|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** | | | | 1. Dr. Mohammed Ali Shaik  2. Dr. T Sampath Kumar  3. Mr. S Naresh Kumar  4. Dr. V. Rajesh  5. Dr. Brij Kishore  6. Dr Pramoda Patro  7. Dr. Venkataramana  8. Dr. Ravi Chander  9. Dr. Jagjeeth Singh | | | | | |
| **CourseCode** | | | 24CS002PC215 | **CourseTitle** | | AI Assisted Coding | | | |
| **Year/Sem** | | | II/I | **Regulation** | | R24 | | | |
| **Date and Day**  **of Assignment** | | |  | **Time(s)** | |  | | | |
| **Duration** | | | 2 Hours | **Applicableto**  **Batches** | |  | | | |
| **AssignmentNumber:3.3**(Present assignment number)/**24**(Total number of assignments) | | | | | | | | | |
|  | | | | | | | | | |
|  | **Q.No.** | **Question** | | | | | | ***ExpectedTime***  ***to complete*** |  |
|  | 1 | Lab 3: Prompt Engineering – Improving Prompts and Context Management  **Lab Objectives:**   * To understand how prompt structure and wording influence AI-generated code. * To explore how context (like comments and function names) helps AI generate relevant output. * To evaluate the quality and accuracy of code based on prompt clarity. * To develop effective prompting strategies for AI-assisted programming.   **Lab Outcomes (LOs):**  After completing this lab, students will be able to:   * Generate Python code using Google Gemini in Google Colab. * Analyze the effectiveness of code explanations and suggestions by Gemini. * Set up and use Cursor AI for AI-powered coding assistance. * Evaluate and refactor code using Cursor AI features. * Compare AI tool behavior and code quality across different platforms.   **Task Description#1**   * Try 3 different prompts to generate a factorial function. * Prompt1: write a python code to generate a factorial function.     **Prompt2:** **write a python code to generate a factorial function for a number given by user**    **Prompt3: Create a Python program to compute the factorial of a number using recursion with proper function docstring.**    **Expected Output#1**   * Comparison of AI-generated code styles   OUTPUT 1:    OUTPUT2:    OUTPUT3:    **Report:** for your given prompts:   1. The first prompt asks to create a Python program that generates a factorial function. 2. The second prompt improves it by letting the user input the number for factorial calculation. 3. The task is implemented using recursion with a clear docstring for better code understanding. 4. This approach helps in learning function design, recursion, and user interaction in Python.     **Task Description#2**   * Provide a clear example input-output prompt to generate a sorting function. * Prompt : write a python code with a clear example input-output prompt to generate a sorting function.   **Expected Output#2**   * Functional sorting code from AI * OUTPUT1:   **Task Description#3**   * Start with the vague prompt “Generate python code to calculate power bill” and improve it step-by-step * **Prompt1:** Write a Python program that takes the number of electricity units consumed as input and calculates the total bill based on the following custome * . * **Prompt2:** Generate Python code to calculate an electricity bill based on the number of units consumed, using a fixed rate per unit.   **Expected Output#3**   * Enhanced AI output with clearer prompts * OUTPUT1:   OUTPUT2:  **Report:** for your electricity bill prompts:   1. The first prompt focuses on writing a program to calculate the electricity bill with given conditions. 2. The second prompt simplifies it by using a fixed rate per unit for billing. 3. Both tasks improve understanding of conditional logic and arithmetic in Python. 4. This helps in applying programming concepts to real-life billing calculations.   **Task Description#4**   * Write structured comments to help AI generate two linked functions (e.g., login\_user() and register\_user()). * Prompt:generate a two linked functions in python (Celsius,fahrenheit)   **Expected Output#4**   * Consistent functions with shared logic   OUTPUT:    **Task Description#5**   * Analyzing Prompt Specificity: Improving Temperature Conversion Function with Clear Instructions * Prompt1: Write a Python function to convert Celsius to Fahrenheit and Kelvin.   **Promt2:**  Write a Python function that converts temperature between Celsius and Fahrenheit, based on user input**.**    **Expected Output#5**   * Code quality difference analysis for various prompts   OUTPUT1:  OUTPUT2:    **Report:** for your temperature conversion prompts:   1. The first prompt requires a Python function to convert Celsius into Fahrenheit and Kelvin. 2. The second prompt extends functionality by allowing user choice between Celsius and Fahrenheit. 3. These tasks enhance knowledge of formulas, functions, and user interaction in Python. 4. Such conversions show practical applications of coding in science and daily life.   **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** | | --- | --- | | Factorial Function (Task#1) | 0.5 | | Sorting Function (Task#2) | 0.5 | | Vogue Vs. Specific Prompting (Task #3) | 0.5 | | Linked Functions (Task #4) | 0.5 | | Temperature Conversion Function (Task #5) | 0.5 | | **Total** | **2.5 Marks** | | | | | | | 03.08.2025 EOD |  |