Unified Food Ordering & Delivery System

1. Task Description

The project has been split between two students as follows:

Module	Student A	Student B
Core CLI & Navigation	CLI loops, input parsing	Utility functions, color macros
Menu Parsing	File I/O and parsing	API design
Cart & Order Logic	Cart struct & operations	Data structures definitions
File Persistence	CSV read/write	Format definitions
Customization Flow	Pizza customization prompts	Validation logic
History Display	Pretty-printing records	CSV utilities
Build & Documentation	Makefile	README & code comments

2. Data Structures Used by the Team

The following data structures are central to the application:

- typedef struct { char itemName[256]; double unitPrice; int quantity; char customizations[256]; } CartItem;

- typedef struct { CartItem items[MAX_CART_ITEMS]; int itemCount; } Cart;

- typedef struct { char name[100]; char address[200]; char phone[20]; } DeliveryInfo;

- typedef struct { int orderID; DeliveryInfo customer; Cart cart; double totalPrice; char dateTime[20]; } OrderRecord;

3. File Structure

- Menu files (*.txt): One line per item: $ (e.g., Pizza Diavola$12.50)

- Order history (orders_history.csv):

- Header: orderID,customerName,address,phone,dateTime,totalPrice,items

- Records: 1,"Alice","123 Main St","071234567","16/05/2025 14:22",45.50,"Pizza Diavola x1; Coke x2"

4. Interacting with Executables

Once built, the binary `foodapp` supports the following interfaces:

4.1 Interactive Mode

```
$ ./foodapp
=== FoodApp Main Menu ===
1) Browse Restaurants
2) View Cart
3) View Order History
4) Exit
Select an option: 1

--- Restaurants ---
1) El Diablo
2) Flavor Foundry
3) Pasta Palace
```

```
4) Back
Select a restaurant: 2

--- Flavor Foundry Menu ---
1) Pizza Diavola — $12.50
2) Burger Royale — $9.00
3) Caesar Salad — $7.50
4) Back
Select an item to add to cart: 1

Enter quantity: 2
Customize pizza? (y/n): y
  Select size (S/M/L): L
  Choose up to 3 toppings (comma-separated): Olives,Pepperoni
  Choose up to 2 sauces (comma-separated): BBQ
Added 2 × "Pizza Diavola (Size:L; Toppings:Olives,Pepperoni; Sauces:BBQ)" to cart.

Press [Enter] to continue…
```

4.2 Quick Order via Flags

```
$ ./foodapp --order "Flavor Foundry" \
            --item "Pizza Diavola" \
            --qty 2 \
            --custom "Size:L;Toppings:Olives,Pepperoni;Sauces:BBQ" \
            --name "Alice" \
            --address "123 Main St" \
            --phone "071234567"

Loading menu for Flavor Foundry…
Adding 2 × Pizza Diavola (Size:L;Toppings:Olives,Pepperoni;Sauces:BBQ) to cart.
Total: $25.00

Collecting delivery info…
  Name: Alice
  Address: 123 Main St
  Phone: 071234567

Order confirmed!
Order ID: 2
Date/Time: 16/05/2025 15:05
```

4.3 View Past History

```
$ ./foodapp --history

--- Order History ---
ID | Date/Time        | Customer | Items                                    | Total
---+------------------+----------+------------------------------------------+-------
1  | 16/05/2025 14:22 | Alice    | Pizza Diavola x2; Coke x1                 | $25.00
2  | 16/05/2025 15:05 | Alice    | Pizza Diavola x2                          | $25.00
```

4.4 Help Screen

```
$ ./foodapp --help

Usage: ./foodapp [OPTIONS]

Interactive mode (no flags):
  ./foodapp

Quick order:
  --order <RestaurantName>
  --item  <ItemName>
  --qty   <number>
  [--custom "<customization-string>"]
  --name    "<Customer Name>"
  --address "<Delivery Address>"
  --phone   "<Phone Number>"

View history:
  --history
Show this help:
```

```
--help
```