	USN												

RV University, Bengaluru
School of Computer Science and Engineering
B.Tech (Hons.)
CP-1 Question Paper – Set1 with Answer keys
Academic Year 2024-2025

Course: IoT and Edge Computing	Course Code: CS3100	Semester: V
Date: 3 Sep 2024 **Duration: 90 minutes**	**Max Marks: 20**	

Sl. No.	Questions	Marks	L1-L6	CO
1.	Draw the layered architecture of IoT and give a short description of each of the layers mentioning their responsibilities.	2	L2	1
2.	What is the ISM band? Is it licensed or unlicensed? Give a couple of wireless protocols that work on this band.	2	L2	1
3.	Write a note on 5G and explain the roles of front haul and backhaul in network architecture and evaluate how each contributes to overall network performance in 5G and and its use cases in the IoT domain.	2	L5	1

4.	Consider the following C code which is executed on a **32-bit** system. Assume that the variable num is located in the memory address **0x00A1234.**
	
	 int main() {
	   int num = 0x10;
	   int* iPtr;
	   iPtr = #
	   iPtr++;
	   num++;
	   printf("The values in iPtr = %p and num = %d\n", iPtr, num);
	   return 0; }
	
	 What will be printed by the program when it runs?
		2	L4	2

5.	L1 = [1, 4, 'Hi', 5]
	 L2 = [5.0, 2.3, 'Hello', 4.0]
	
	 for elem in L1:
	   for x in L2:
	     if type(elem) == type(x):
	       print(elem, x)
	
	 What will be printed by this program when it runs?
		2	L4	2

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

6.	Conisder the definition of union type below.
	typedef union {
	    int i;
	    float f;
	    char c[2];
	} MY_DATA;
	MY_DATA myData;
	
	a. What is the size of **myData** in a 32-bit system?
	b. If myData.i = 0x41424344, what will be the values of myData.c[0], myData.c[1], myData.c[2], and myData.c[3] assuming little-endian memory representation?
		2	L3	2

7.	What is the differene between the APIs **xTaskCreate()** and **xTaskCreatePinnedToCore()?** Give sample invocation of these APIs with suitable parameters and illustrate the role of stack size as one of the paarmeters to the above APIs	2	L5	3
8.	Compare and Contrast the key differences between NB-IoT and LTE-M in terms of latency, throughput, and deployment?	1	L4	1
9.	Write a single line note on the four characteristics of WSN.	1	L2	1

10	void myFn(void)
	{
	    char charArr[] = "ABCDE";
	    int num1 = charArr[2] - charArr[1];
	    int num2 = charArr[4] - charArr[1];
	
	    printf("num1 = %d and num2 = %d\n", num1, num2);
	}
	What will be printed when the above function executes?
		1	L4	2

11	Write a Python program to flatten the given matrix by using comprehension and the output should be the square of the individual elements.
	
	myMatrix = [[1, 2, 3],
	            [4, 5],
	            [6, 7, 8, 9, 10]]
		1	L6	2

12	What is the OS running on ESP32 (lab execrcises) ? How it is different from any other commercial OS's like Windows/ Linux ?	1	L2	3
13	Write a short note on each of the macros used to access variable parameters passed to a function in a C program.	1	L2	2

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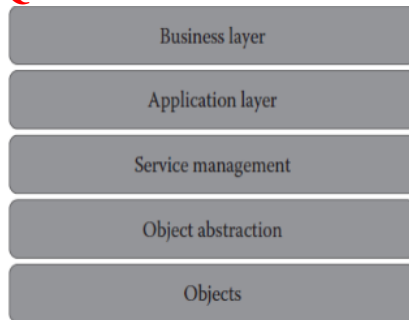
Course Outcomes

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

Marks Distribution									
L1	L2	L3	L4	L5	L6	CO1	CO2	CO3	CO4
0	7	2	6	4	1	8	9	3	0

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

Answers

Q1.

Business layer

Application layer

Service management

Object abstraction

Objects

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

Application Layer — Applications Provided — Smart Health, Smart Farming, Smart City

Service Layer — Database and Processing — Ubiquitous Computing, Information Processing

Network Layer — Medium and Technology — Wi-Fi, 3G

Device Layer — Devices
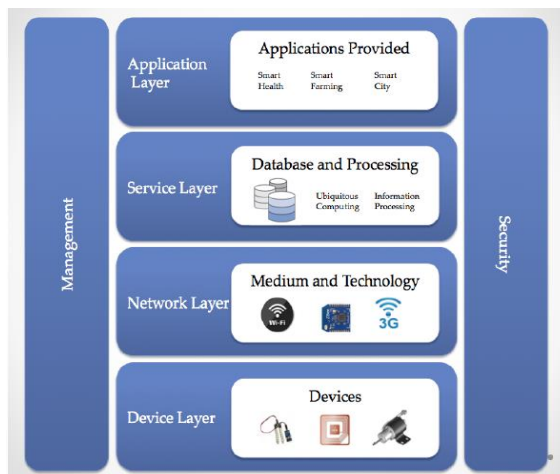
Management

Security

Diagram-**1 mark**
Explanation-**1 mark**

Q2. Industrial, scientific, and medical (ISM) bands are radio frequency (RF) bands that are reserved for use in science, industry, and medicine, rather than telecommunications.
They are unlicensed frequency spectrum.
WiFi,Bluetooth,UWB, Zigbee, NFC and RFID are the wireless protocols using ISM band and they are of LAN and PAN type networks.

Q3. 5G features-MU-MIMO,Beamforming ,data rate and latency
     Backhaul connects the mobile network to the wired network for large data transfer, while fronthaul
      connects remote cell sites to the baseband processing unit (BBU) for higher data speed.

Q4. The values in iPtr = 00A1238 and num = 17

Q5. Hi Hello

Q6. Union allocates the maximum size required by the element within in it. Since it is mentioned that it is

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

on 32-bit system, int here will be of 4 bytes and float is always four bytes.

Size of myData = 4 bytes – 1 mark
Assuming little endian machine, the lower address or c[0] and c[1] are the valid elements since the given char array is only of two bytes. Accessing myData.c[2] and c[3 will give a compilation error. But the memory contents will be as shown below.
myData.c[0] = 44, myData.c[1] = 43, myData.c[2] = 42 and myData.c[3] = 41 – 1 mark
Note: No marks will be reduced even if compilation error is not mentioned.

Q7. xTaskCreate() creates task that can be scheduled on any available core and
xTaskCreatePinnedToCore() creates task that needs to be run on a specific core (last parameter)

```
   void setup() {
    // Create a task
    xTaskCreate(              - 0.5 Mark
     vTaskCode,      /* Task function. */
     "Task 1",      /* Name of the task. */
     1000,          /* Stack size in words, not bytes. */
     NULL,          /* Task input parameter. */
     1,             /* Priority of the task. */
     NULL           /* Task handle. */
    );
   }

   // Create a task pinned to core 1
   xTaskCreatePinnedToCore(              - 0.5 Mark
    vTaskCode,      /* Task function. */
    "Task 2",      /* Name of the task. */
    1000,          /* Stack size in words, not bytes. */
    NULL,          /* Task input parameter. */
    1,             /* Priority of the task. */
    NULL,          /* Task handle. */
    1              /* Core on which the task should run. */
   );
  }
```

The stack size parameter is essential for defining how much memory is allocated to a task's stack. This allocation is crucial for the stability and reliability of task execution, as it directly impacts the task's ability to handle local data and function calls without running into stack overflow issue. **– 1 Mark**

Q8.
   ▢ **Latency:** LTE-M offers lower latency compared to NB-IoT, making it better for real-time applications.
   ▢ **Throughput:** LTE-M provides higher throughput, suitable for applications requiring more data transmission, whereas NB-IoT is designed for lower throughput needs.
   ▢ **Deployment:** NB-IoT is optimized for deep coverage and high device density, while LTE-M is more suitable for applications requiring mobility and higher data rates.

Q9. WSN: Wireless Sensor Networks.
   The four characterstics are: Redundancy Exploitation, Data-centric Routing, Data Aggregation and Localized Algorithms. Write a single line note on each.

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

Q10. num1 = 1 and **num2 = 3   - 0.5 marks each**

Q11. # Flaten it using comprehension and get the square of each element
flatMatrix = [x*x for row in myMatrix for x in row]

Q12. FreeRTOS is the Real Time OS running on ESP32.
FreeRTOS is an open-source Real-Time Operating System (RTOS) designed for embedded systems. It provides multitasking capabilities, allowing multiple tasks to run concurrently on a single microcontroller This highly responsive to events and predictable latencies involved in its execution compared to other commercial OS such as Windows and Linux. FreeRTOS is also having a smaller footprint in terms of its memory requirments and also has simpler structure in terms of process/task management. It does not maintain parent-child relationship as well as concept of threads running within the context of tasks or processes.
Note: Need to mention any two of the above need to be mentioned. Additional points which are valid will also be considered for evaluation.

Q13. va_start(), va_arg() and va_end()
va_list args; - Variable of type va_list is declared to hold the starting address of arguments.
va_start(args, fmt); - Initializes the starting addr of list of variables passed into the function
va_arg(args, type) accesses the argument of a particular type by updating the argument list
va_end(args); - Clears the list created above.

******