University of Information Technology and Sciences

Data Structures and Algorithms Lab (CSE0613212)

Project Title: Inventory Management System with C++

<u>Submitted by:</u>

Khandakar Borhan Uddin
Id: 0432310005101008
Batch: 55
Section: B1

<u>Submitted to:</u> (Saima Siddique Tashfia)

<u>Submission Date:</u> 26th May 2025

Project Title: Inventory Management System

Language: C++
 File Dependencies:

- `inventory.txt` (Data file for storing inventory records)

Overview

This program is a simple **Inventory Management System** written in C++. It allows users to **add**, **update**, **delete**, **search**, **display**, and **save** items in an inventory. Each item has the following attributes:

- ID (integer)

- Name (string)

- Quantity (integer)

- Price (float)

Main Features

1. **Persistent Storage**

   ○ The system loads inventory data from `inventory.txt` at startup.

   ○ Data is saved back to `inventory.txt` upon exiting the program.

2. **Menu-driven Interface**

   ○ A text-based menu allows users to interactively perform various inventory operations.

Functional Description

1. `loadFromFile()`

- Reads item data from `inventory.txt` and stores it in memory (in the `inventory` vector).

2. `saveToFile()`

- Writes the current `inventory` vector back into `inventory.txt`, overwriting previous data.

3. `addItem()`

- Prompts the user to enter new item details.

- Appends the new item to the inventory list.

4. `updateItem()`

- Asks for an item ID.

- If found, allows the user to modify the item's name, quantity, and price.

5. `deleteItem()`

- Asks for an item ID.

- If found, deletes the item from the inventory list.

6. `searchItem()`

- Asks for an item ID.

- If found, displays the details of the item.

7. `displayItems()`

- Lists all items in a tabular format with ID, Name, Quantity, and Price.

How to Use

1. **Compile the program** using a C++ compiler (e.g., g++) Or Use **CodeBlocks**:

   ```
   g++ -o inventory inventory.cpp
   ```

2. **Run the program**:

   ```
   ./inventory
   ```

3. **Use the menu** to perform actions such as adding, updating, or viewing items.

4. When done, select **option 6 (Save & Exit)** to save all changes to `inventory.txt`.

File Format (inventory.txt)

Each line in `inventory.txt` contains an item with space-separated fields:

```
<id> <name> <quantity> <price>
```

Example:

```
101 Apple 50 1.99
```

```
102 Banana 30 0.99
```

Note

- This code assumes valid input types from the user. Error handling for invalid input is not implemented.

- Item names must be single words (no spaces) due to how input is parsed using `cin`.