

Problem Statement 1: "Smart Billing System for a Tech Store"

🔗 Objective:

Design a C++ program to manage customer purchases in a tech store, including item management, billing, and persistent record-keeping.

🔗 Requirements:

1. OOP Design

Create a class Item with attributes:

itemID, itemName, price, stockQuantity

Create a class Bill with:

billID, customerName, list of purchased items

2. Operator Overloading

Overload + to add an item to the bill

Overload << to display the bill details

3. Exception Handling

Throw exceptions for:

Negative price or quantity

Attempt to purchase more than available stock

Empty bill generation

4. File Handling

Save all bills to bills.txt

Load previous bills at startup

Append new bills without overwriting

📄 Problem Statement 2: "Student Gradebook & Analytics System"

🔗 Objective:

Build a C++ application to manage student records, calculate grades, and store data securely.

🔗 Requirements:

1. OOP Design

Class Student with:

rollNo, name, marks in 3 subjects

Class Gradebook to manage multiple students

2. Operator Overloading

Overload ++ to increment marks (e.g., bonus marks)

Overload > to compare students based on average marks

3. Exception Handling

Throw exceptions for:

Invalid marks (e.g., >100 or <0)

Duplicate roll numbers

Empty gradebook access

4. File Handling

Save student records to gradebook.txt

Load records at startup

Allow search by roll number and update marks