User Manual

05 - ONLINE MARKETPLACE INVENTORY & ORDER MATCHING ENGINE

[Bilal Azfar - Huzaifa Soomar]

1. Introduction

This document serves as the user manual for the Online Marketplace System developed as part of a Data Structures and Algorithms (DSA) final project. The system provides an interactive command-line interface that allows users to explore product categories, view inventory, place orders, execute order matching, and view product recommendations.

The system is designed for educational demonstration purposes and operates entirely in memory during runtime.

2. System Requirements

- Operating System: Windows, Linux, or macOS
- Compiler: C++ compiler with C++17 support
- Execution Environment: Command-line terminal

3. Starting the Program

After compiling the project, run the executable from the terminal:

```
> main.exe
```

Once started, the program loads initial data from text files and displays the main menu.

4. Main Menu Overview

```
================ ONLINE MARKETPLACE (DSA PROJECT) ================
1) Show category tree
2) Show inventory (all products)
3) Lookup product by ID
4) Place BUY order
5) Place SELL order
6) View best BUY/SELL (top of order book)
7) View order by ID
8) Run matching for a product
9) Demo recommendations for sample data
0) Exit
```

Users interact with the system by entering the corresponding menu number.

5. Menu Options Explained

5.1 Show Category Tree (Option 1)

Displays the hierarchical category structure starting from the ROOT category. This helps users understand how products are organized.

Example output:

```
ROOT
   Electronics
      Mobile
      Laptop
      Accessories
   Clothing
      Jackets
      Trousers
```

5.2 Show Inventory (Option 2)

Displays all products currently loaded in the catalog, including:

- Product ID
- Product name
- Category path
- Price

Stock is intentionally not displayed, as inventory quantities are not enforced in matching.

5.3 Lookup Product by ID (Option 3)

Allows the user to search for a product by entering its product ID.

- If the product exists, its details are displayed.
- If not, an appropriate message is shown.

This feature uses a HashMap for fast lookup.

5.4 Place BUY Order (Option 4)

Allows the user to place a BUY order by entering:

- Product ID
- Quantity
- Price

Validation rules:

- The product ID must exist in the catalog.
- If the product does not exist, the order is rejected.

Each order is assigned a unique order ID automatically.

5.5 Place SELL Order (Option 5)

Similar to BUY orders, but places a SELL order instead.

Users provide:

- Product ID
- Quantity
- Price

SELL orders represent offers to sell at the given price.

5.6 View Best BUY/SELL Orders (Option 6)

Displays:

- The highest-priority BUY order
- The highest-priority SELL order

Priority is determined by:

- Price
- Timestamp (for equal prices)

This reflects the current top of the order book.

5.7 View Order by ID (Option 7)

Allows users to search for an order using its order ID.

- If found, full order details are displayed.
- If not found, a message is shown.

This feature is powered by a HashMap index of orders.

5.8 Run Matching for a Product (Option 8)

Executes the order matching engine for a specific product ID.

Behavior:

- Matches BUY and SELL orders for the selected product
- Applies price–time priority
- Supports partial order execution
- Logs each successful match

Matching continues until no more valid matches exist for that product.

5.9 Demo Recommendations for Sample Data (Option 9)

Displays Top-K product recommendations based on historical co-purchase data.

- Recommendations are shown using product names
- Based on a weighted graph of frequently bought-together products

This demonstrates the recommendation engine functionality.

5.10 Exit (Option 0)

Terminates the program.
All runtime data is discarded, as the system does not persist state after execution.

6. Data Files

The system loads initial data from the following text files:

- `categories.txt` – Category hierarchy

- `products.txt` – Product catalog
- `orders.txt` – Initial orders
- `purchases.txt` – Co-purchase data for recommendations

These files are read at startup and are not modified during runtime.

7. Important Usage Notes

- The system is order-driven, not inventory-driven.
- Product stock exists as metadata but is not enforced during matching.
- Recommendations are based on historical data and do not update dynamically.
- All operations are performed in memory for a single execution session.

8. Intended Audience

This system is intended for:

- Students studying Data Structures and Algorithms
- Instructors evaluating DSA concepts
- Demonstration of custom data structure implementations

9. Conclusion

The Online Marketplace System provides a complete, interactive demonstration of key data structures and algorithms in a realistic problem setting. By following this user manual, users can effectively explore and test all major system features through the command-line interface.