**Module 3 Introduction to OOPS Programming**

**1.Introduction to C++**

**THEORY EXERCISE:**

1. **What are the key differences between Procedural Programming and Object-Oriented Programming (OOP) ?**

* **Procedural Programming (POP) focuses on functions and procedures, following a top-down approach. In contrast, OOP organizes code into objects and classes, using a bottom-up approach. POP lacks features like inheritance and encapsulation, while OOP supports them, making code more modular and reusable. OOP also promotes data hiding and abstraction.**

**2.List and explain the main advantages of OOP over POP ?**

* **OOP offers better code reusability through inheritance and promotes data security via encapsulation. It simplifies complex programs by modeling real-world entities using objects. OOP supports modularity, making programs easier to debug, maintain, and scale. It also enhances code flexibility through polymorphism.**

**3. Explain the steps involved in setting up a C++ development environment.**

* **First, install a C++ compiler like GCC or MSVC. Then, install an IDE or text editor such as Code::Blocks, Dev C++, or VS Code. Configure the compiler in the IDE settings. Optionally, set environment variables if using command line tools. Finally, write and run your first program to test the setup.**

**4. What are the main input/output operations in C++? Provide examples?**

* **C++ uses cin for input and cout for output, both from the <iostream> header. For example: cpp , Copy Edit**

**int age;**

**cin >> age; // Input**

**cout << "Age is: " << age; // Output**