|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** | | | Week10 - Thursday | **Time(s)** | |  | | | |
| **Duration** | | | 2 Hours | **Applicableto**  **Batches** | |  | | | |
| **AssignmentNumber:19.4**(Presentassignmentnumber)/**24**(Totalnumberofassignments) | | | | | | | | | |
|  | | | | | | | | | |
|  | **Q.No.** | **Question** | | | | | | ***ExpectedTime***  ***to complete*** |  |
|  | 1 | **Lab 19 – Code Translation: Converting Between Programming Languages**  **Lab Objectives:**   * Understand how AI tools can assist in translating code between different programming languages. * Learn to verify correctness and functionality after translation. * Explore syntactic and semantic differences between languages (e.g., Python, Java, C++). * Practice debugging and optimizing AI-translated code.   **Task 1: Translate a Simple Program (Python → JavaScript)**   * **Instructions:**   + Write a Python function print\_numbers() that prints the first 10 natural numbers using a loop.   + Translate the function into JavaScript as a reusable function printNumbers().   + Call the function in both languages to display results. * **Expected Output:** * **1** * **2** * **3** * **...10**   **Task 2: Convert Conditional Statements (Java → Python)**   * **Instructions:**   + Write a Java method checkNumber(int num) that checks if a number is positive, negative, or zero.   + Translate the method into a Python function check\_number(num).   + Call the function/method with different inputs and compare outputs. * **Expected Output:**   1. Input: -5 → Output: The number is negative   2. Input: 0 → Output: The number is zero   3. Input: 7 → Output: The number is positive   **Task 3: Translate Recursive Function (Python → C++)**   * **Instructions:**   + Write a Python function factorial(n) that calculates factorial of a number using recursion.   + Translate the same into a C++ function int factorial(int n).   + Call the function in both languages with inputs 5 and 0. * **Expected Output:**   1. **Input: 5 → Output: Factorial = 120**   2. **Input: 0 → Output: Factorial = 1**   .  **Task 4: Data Structures with Functions (JavaScript → Python)**   * **Instructions:**   + Write a JavaScript function printStudents(students) that takes an array of student names and prints each name.   + Translate it into a Python function print\_students(students) using a list.   + Test both functions with sample student names. * **Expected Output:** * Student List: * Alice * Bob * Charlie   **Task 5: Class & Object Translation (Python → Java)**   * **Instructions:**   1. Write a **Python class** Car with attributes: brand, model, year.   2. Add a **method**display\_details() that prints car details.   3. Translate the same into a **Java class** Car with attributes and a **method**displayDetails().   4. Create an object in both languages and call the method. * **Expected Output:** * Car Details: * Brand: Toyota * Model: Corolla   Year: 2020  ✅ Deliverables (For All Tasks)   1. AI-generated prompts for code and test case generation. 2. At least 3 assert test cases for each task. 3. AI-generated initial code and execution screenshots. 4. Analysis of whether code passes all tests. 5. Improved final version with inline comments and explanation. 6. Compiled report (Word/PDF) with prompts, test cases, assertions, code, and output.   Top of Form | | | | | | Week10 - Thursday |  |