Session 3: Introduction to Programming, Problem Solving, Pseudo Code, Types of Languages, Intro to Compilers, Control Flow, and Functions

Duration: 2 hours

Session Objectives:

- Understand the fundamentals of programming and problem-solving.

- Learn the concept of pseudo code and its importance.

- Explore different types of programming languages and their characteristics.

- Gain insights into the role of compilers in the programming process.

- Understand control flow structures and their application.

- Grasp the concept of functions and their significance in programming.

Student Capability After the Sessions

- Ability to write basic algorithms using pseudo code.

- Knowledge of different programming language types and their applications.

- Understanding the role of compilers in the software development process.

- Proficiency in using control flow structures in programming.

- Ability to create and use functions for modular programming.

Agenda:

1. Welcome and Overview (5 mins)

- Introduction to the session objectives.

2. Programming Fundamentals and Problem Solving (15 mins)

- Overview of programming and problem-solving.

- Introduction to algorithmic thinking.

3. Introduction to Pseudo Code (20 mins)

- Definition and purpose of pseudo code.

- Writing simple algorithms in pseudo code.

- Importance of clear and readable pseudo code.

4. Types of Programming Languages (15 mins)

- Overview of programming language classifications.

- Examples of popular programming languages.

5. Intro to Compilers (10 mins)

- Explanation of compilers and their role in programming.

- Differentiating between interpreted and compiled languages.

6. Control Flow in Programming (20 mins)

- Explanation of control flow structures: sequential, conditional, and iterative.

- Examples demonstrating control flow in programming.

7. Functions in Programming (15 mins)

- Definition and purpose of functions.

- Creating and using functions.

- Advantages of modular programming.

8. Interactive Examples and Coding Exercise (10 mins)

- Live coding examples and hands-on exercise for participants.

9. Q&A and Discussion (10 mins)

- Addressing questions from participants.

- Facilitating a discussion on covered topics.

Practice Questions for Students:

1. Write a simple algorithm for finding the maximum of three numbers using pseudo code.

2. Compare and contrast two different programming languages, highlighting their strengths and weaknesses. Java Vs Python | Angular Vs React JS | SQL Database (MySQL) Vs No SQL database(Mongo Db)

3. Explain the compilation process and how it differs from interpretation.

4. Create a flowchart for a program that calculates the factorial of a given number.

5. Write a function in your preferred programming language to calculate the area of a rectangle.