Department of Computer Science, New Campus

UNIVERSITY OF ENGINEERING AND TECHNOLOGY, LAHORE

Mid Exam – 2025 **Mid-Time: 40 minutes**

Designing and Implementing an Inventory Management System in C++

Objective:
The goal of this project is to design and implement an **Inventory Management System** in C++ for a store, which tracks product details such as name, price, and quantity. The system should allow users to manage inventory, including adding products.

Key Concepts Covered:

1. **(CLO2):**
    - **Constructors** are used to initialize products with attributes like name, price, and quantity.
    - The **copy constructor** ensures proper handling of deep copies when duplicating product objects.
    - **Destructors** manage memory cleanup when product objects are deleted.
    - **Constructor initializer lists** are used for efficient initialization of product attributes.
2. **(CLO2):**
    - **Separation of declaration and definition** for member functions improves code readability.
    - **Accessors** (getter functions) and **utility methods** are implemented to retrieve product details and update quantities.
    - Functions that accept and return **objects as arguments** and handle **cascaded calls** are demonstrated for better code organization.
3. **(CLO2, CLO3):**
    - **Static members** track the total number of products in the inventory.
    - **Const members** are introduced to set unmodifiable values, such as MAX_PRODUCTS, which defines the maximum number of products in the inventory.
    - Use of the **implicit this pointer** for accessing the calling object in member functions.
4. **(CLO3):**
    - **Dynamic memory allocation** is used to create product objects using the new operator, allowing flexibility in the system's memory management.
    - The **arrow (->) operator** is used to access object methods via pointers.

- The **delete operator** ensures that dynamically allocated memory is properly deallocated when no longer in use.

5. **(CLO3):**
   - **Operator overloading** is implemented to enhance the usability of the Product class. Operators such as + is overloaded for combining products.