|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SCHOOLOFCOMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE** | | | | | **DEPARTMENTOFCOMPUTER SCIENCE ENGINEERING** | | | | |
| **ProgramName:**B. Tech | | | | **AssignmentType: 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 | | | |
| **DateandDay**  **of Assignment** | | | Week6 - WednesDay | **Time(s)** | |  | | | |
| **Duration** | | | 2 Hours | **Applicableto**  **Batches** | |  | | | |
| **AssignmentNumber:12.3**(Presentassignmentnumber)/**24**(Totalnumberofassignments) | | | | | | | | | |
|  | **Q.No.** | **Question** | | | | | | ***ExpectedTime***  ***to complete*** |  |
|  | 1 | **Lab 12 – Algorithms with AI Assistance: Sorting, searching, and optimizing algorithms**  **Lab Objectives**   * To implement classical algorithms (sorting, searching) with the help of AI tools. * To analyze AI suggestions for efficiency and correctness. * To explore AI-assisted optimizations of existing algorithms. * To compare naive vs. optimized approaches generated by AI.   **Learning Outcomes**  After completing this lab, students will be able to:   Implement sorting and searching algorithms using AI suggestions.   Compare AI-generated algorithm variants in terms of readability and efficiency.   Use AI to optimize brute-force algorithms into more efficient ones.   Analyze algorithm complexity (time and space) with AI explanations.   Critically reflect on correctness, clarity, and maintainability of AI-generated algorithms.  **Task Description #1 – Linear Search implementation**  Task: Write python code for linear\_search() function to search a value in a list and extract it’s index.  PROMPT 1: Write a Python function linear\_search(arr, target) that searches for a value in a list and returns its index if found, or -1 if not found. Add comments explaining how the algorithm works and its time complexity.  Output:      **Task Description #2 – Sorting Algorithms**  Task: Ask AI to implement Bubble Sort and check sorted output  Prompt 2: implement Bubble Sort and check sorted output  Output:    **Task Description #3 – Optimization**  Task: Write python code to solve below case study using linear optimization    **Prompt 3:** Write Python code using the PuLP or scipy.optimize library to solve a simple linear optimization problem (for example: maximize 3x + 5y subject to constraints 2x + 3y ≤ 12, x + y ≤ 5, and x, y ≥ 0). Show the optimal values and the maximum result.  Output:    **Task Description #4 – Gradient Descent Optimization**  Task: Write python code to find value of x at which the function f(x)=2X3+4x+5 will be minimum  Prompt 4: Write a Python program to find the value of x at which the function f(x) = 2x^3 + 4x + 5 reaches its minimum using gradient descent. Use a suitable learning rate and stopping condition. Plot the function and mark the minimum point.  Output: | | | | | | Week5 - Monday |  |