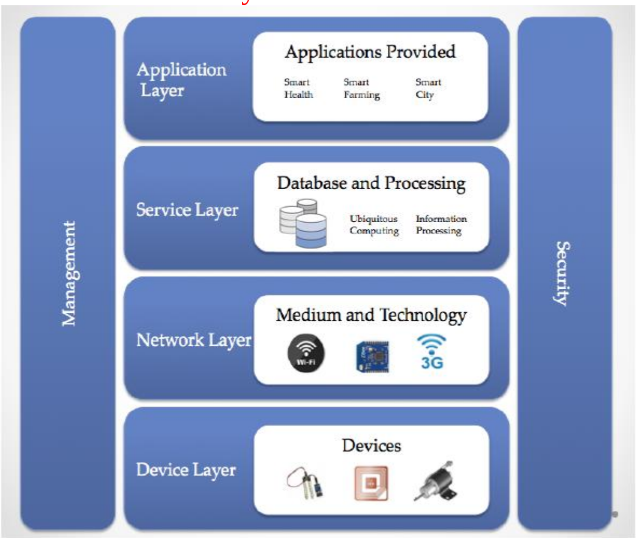
4. **Quality of Service (QoS) levels**: MQTT specifies different QoS levels to ensure reliable message delivery.
5. **Persistent sessions**: MQTT supports persistent sessions between devices and servers, reducing reconnection time over unreliable networks.
6. **Security features**: MQTT supports TLS encryption for message confidentiality and authentication protocols for client verification.

Part B

Q13.



Short description about each layer and its responsibilties. You can also draw the below diagram with a short note on each layer.



2. Artificial Intelligence (AI) and Machine Learning (ML) can significantly enhance a fan control system in a smart home by making it more accurate and responsive to varying conditions. like Data-Driven Decision Making,Personalized Control,Improved Accuracy and Efficiency

Q14. The **field number** in ThingSpeak determines which specific data stream within a channel to retrieve. Each channel can have up to 8 fields, and each field can store a different type of data (e.g., temperature, humidity, pressure). When retrieving data, the field number allows the ESP32 to target a specific field for the requested data.

Example:

- fields/1 would retrieve data from **Field 1** in a channel.
- fields/2 would retrieve data from **Field 2** in the same channel

Q15. The Serial.begin(115200); function in the setup function initializes serial communication with a specified baud rate of 115200 bits per second. This command configures the ESP32 microcontroller to communicate with a connected computer or another device (such as a monitoring tool or debugging interface) over the serial port at this speed.

****