|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE** | | | | | **DEPARTMENT OF COMPUTER SCIENCE ENGINEERING** | | | | |
| **ProgramName:**B. Tech | | | | **Assignment Type: Lab** | | | **AcademicYear:**2025-2026 | | |
| **CourseCoordinatorName** | | | | Venkataramana Veeramsetty | | | | | |
| **Instructor(s)Name** | | | | |  | | --- | | Dr. V. Venkataramana (Co-ordinator) | | Dr. T. Sampath Kumar | | Dr. Pramoda Patro | | Dr. Brij Kishor Tiwari | | Dr.J.Ravichander | | Dr. Mohammand Ali Shaik | | Dr. Anirodh Kumar | | Mr. S.Naresh Kumar | | Dr. RAJESH VELPULA | | Mr. Kundhan Kumar | | Ms. Ch.Rajitha | | Mr. M Prakash | | Mr. B.Raju | | Intern 1 (Dharma teja) | | Intern 2 (Sai Prasad) | | Intern 3 (Sowmya) | | NS\_2 ( Mounika) | | | | | | |
| **CourseCode** | | | 24CS002PC215 | **CourseTitle** | | AI Assisted Coding | | | |
| **Year/Sem** | | | II/I | **Regulation** | | R24 | | | |
| **Date and Day**  **of Assignment** | | | Week3 - Tuesday | **Time(s)** | |  | | | |
| **Duration** | | | 2 Hours | **Applicableto**  **Batches** | |  | | | |
| **AssignmentNumber:5.2**(Present assignment number)/**24**(Total number of assignments) | | | | | | | | | |
|  | | | | | | | | | |
|  | **Q.No.** | **Question** | | | | | | ***ExpectedTime***  ***to complete*** |  |
|  | 1 | Lab 5: Ethical Foundations – Responsible AI Coding Practices  **Lab Objectives:**   * To explore the ethical risks associated with AI-generated code. * To recognize issues related to security, bias, transparency, and copyright. * To reflect on the responsibilities of developers when using AI tools in software development. * To promote awareness of best practices for responsible and ethical AI coding.   **Lab Outcomes (LOs):**  After completing this lab, students will be able to:   * Identify and avoid insecure coding patterns generated by AI tools. * Detect and analyze potential bias or discriminatory logic in AI-generated outputs. * Evaluate originality and licensing concerns in reused AI-generated code. * Understand the importance of explainability and transparency in AI-assisted programming. * Reflect on accountability and the human role in ethical AI coding practices..   **Task Description#1 (Privacy and Data Security)**   * Use an AI tool (e.g., Copilot, Gemini, Cursor) to generate a login system. Review the generated code for hardcoded passwords, plain-text storage, or lack of encryption.   **Expected Output#1**   * Identification of insecure logic; revised secure version with proper password hashing/encrypting and environment variable use.   **Prompt:** Write a python program to generate a login system and reviewthe generated code for hardcoded passwords, plain-text storage, or lack of encryption  **Code & Output**        **Explaination:**   * **The code defines an Applicant with name, age, income, credit score.** * **It has two evaluators:** * **Unbiased: approves if age ≥ 21, income ≥ 30000, score ≥ 650.** * **Biased: raises thresholds for inferred-female names and penalizes the name “Priya”.** * **It parses “loan approval for John” and “loan approval for Priya”, evaluates both, prints results and a bias analysis showing how biased rules treat identical profiles differently.**   **Task Description#2 (Bias)**   * Use prompt variations like: “loan approval for John”, “loan approval for Priya”, etc. Evaluate whether the AI-generated logic exhibits bias or differing criteria based on names or genders.   **Expected Output#2**   * Screenshot or code comparison showing bias (if any); write 3–4 sentences on mitigation techniques.   **Prompt**: Write a Python program to Generate a loan approval system that takes applicant details like name, age, income, and credit score. Test it with inputs such as *‘loan approval for John’* and *‘loan approval for Priya’*. Compare the logic or outputs to check if the system applies different criteria based on gender or names. Highlight any bias you find.  **Code & Output:**        **Explaination:**  **This file has a function called fibonacci(n) that returns the nth number in the Fibonacci sequence. The sequence starts with 0 and 1, and every next number is the sum of the previous two.**   * **First, it checks if n is negative. If it is, it raises an error because Fibonacci positions can’t be negative.** * **Then it handles the two simplest cases:** * **If n == 0, it returns 0.** * **If n == 1, it returns 1.** * **For any other n, it uses recursion: it calls itself to compute the two previous Fibonacci numbers and adds them: fibonacci(n - 1) + fibonacci(n - 2). This mirrors the mathematical definition exactly.**   **At the bottom, there’s a small demo that runs only if you execute this file directly. It loops from 0 to 9 and prints each Fibonacci value so you can see the function working.Note: This plain recursive method is easy to read but gets slow as n grows because it recalculates the same values many times. For big n, using memoization or an iterative loop would be faster.**  **Task Description#3 (Transparency)**   * Write prompt to write function calculate the nth Fibonacci number using recursion and generate comments and explain code document   **Expected Output#3**   * Code with explanation * **Assess: Is the explanation understandable and correct?**   **Prompt:** Write a Python function to calculate the nth Fibonacci number using recursion and generate comments  **Code & Output:**    **Explaination:**   * **The function fibonacci(n) gives you the nth number in the Fibonacci sequence, where the sequence starts 0, 1, 1, 2, 3, 5, 8, ....** * **It first checks if n is negative. If yes, it raises an error because Fibonacci isn’t defined for negative positions.** * **It has two “stop” points:** * **If n is 0, it returns 0.** * **If n is 1, it returns 1.** * **For any other n, it calls itself twice to add the two previous Fibonacci numbers:** * **fibonacci(n - 1) and fibonacci(n - 2).** * **That’s exactly how the Fibonacci sequence is defined.** * **At the bottom, there’s a small demo that only runs when you run this file directly. It prints F(0) through F(9) so you can see the results.** * **This method is clear but gets slow for big n because it repeats work. For larger values, you’d use memoization or a loop.**   **Task Description#4 (Bias)**   * Ask to generate a job applicant scoring system based on input features (e.g., education, experience, gender, age). Analyze the scoring logic for bias or unfair weightings.   **Expected Output#4**   * Python code * Analyze is there any bias with respect to gender or any   **Prompt:** Write a python program which ask to generate a job applicant scoring system based on input features (e.g., education, experience, gender, age). Analyze the scoring logic for bias or unfair weightings  **Code & Output:**        **Explaination:**   * **It asks for education, experience, skills, gender, and age, then converts them to 0–1.** * **You enter weights for each factor; the program normalizes them so they sum to 1.** * **Final score = sum of each feature’s value × its normalized weight, with a printed breakdown.** * **Bias check warns if gender/age have any non-zero weight, if one factor dominates too much, or if a factor has a negative weight that could unfairly penalize groups.** * **Fair use: keep gender/age at 0, balance experience with skills, and make weights reflect real job needs.**   **Task Description#5 (Inclusiveness)**   * Code Snippet     **Expected Output#5**   * Regenerate code that includes **gender-neutral** also   **Prompt:** Write a python code to **re**generate the above code that includes **gender-neutral** also  **Code & Output:**   1. **Before Regenerating**       **2.After Regerating**    **Note: Report should be submitted a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots**  **Evaluation Criteria:**   | **Criteria** | **Max Marks** | | --- | --- | | Transparency | 0.5 | | Bias | 1.0 | | Inclusiveness | 0.5 | | Data security and Privacy | 0.5 | | **Total** | **2.5 Marks** | | | | | | | Week3 - Wednesday |  |