Name of the Course: Computer Science Practical 4

Sr. No.	Heading	Particulars
1	**Description the course:**	**Introduction:** This major practical course is a blend of two rapidly evolving areas in computing — Computer Networking and the Internet of Things (IoT). While the first part equips students with hands-on experience in network creation, routing, and traffic analysis, the second part transitions them into the world of intelligent connected devices. Students explore the design and deployment of smart systems using Raspberry Pi and Arduino kits, integrate sensors/actuators, and implement cloud-based interactions. **Relevance:** This course is highly relevant in today's digital world where interconnected devices and reliable communication networks form the backbone of every industry. The integration of Computer Networks and IoT ensures students gain comprehensive knowledge about both data communication and real-time embedded systems. As businesses and governments adopt automation and smart systems, professionals with expertise in both domains are in increasing demand. **Usefulness:** The course equips students with practical skills to configure networks, interface hardware, and use protocols for communication and control. It bridges the gap between theory and practice by allowing learners to build, simulate, and test complete systems. These hands-on experiences enhance understanding and build a strong foundation for advanced learning or employment. **Application:** The knowledge gained in this course can be directly applied to developing IoT systems like smart homes, environmental monitoring, health devices, and automated machinery. Students also learn to simulate network infrastructures and deploy routing protocols, which are essential in setting up enterprise-level networks. Real-world applications of this course span domains such as agriculture, transportation, education, and urban planning. **Interest:** This course captures students' interest through interactive hardware activities like blinking LEDs, controlling motors, and streaming sensor data. Tools like Cisco

		Packet Tracer, Node-RED, and Wireshark provide engaging visual and analytical insights into networking and IoT systems. Students are encouraged to innovate and experiment, keeping the learning experience dynamic and enjoyable. **Connection with Other Courses:** The course complements core subjects such as Computer Networks, Microprocessor and Embedded Systems, Operating Systems, and Web Programming. It allows students to apply concepts from these theory courses in a practical setup, promoting interdisciplinary learning. The use of programming and cloud communication also ties into courses like Data Science and Cloud Computing. **Demand in the Industry:** There is a significant and growing demand for professionals skilled in IoT and network technologies across industries such as healthcare, smart infrastructure, manufacturing, and IT services. Companies look for candidates who can develop, deploy, and maintain systems involving both physical hardware and software networking components. This course provides the skillset that aligns with industry trends and emerging technologies. **Job Prospects:** Students completing this course are better prepared for roles such as IoT Developer, Network Administrator, Embedded Systems Programmer, and Cloud-IoT Integrator. They can work in sectors including telecom, automation, R&D, and smart technologies. The practical skills gained make them suitable for both core technical roles and interdisciplinary project teams.
2	**Vertical:**	Major
3	**Type:**	Practical
4	**Credits:**	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	**Hours Allotted:**	60 Hours
6	**Marks Allotted:**	50 Marks
7	**Course Objectives(CO):** **CO 1.** Develop hands-on proficiency in computer networking tools, commands, and simulation using Cisco Packet Tracer. **CO 2.** Explore network topologies, routing protocols (RIP, OSPF, BGP), and IP addressing techniques (static/dynamic). **CO 3.** Understand and apply various IoT communication protocols, platforms, and hardware interfaces using Raspberry Pi/Arduino.	

	CO 4. Interface sensors, actuators, and edge devices for real-world IoT applications. **CO 5.** Integrate hardware with software tools like Node-RED and cloud platforms for complete IoT solutions.
8	**Course Outcomes (OC):** After successful completion of this course, students would be able to - **OC 1.** Use network diagnostic and configuration commands effectively on Windows and Linux systems. **OC 2.** Design and simulate wired and wireless networks using Cisco Packet Tracer with IP configurations and routing protocols. **OC 3.** Analyze network traffic using Wireshark and identify protocol layers and data flow. **OC 4.** Configure and test IoT hardware platforms for device communication and data acquisition. **OC 5.** Implement real-time IoT applications using sensors, actuators, and cloud communication. **OC 6.** Develop integrated solutions using web technologies, IoT protocols, and dashboarding tools.
9	**Modules:-** **Module 1 (30 hours):**
	Practical based on Computer Networks **Exploring Networking Commands via Windows CMD / LINUX Terminal** • Execute and observe the output of: ping, traceroute / tracert, netstat, arp, ipconfig / ifconfig, getmac, hostname, nslookup, pathping, systeminfo • Discuss the purpose and interpretation of each command's output **Learning Focus:** Understanding diagnostic and configuration commands. **Building a Basic Peer-to-Peer Network** Create a network with **two PCs** connected using a **crossover cable** using Cisco Packet Tracer • Assign static IP addresses • Test connectivity with ping and Packet Tracer simulation **Static IP Setup with One Server and Two Clients** Connect **1 server** and **2 computers** using a switch (Cisco Packet Tracer) • Use **static IP addresses** • Configure server services (e.g., HTTP or FTP) and test connectivity from clients **Dynamic IP Allocation with Server and Clients** Use **DHCP service** from a server to assign IPs to **two PCs** (Cisco Packet Tracer) • Enable and configure DHCP on the server

	• Verify IP allocation and connectivity using ipconfig **Creating a Mixed Network with Wired and Wireless Devices** One server, two wired PCs, and two **mobile/wireless devices (**Cisco Packet Tracer**)** • Use appropriate cabling and access points • Assign IPs and test cross-device communication **RIP Version 1 Routing Across Three Routers** Three routers, each connected to at least **three PCs (**Cisco Packet Tracer**)** • Implement **RIPv1** routing between routers • Verify inter-network connectivity using ping and route tables **RIP Version 2 Implementation** Three routers, each connected to at least **three PCs (**Cisco Packet Tracer**)** • Enable **RIPv2** and observe subnet mask handling • Use Packet Tracer's simulation mode to observe routing updates **OSPF Routing and Network Hierarchies** Three routers and their networks using **OSPF** **(**Three routers, each connected to at least **three PCs (**Cisco Packet Tracer**)** • Assign area IDs, router IDs, and enable OSPF • Monitor OSPF neighbour relationships and path selections **BGP for Inter-domain Routing** Three autonomous systems (AS) with routers running **BGP** Three routers, each connected to at least **three PCs (**Cisco Packet Tracer**)** • Configure BGP with different AS numbers • Establish peerings and test inter-AS communication **Protocol Analysis with Wireshark** • Set up network transactions for each protocol: ICMP (ping), TCP (web browsing), HTTP (via browser), UDP (DNS), FTP (file transfer) • Apply filters and observe packet contents
	Module 2 (30 hours):
	Practical based on IoT Technologies
	Preparing the IoT Hardware • Set up Raspberry Pi OS / Arduino IDE • Configure GPIO settings and test basic connectivity • Demonstrate pin layout and onboard peripherals

	GPIO – Light the LED (with and without Button) • Blink LED using Python (Raspberry Pi) or C++ (Arduino) • Add a push button to toggle LED ON/OFF **SPI Interface – Camera Module Integration** • Connect a Pi camera module (or SPI camera for Arduino) • Capture an image or short video • Store file or stream it locally **8x8 LED Grid Control (Matrix LED Programming)** • Connect an 8×8 LED matrix module • Program animations or scrolling text patterns • Explore logical formulas for patterns **PWM – Stepper Motor Control** • Interface a stepper motor using a motor driver • Control direction and vary speed using PWM signals • Observe effect of duty cycle changes on motor movement **Node-RED for IoT Dashboard** • Install and configure Node-RED on Raspberry Pi • Create a flow to turn LED ON/OFF via browser • Add visual interface for sensor data (e.g., temperature) **Sensor Integration – Analog & Digital Sensors** • Interface multiple sensors (LDR, DHT11, Gas) • Collect and display data on serial monitor / OLED / LCD • Trigger actions (e.g., fan ON if temp > threshold) **Web Trigger – Control GPIO from Web Server** • Host a simple Flask web app (Raspberry Pi) or ESP Web Page (Arduino) • Control a set of LEDs via buttons on a webpage • Ensure real-time response and feedback **IoT Protocol – Send Sensor Data Online** • Use HTTP or MQTT to push sensor values to a cloud server (e.g., Thingspeak) • Implement publishing logic with timestamps • Use Arduino/NodeMCU or Raspberry Pi as the publisher

	Integration – Smart Monitoring System • Combine sensors, actuators, communication protocol, and web/cloud • Example: A Smart Weather Station that logs temp/humidity online and triggers fan/LED alerts ***Note: The above practicals can be performed on Raspberry Pi Kits and / or Arduino kits as per the need***	
10	**Text Books** 1. Kurose, J.F. & Ross, K.W. (2021). *Computer Networking: A Top-Down Approach* (7th ed.). Pearson Education India. 2. Ramya, V., & Shanmuga Priya, K. (2019). *Practical Internet of Things: Concepts, applications and security*. Chennai: Wiley India Pvt. Ltd.	
11	**Reference Books** 1. Forouzan, B.A. (2017). *Data Communications and Networking* (5th ed.). McGraw Hill Education India. 2. Bahga, A. & Madisetti, V. (2014). *Internet of Things: A Hands-on Approach*. Universities Press India.	
12	**Internal Continuous Assessment: 40%**	**Semester End Examination: 60%**