**Restaurant Management System**

**Introduction:**

The Restaurant Management System is designed to automate and streamline various operations within a restaurant. This system is implemented using the concept of linked lists in the C programming language. Linked lists offer dynamic memory allocation and efficient data manipulation, making them suitable for managing restaurant operations like storing menu items, managing orders, and handling customer information.

**Objectives:**

* Create a menu structure using linked list nodes to store various menu items.
* Implement functionalities for adding, updating, and deleting menu items.
* Manage customer orders using linked list data structures.
* Provide functionalities to process orders, calculate bills, and manage table occupancy.
* Optimize memory usage and data retrieval for efficient system performance.

**System Design:**

**Menu Management**-

The menu items are stored in a linked list structure. Each node in the linked list represents a menu item and contains information such as item ID, name and price.

**Order Management**-

Another linked list structure is used to manage customer orders. Each node in this linked list represents an order and contains details such as table ordered items, quantity, and total price.

**Data Structures Used:**

* **Menu Linked List**: Contains nodes representing menu items.
* **Order Linked List**: Contains nodes representing customer orders.

**Functionality:**

1. **Menu Operations**:
   * Add Item: Adds a new menu item to the list.
   * Update Item: Modifies information about a menu item.
   * Delete Item: Removes a menu item from the list.
2. **Order Processing**:
   * Place Order: Allows customers to place their orders.
   * Calculate Bill: Calculates the total bill for a table based on ordered items.
   * Update Order: Modify or cancel an existing order.
3. **Table Management**:
   * Track Table Occupancy: Records which tables are occupied and available.
   * Manage Reservations: Allows booking tables in advance.

**Implementation Details:**

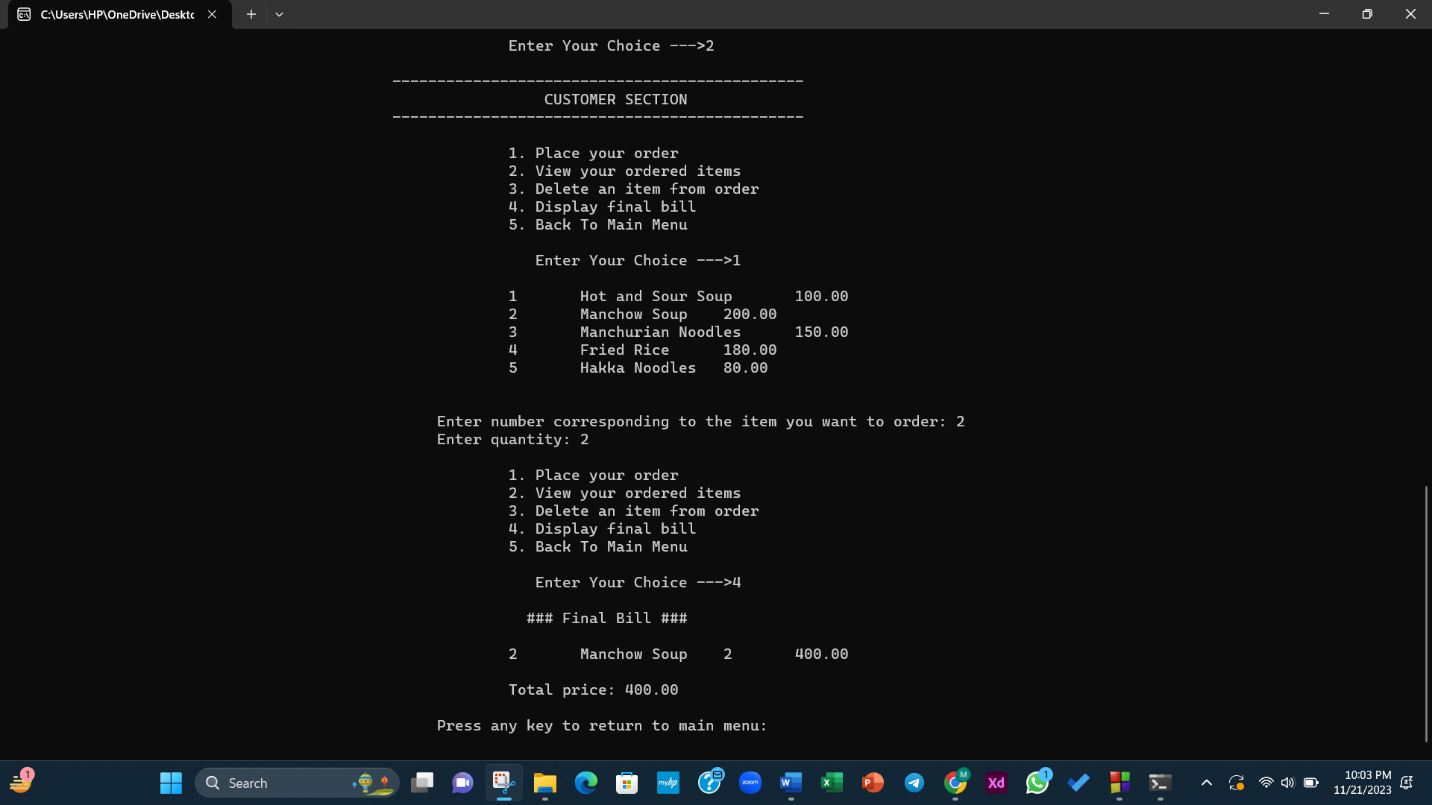
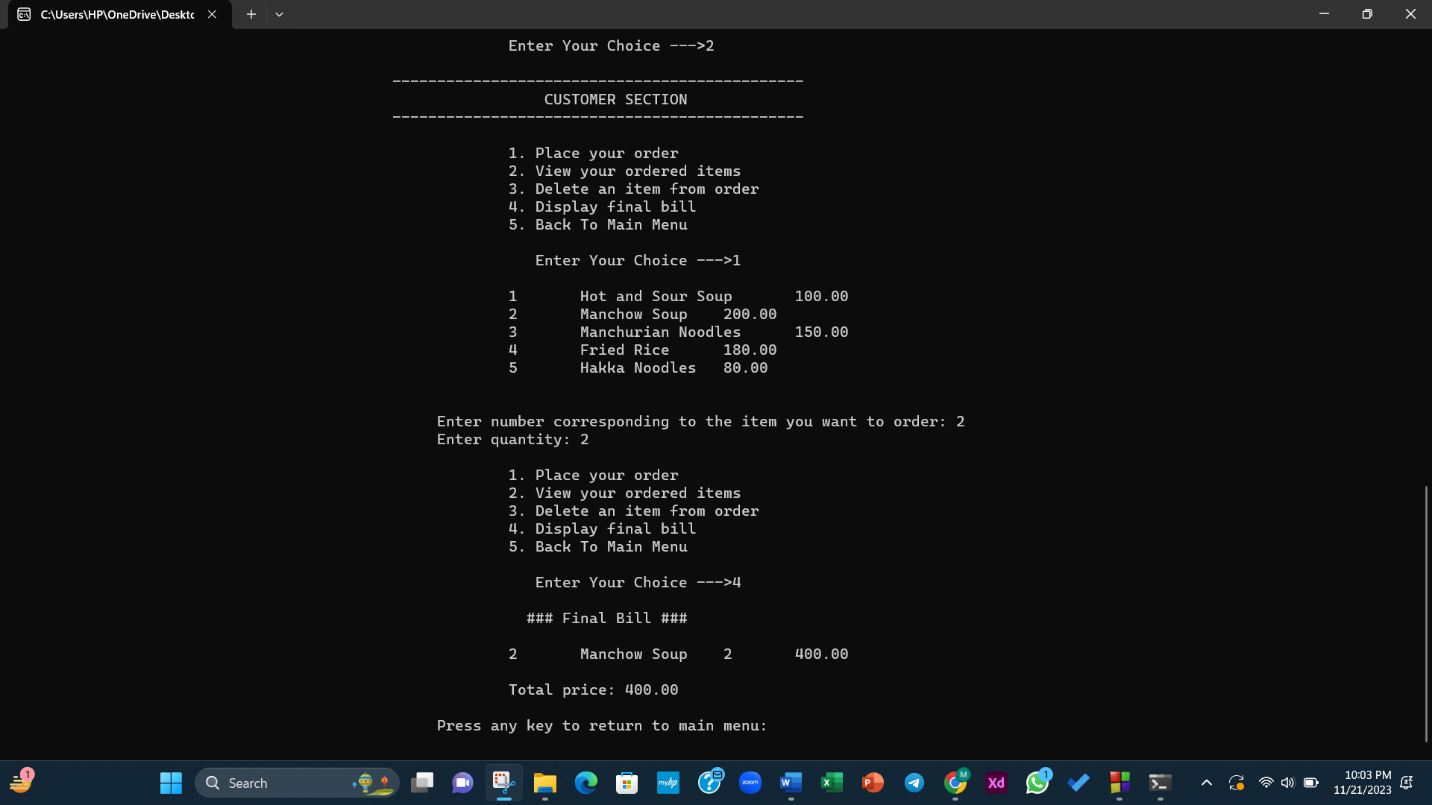
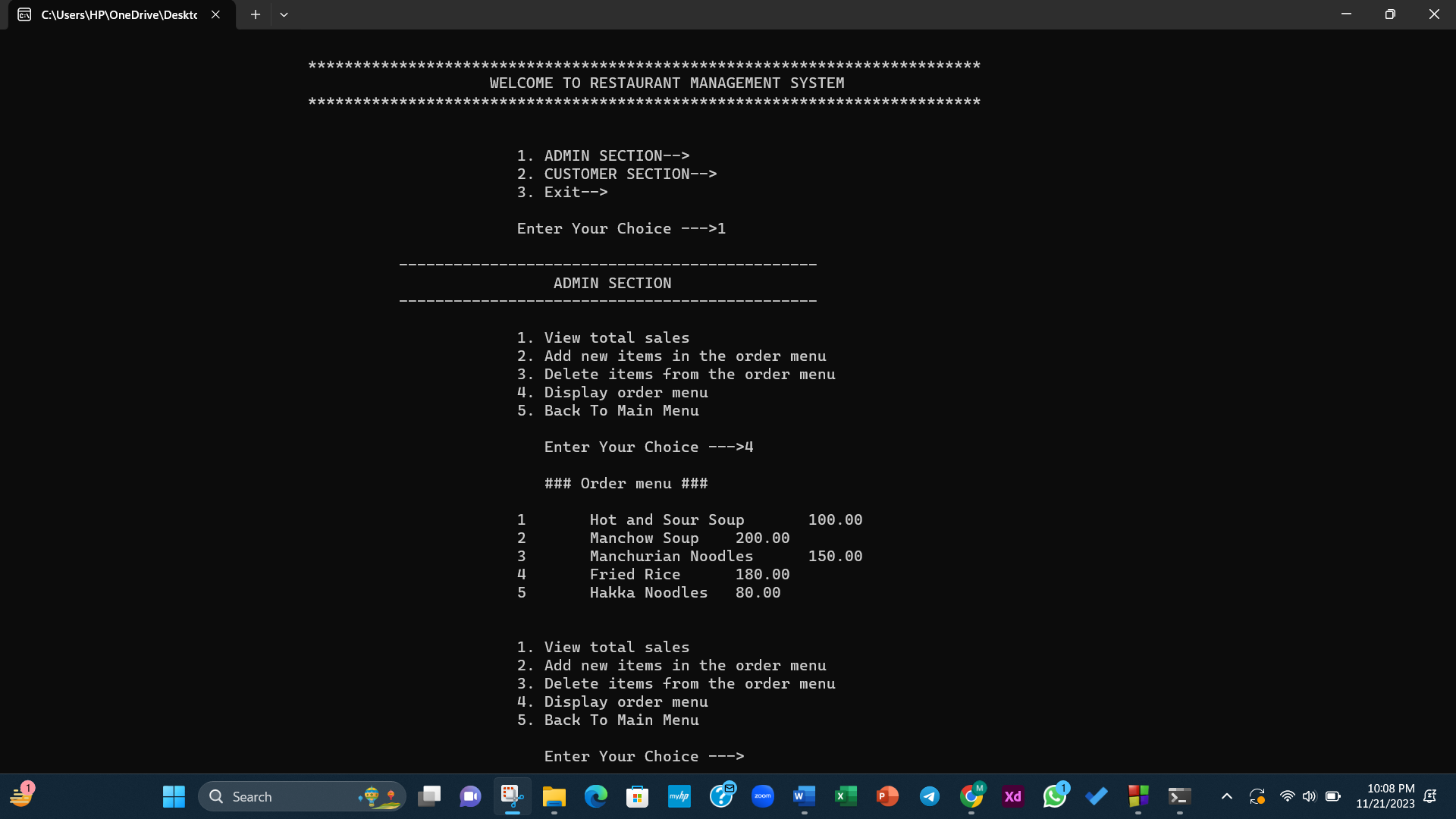
* **Language**: C programming language
* **Data Structures**: Linked lists for menu and orders.
* **Functions**: Modular approach for managing menus, orders, and tables.
* **Memory Management**: Dynamic memory allocation for linked list nodes.
* **User Interface**: Console-based interface for user interaction.

**Future Enhancements:**

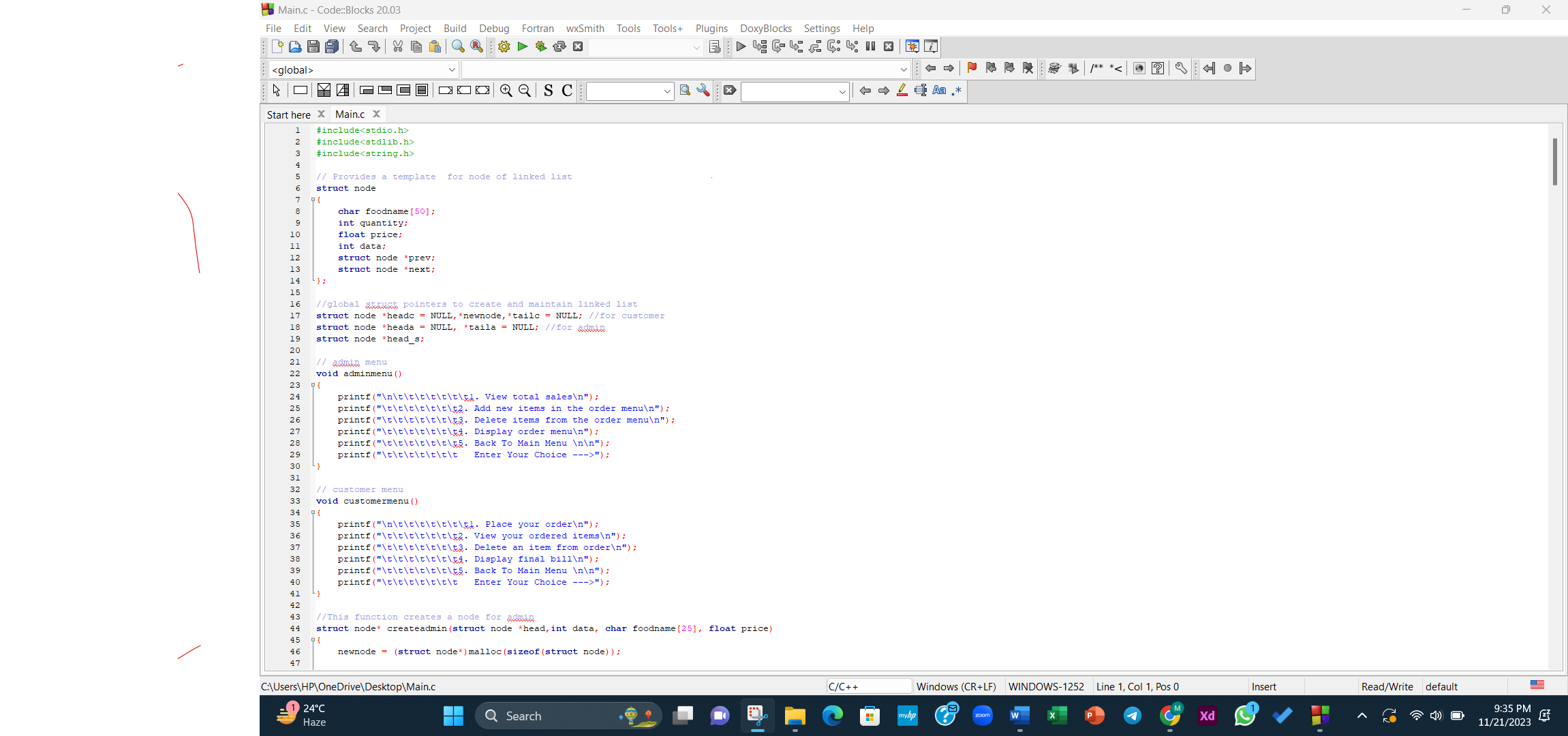
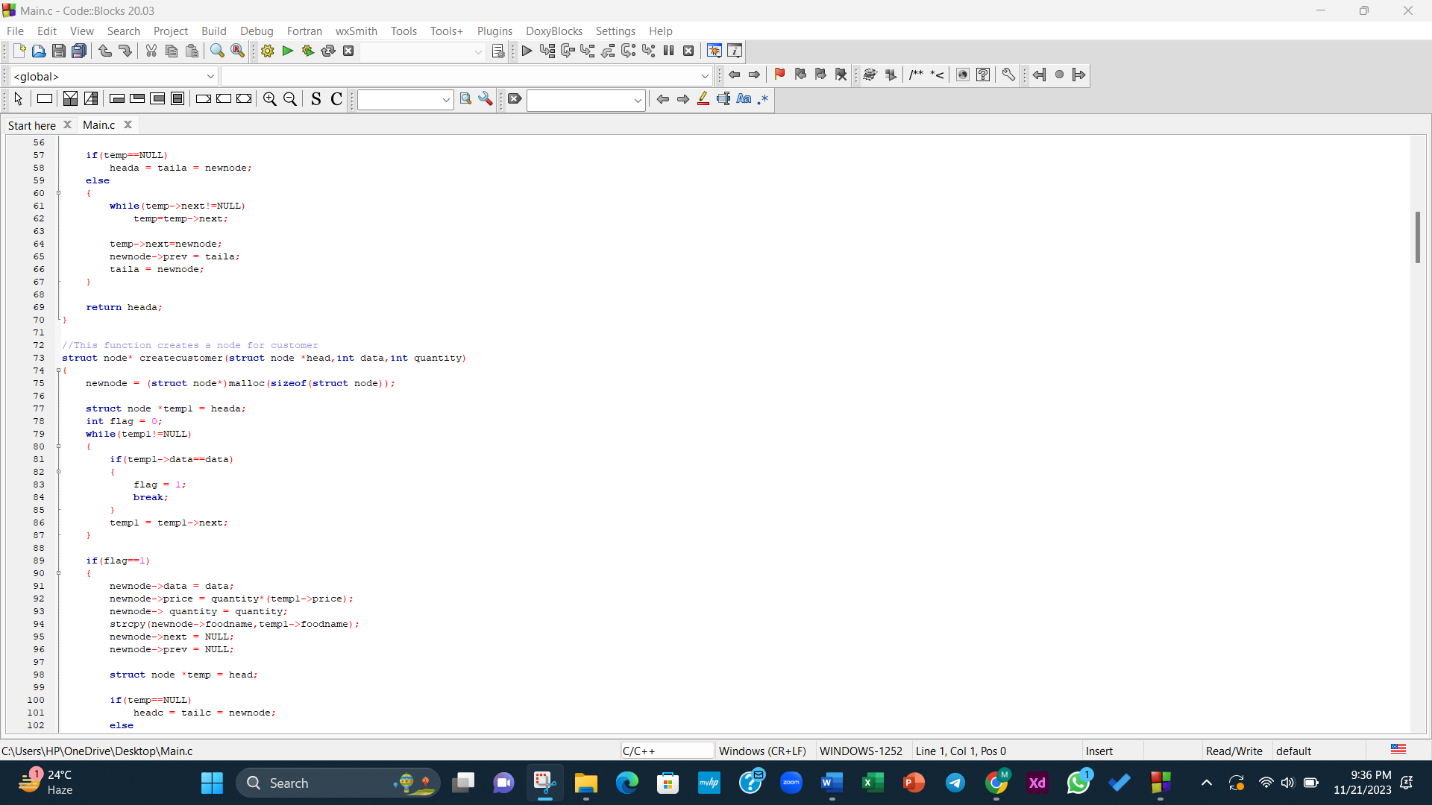
1. **User Interface Improvement**: Develop a graphical user interface for better user experience.
2. **Database Integration**: Implement a database system for persistent data storage.
3. **Billing and Payment Integration**: Integrate payment processing for customer bills.

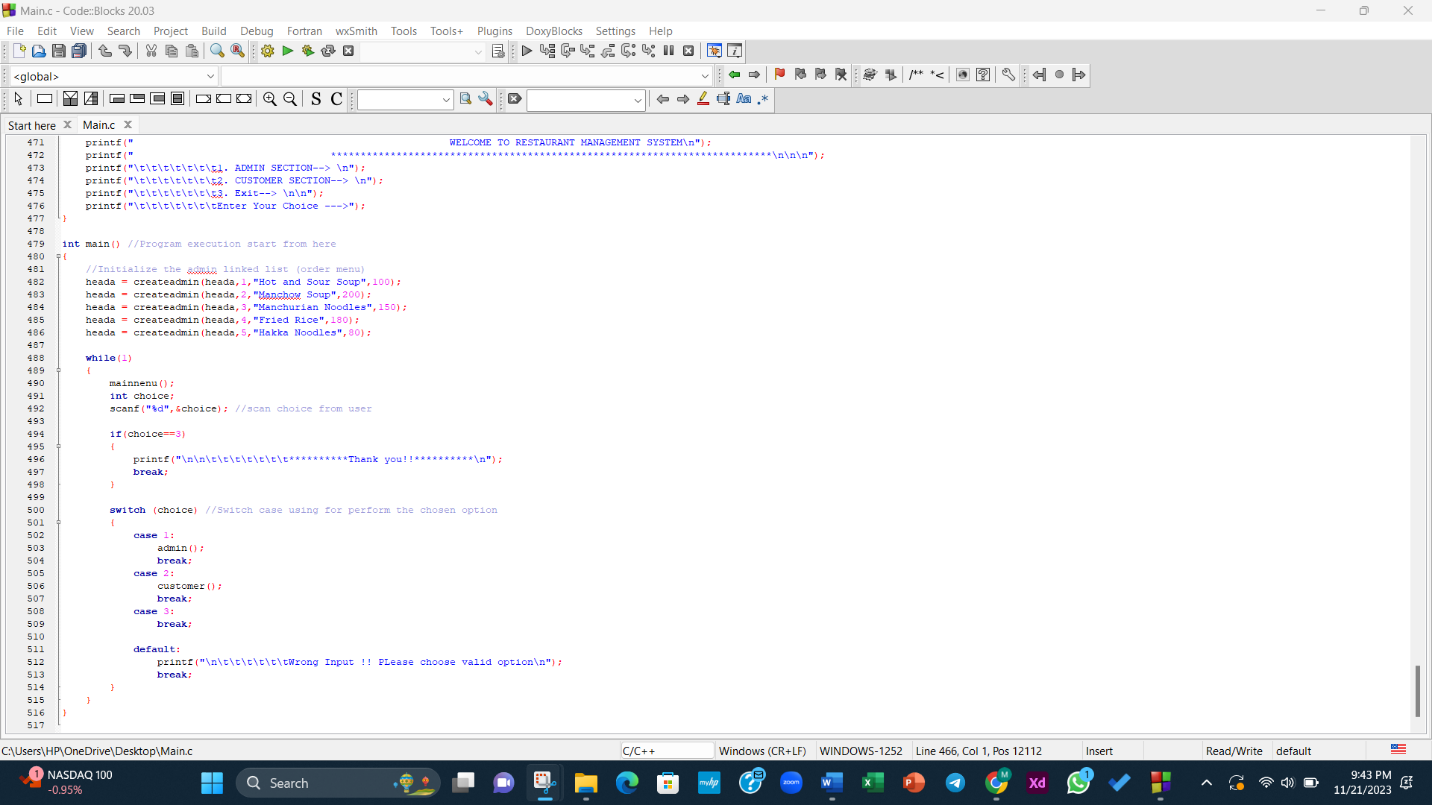
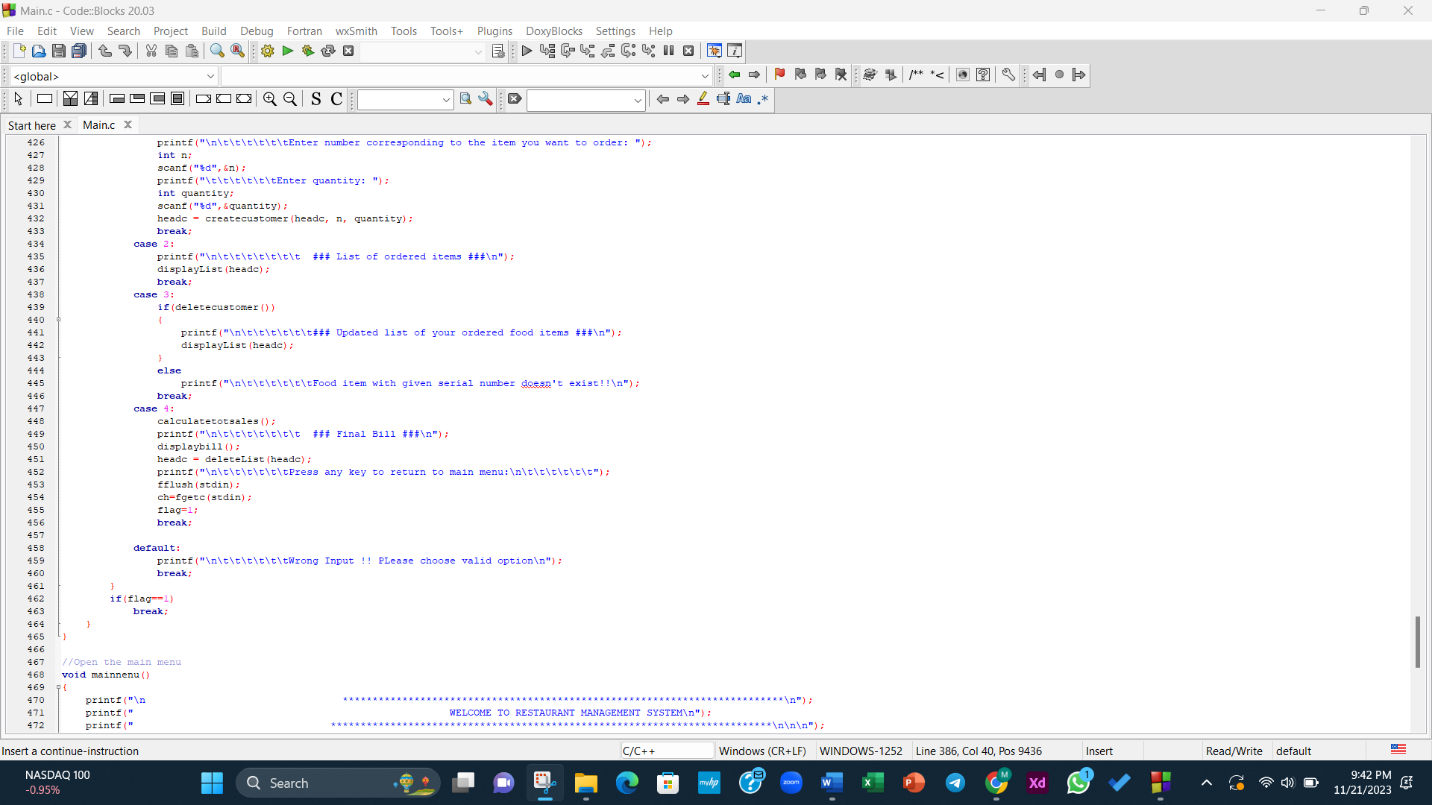
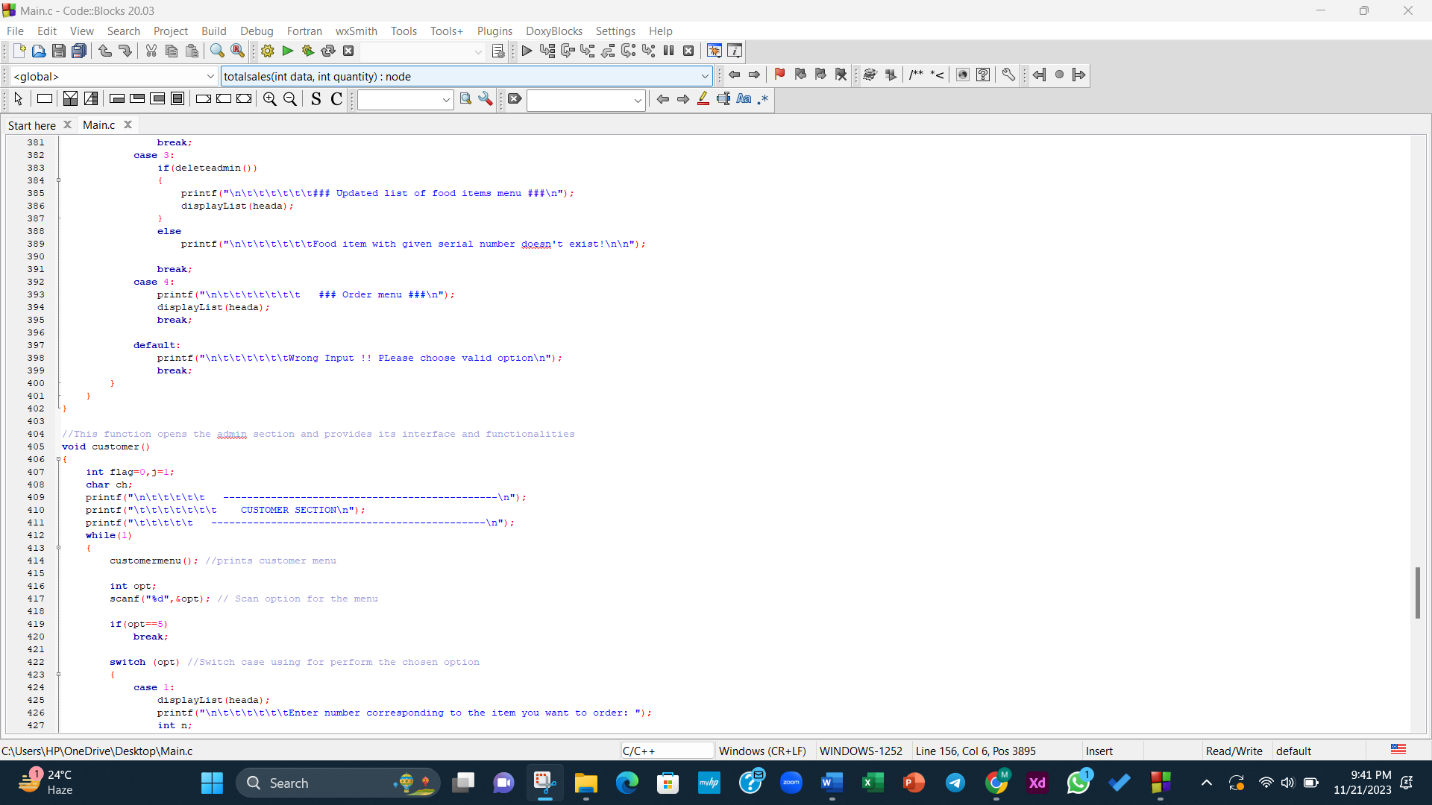
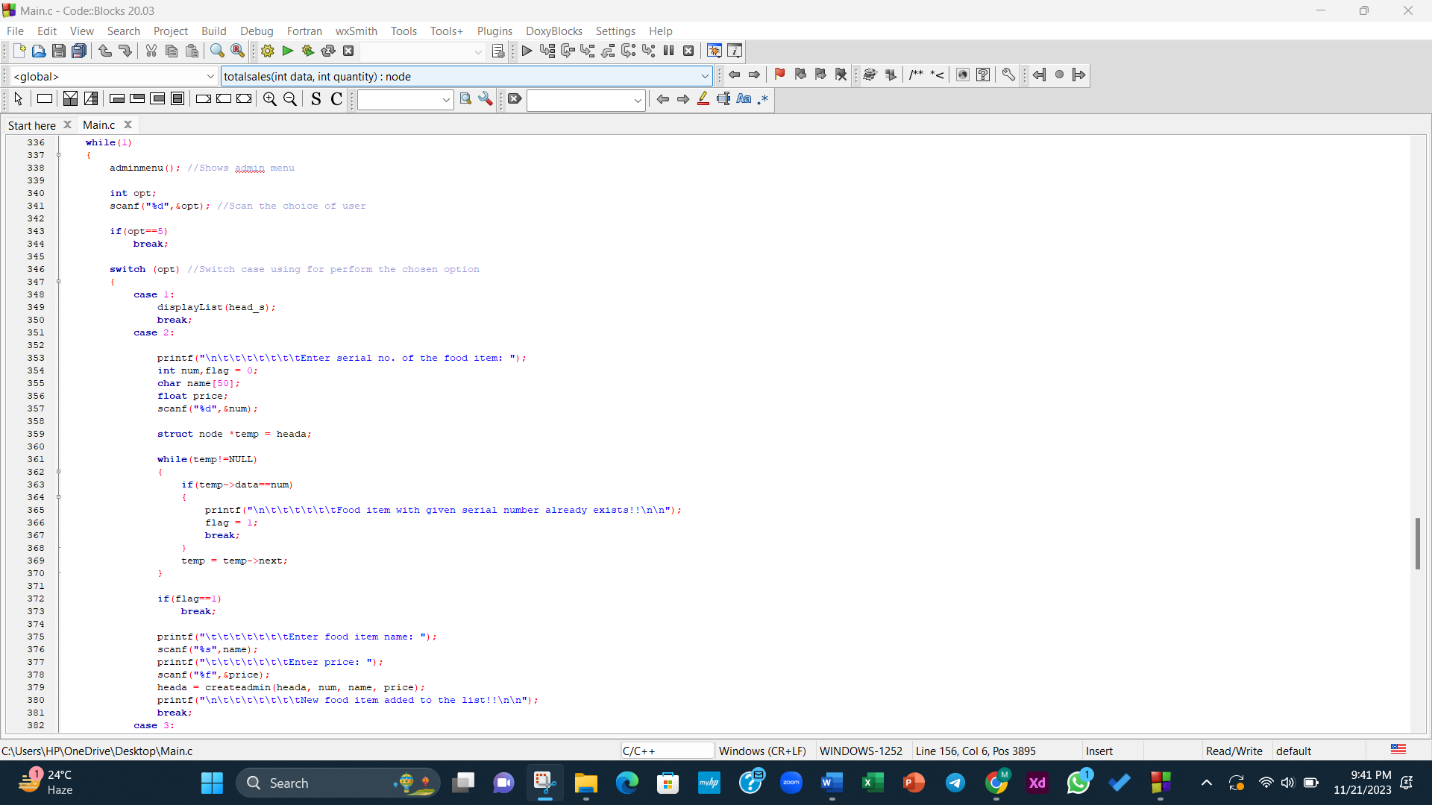
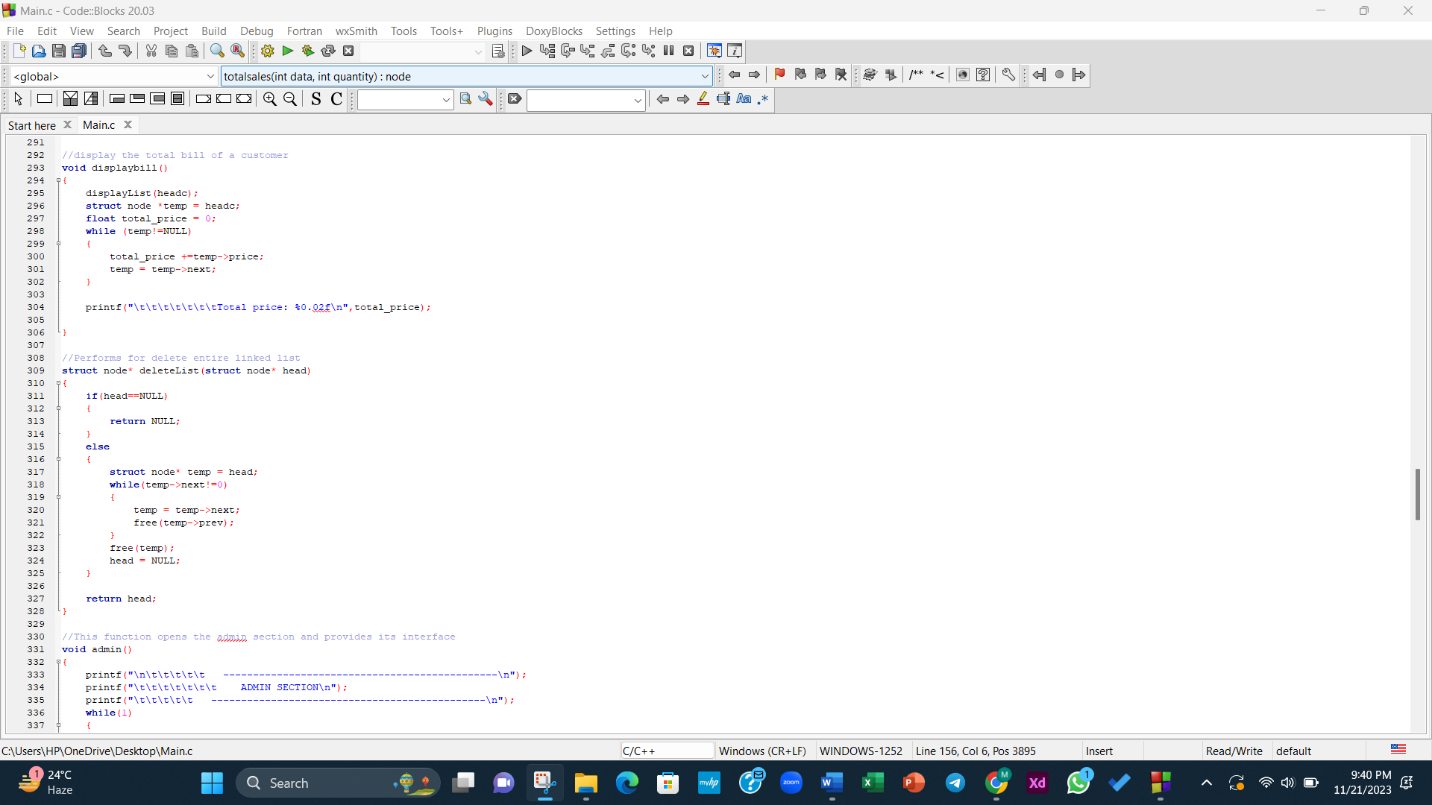
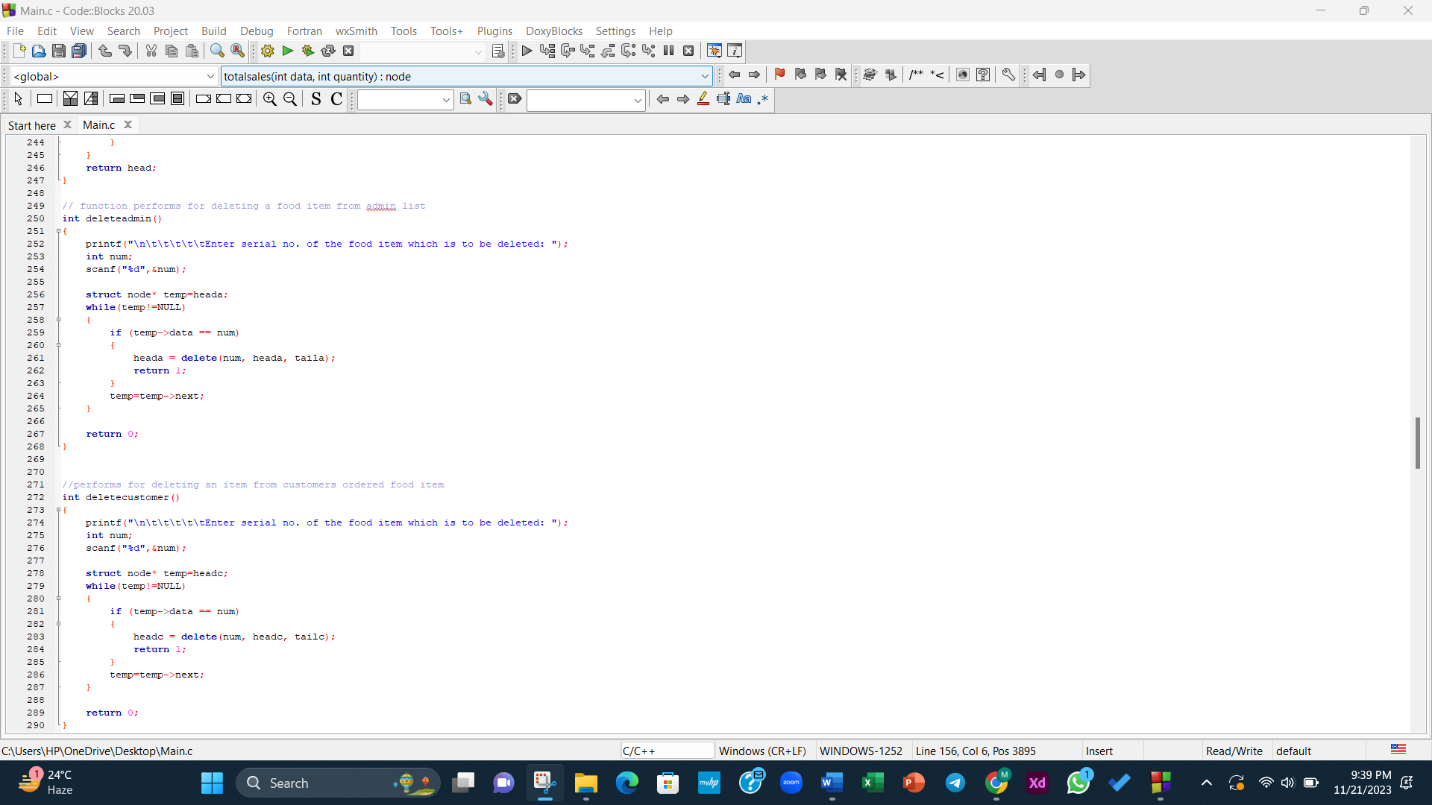
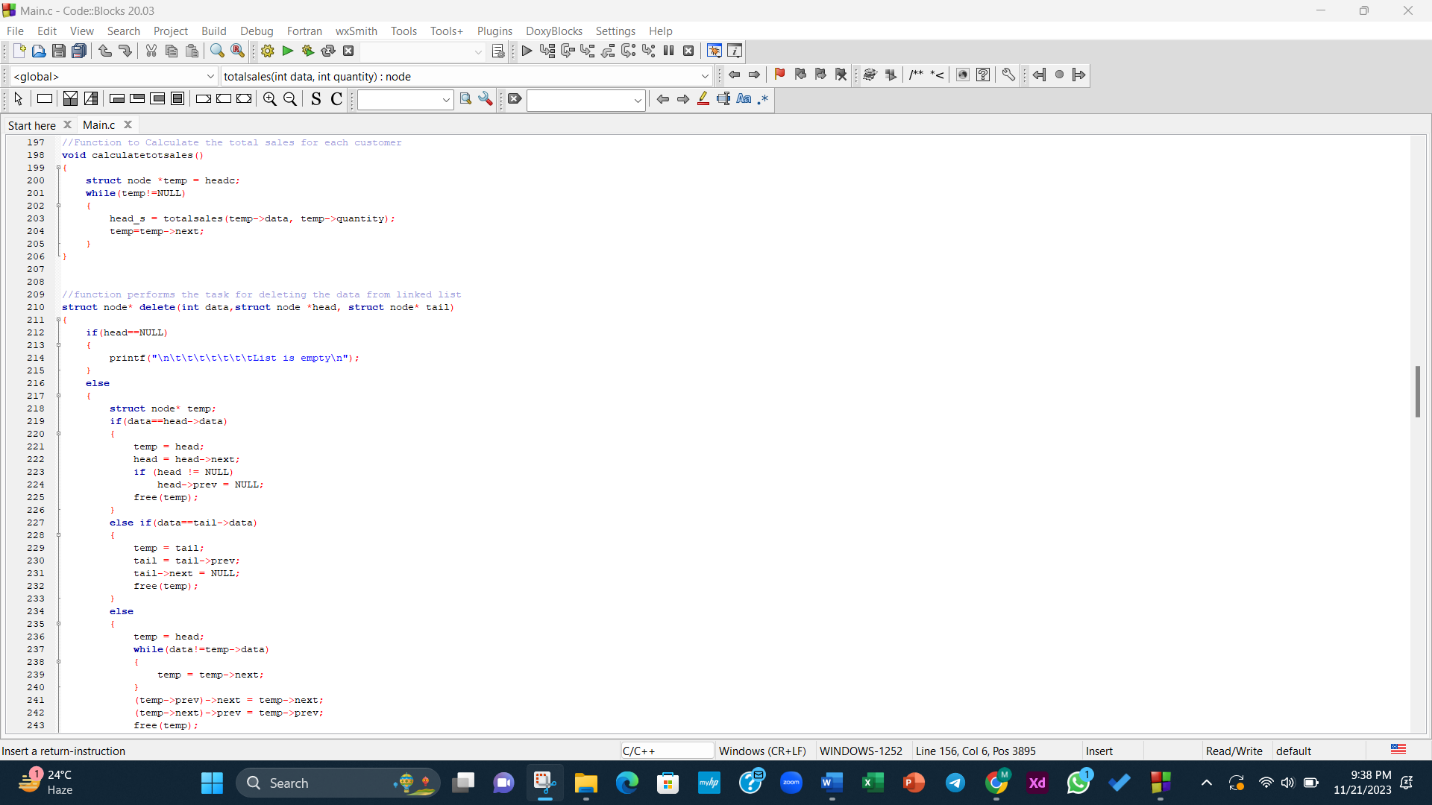
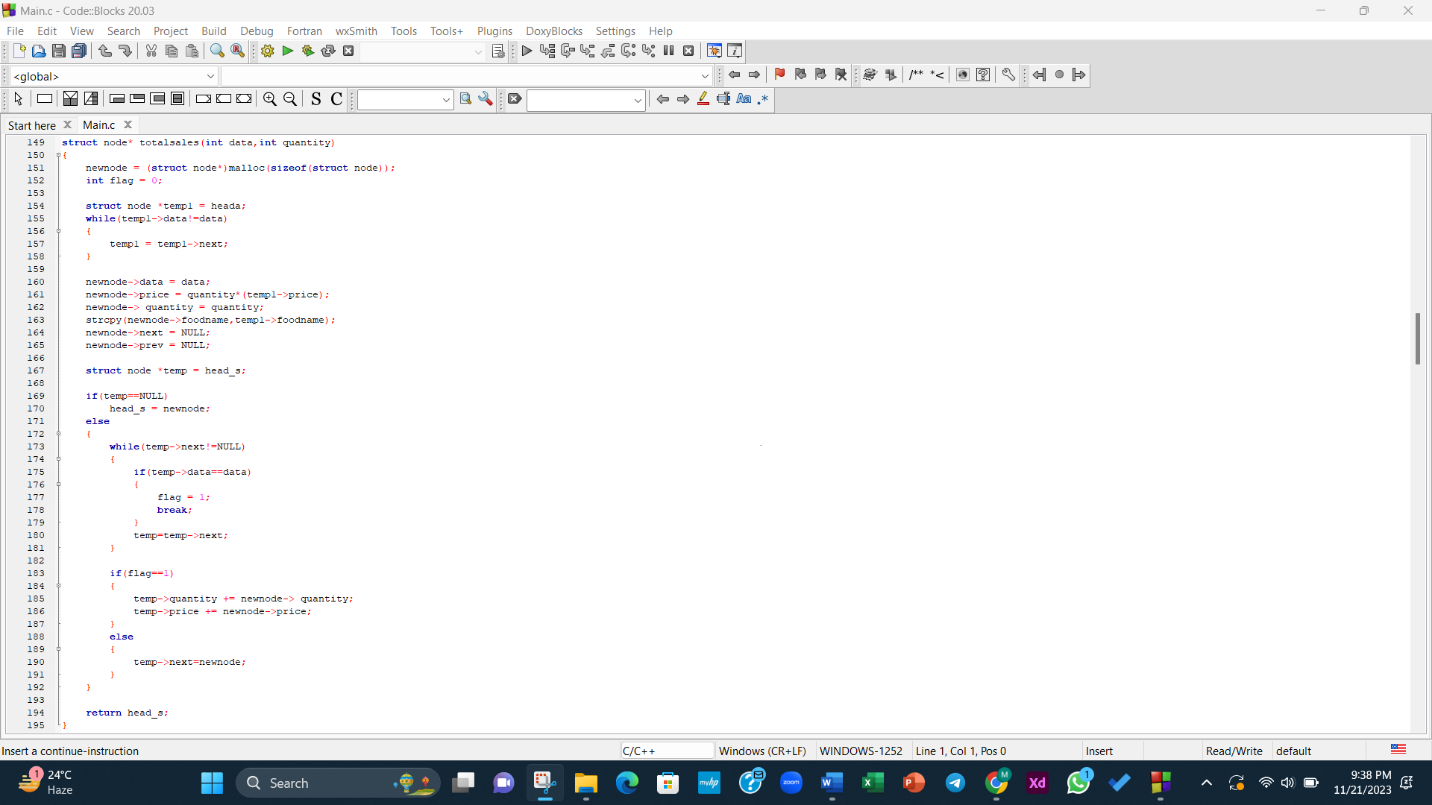
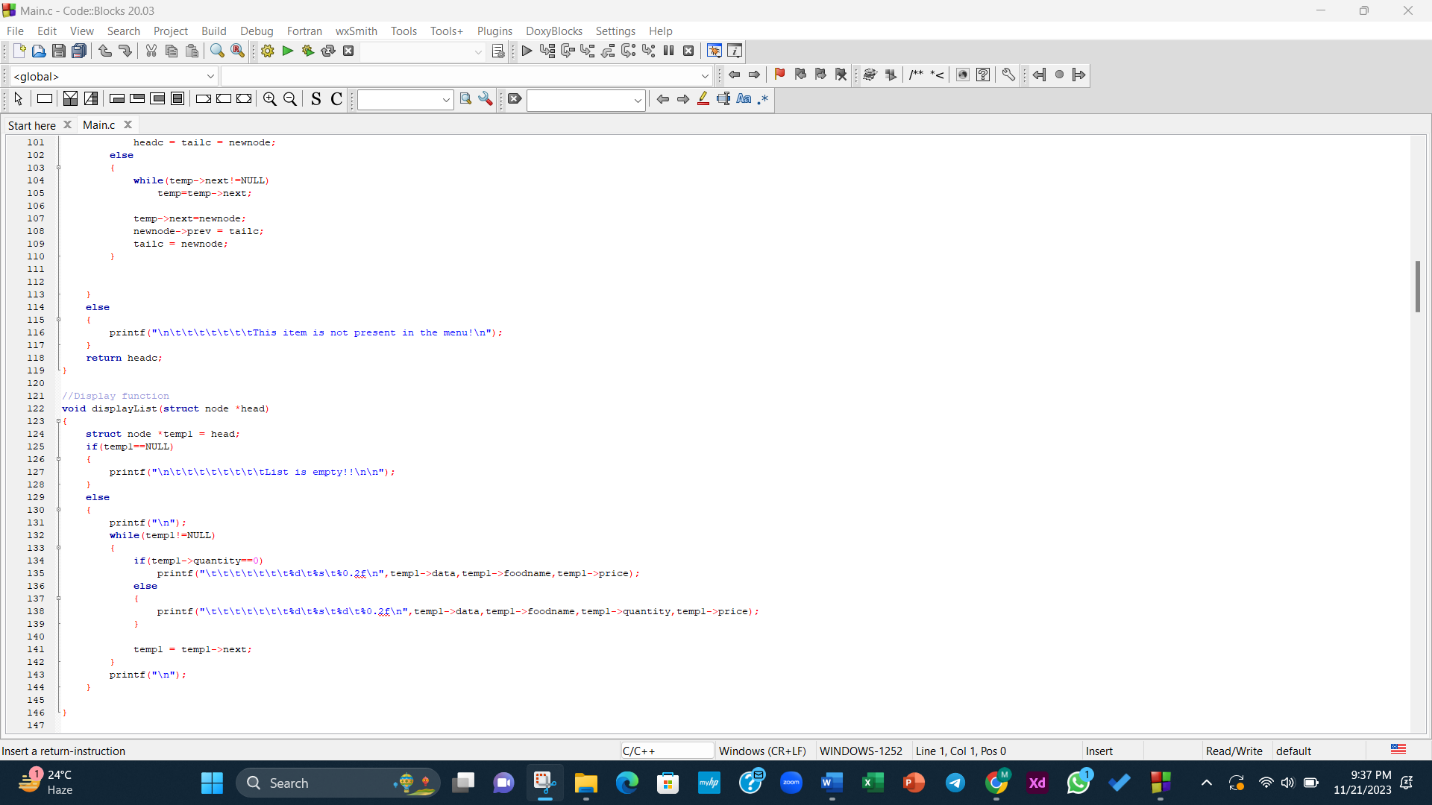
**Conclusion:**

The Restaurant Management System offers a structured approach to manage menu items, customer orders, and table occupancy efficiently. It provides a foundation for further enhancements and expansions to meet the evolving needs of the restaurant industry.

**Project Overview: **

**Source Code:**

****

****