NORTHERN UNIVERSITY
B A N G L A D E S H
Knowledge for Innovation and Change

Department of CSE

Software Development I

CSE 1102

Project : Super shop management systems

SUPERSHOP
TAGLINE HERE

Submitted By - Group : E

Name : Md Kamruzzaman	Name : Gazi Shihab Hossain	Name : Md Imran Badsha
ID : 42250102220	ID : 42250102254	ID : 42250102262
Section : 2C	Section : 2C	Section : 2C

Introduction

Super shop management systems

This project, Super-Shop-Management-System, is developed in the C programming language. It simulates a shopping environment where customers can choose products, add them to a cart, update quantities, and apply discounts. The system also supports different payment methods, making it a simple yet practical example of applying C programming in real-life scenarios.

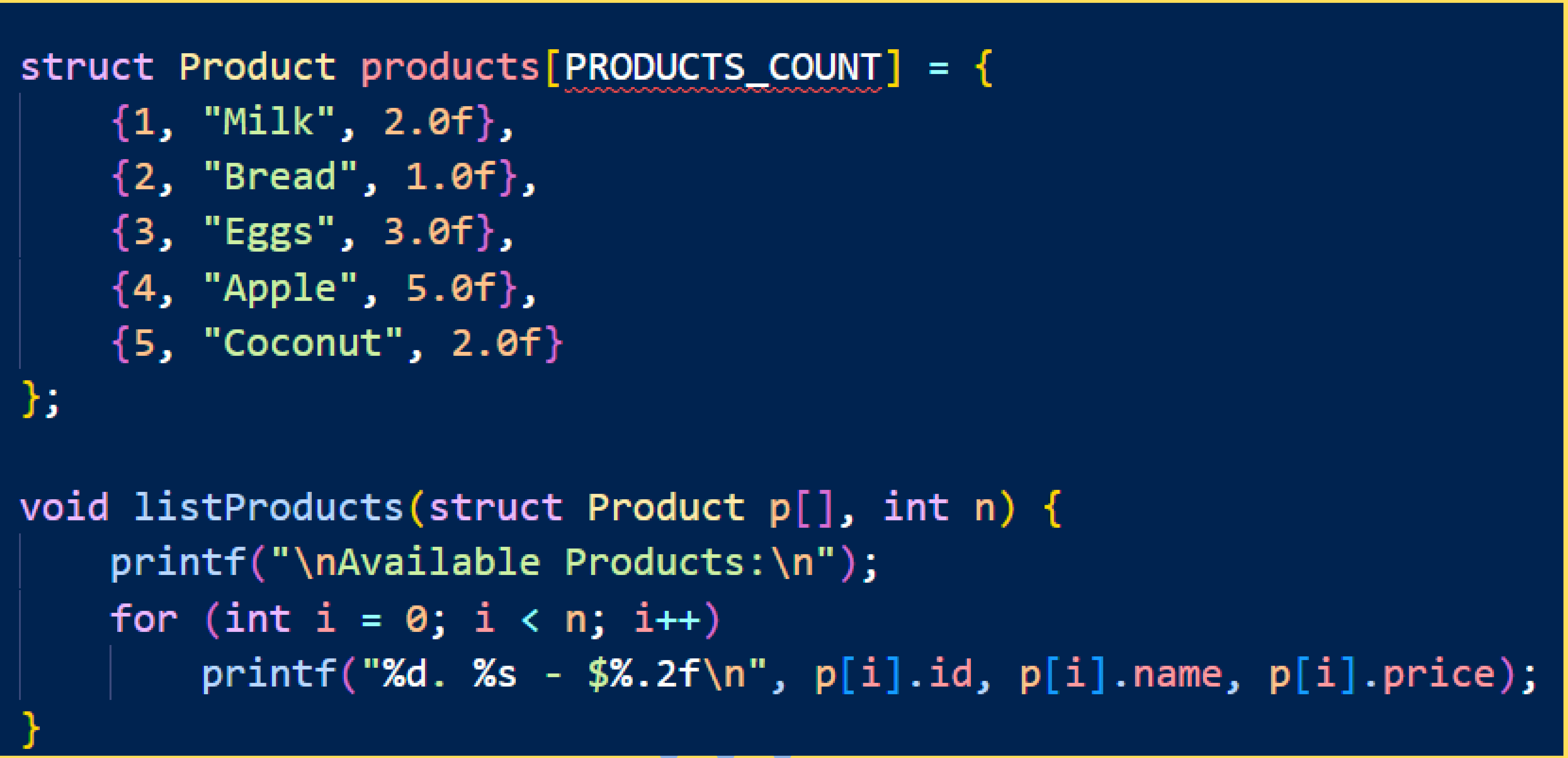
SUPERSHOP
TAGLINE HERE

Features

SUPERSHOP
TAGLINE HERE

- Product Catalog - Displays 5 pre-defined products with price.
- Add to Cart - Select product ID & quantity, adds or updates items.
- Display Cart - Shows all items currently in the cart.
- Edit Cart - Update or remove items before checkout.
- Total Quantity Count - Calculates total units in cart.
- Per-item Discount - 10% off if product quantity ≥ 10.
- Final Price Calculation - Computes total bill with discounts.
- Super Offer - 20% off if all 5 items and 15+ units are purchased.
- Payment Options - Bkash, Nagad, Rocket, BankCard, GPay.
- Checkout Summary- Displays final cart, discounts, and payment confirmation.

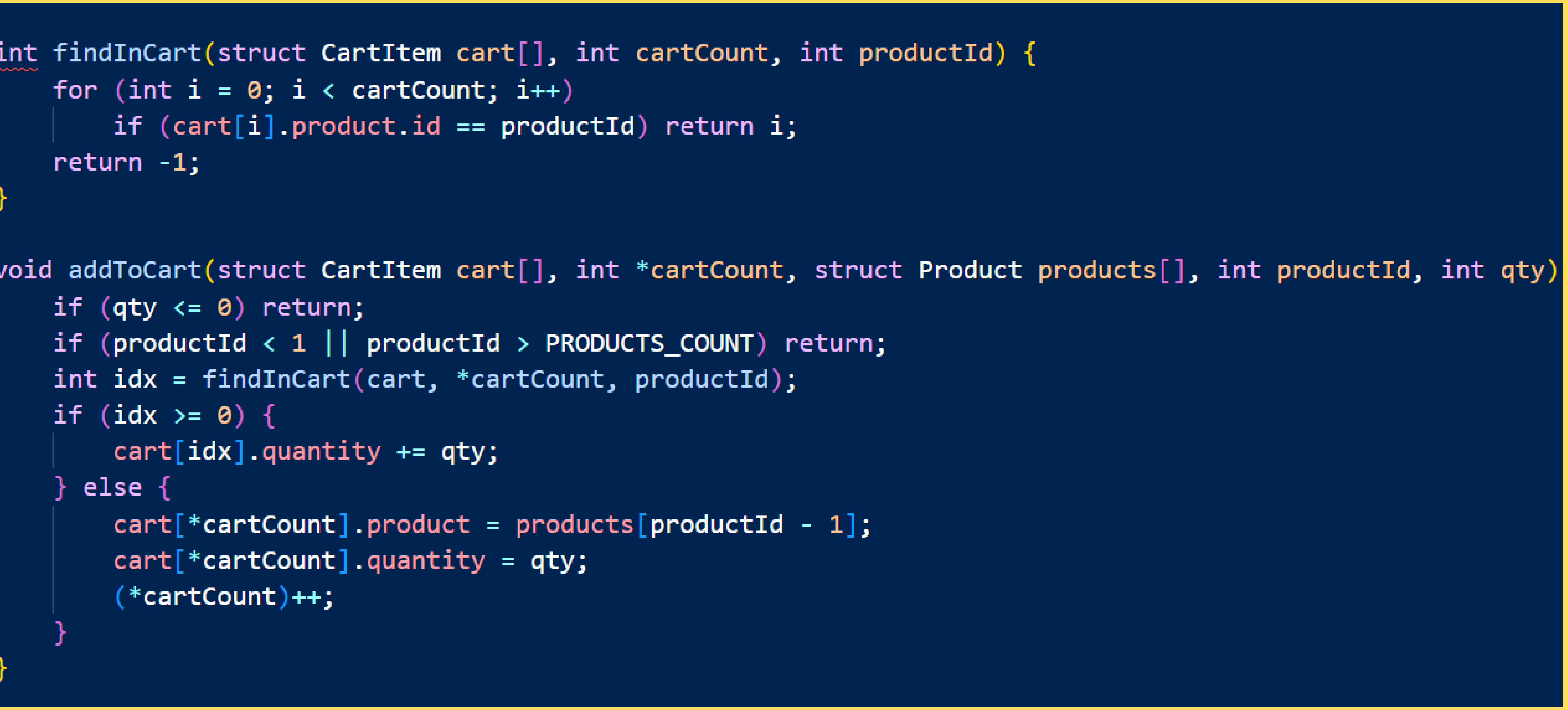
Product Catalog



```
struct Product products[PRODUCTS_COUNT] = {
    {1, "Milk", 2.0f},
    {2, "Bread", 1.0f},
    {3, "Eggs", 3.0f},
    {4, "Apple", 5.0f},
    {5, "Coconut", 2.0f}
};

void listProducts(struct Product p[], int n) {
    printf("\nAvailable Products:\n");
    for (int i = 0; i < n; i++)
        printf("%d. %s - $%.2f\n", p[i].id, p[i].name, p[i].price);
}
```

```
PRODUCT CATALOG
1. Milk - $2.00
2. Bread - $1.00
3. Eggs - $3.00
4. Apple - $5.00
5. Coconut - $2.00
```

ADD TO CART

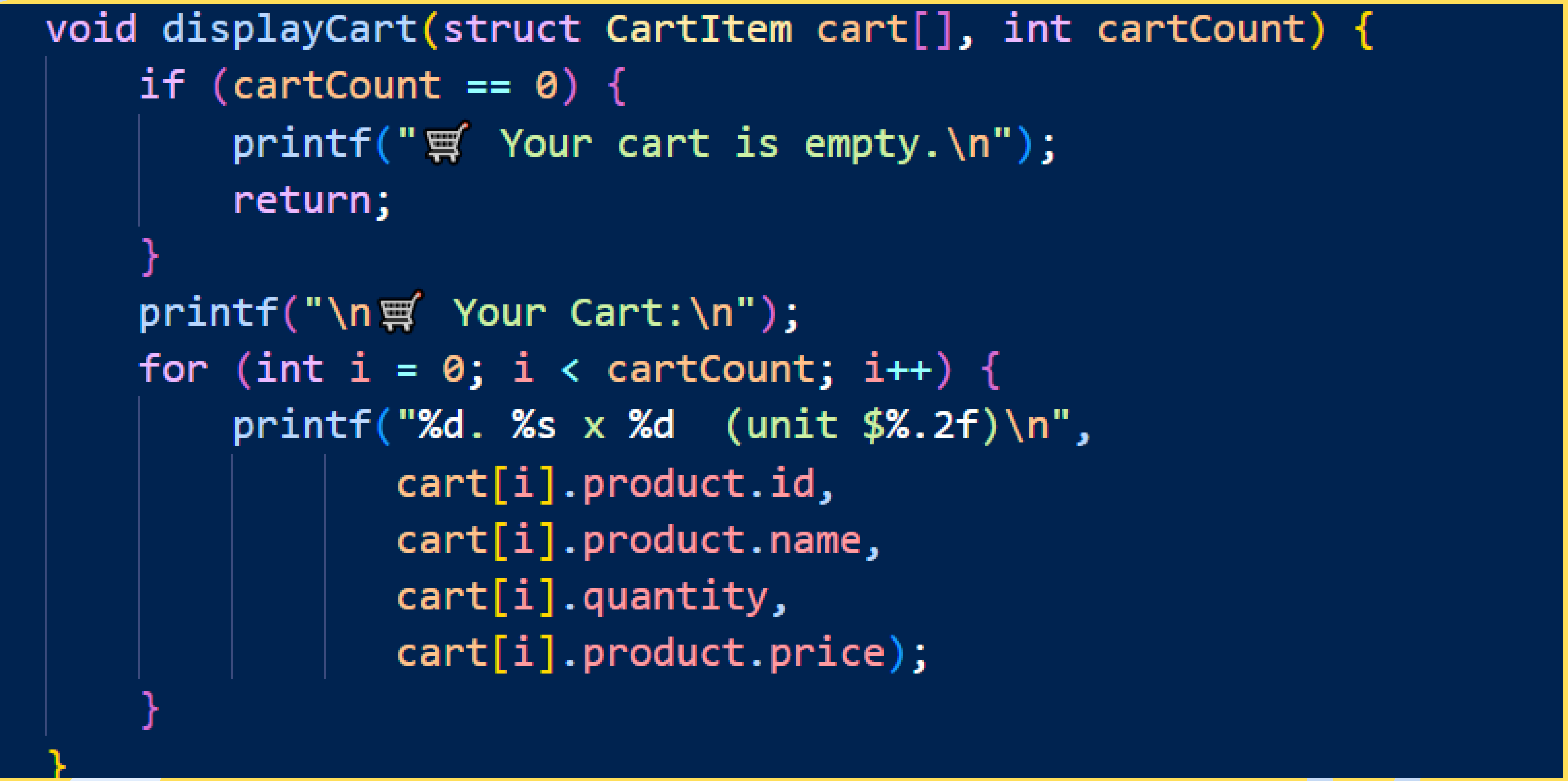


```c
int findInCart(struct CartItem cart[], int cartCount, int productId) {
    for (int i = 0; i < cartCount; i++)
        if (cart[i].product.id == productId) return i;
    return -1;
}

void addToCart(struct CartItem cart[], int *cartCount, struct Product products[], int productId, int qty)
    if (qty <= 0) return;
    if (productId < 1 || productId > PRODUCTS_COUNT) return;
    int idx = findInCart(cart, *cartCount, productId);
    if (idx >= 0) {
        cart[idx].quantity += qty;
    } else {
        cart[*cartCount].product = products[productId - 1];
        cart[*cartCount].quantity = qty;
        (*cartCount)++;
    }
}
```

ADD TO CART

```
🛒 Cart contents:
ID  Name     Qty  Unit($)  LineTotal($)
----------------------------------------
 1  Milk      2    2.00     4.00
 2  Bread    12    1.00    12.00

Total units: 14
Has all products? No
Qualifies Super Offer? No

Original total: $16.00
Per-item discount amount: $1.20
Total after discounts: $14.80
```

Display Cart

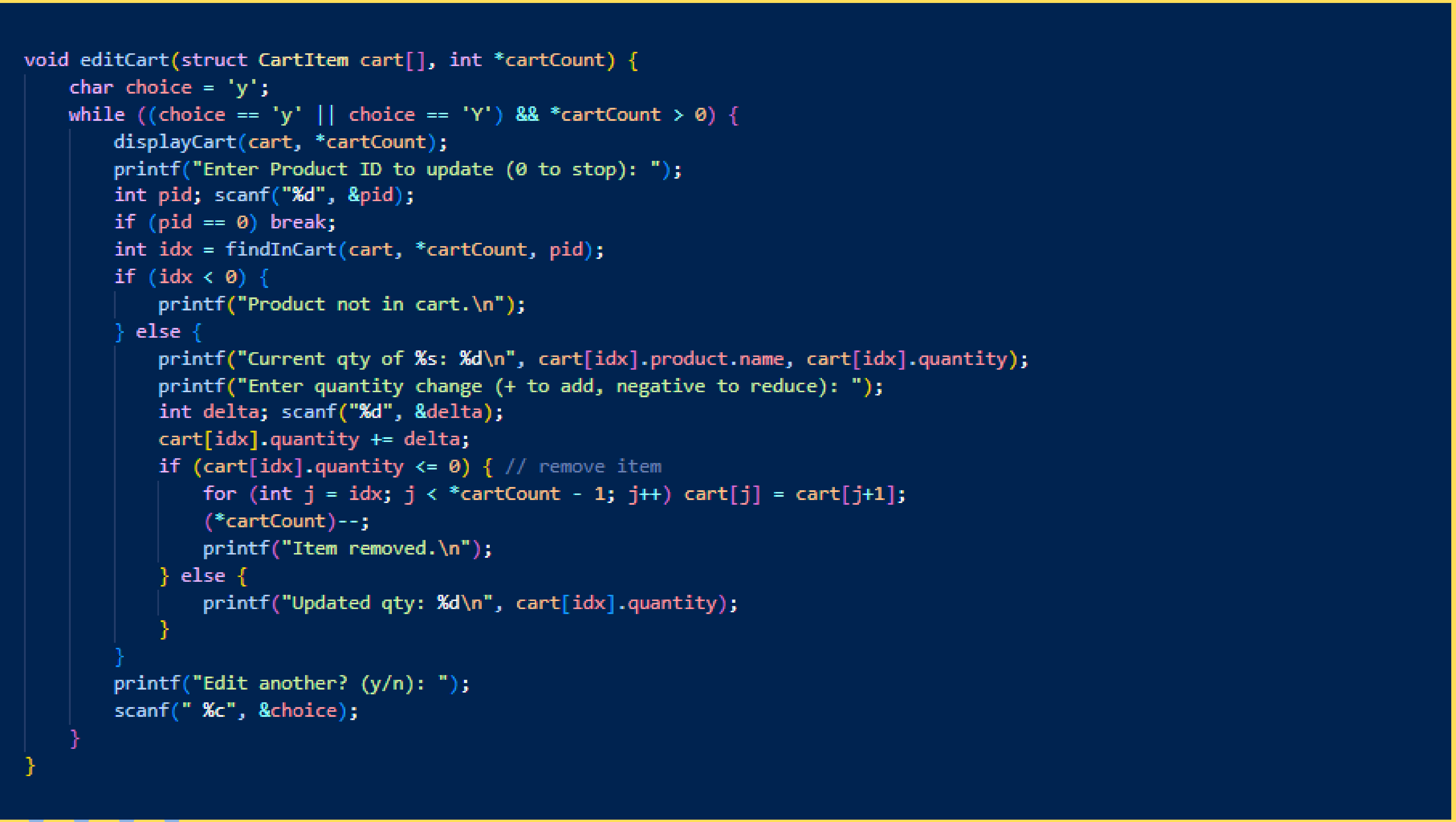


```c
void displayCart(struct CartItem cart[], int cartCount) {
    if (cartCount == 0) {
        printf("🛒 Your cart is empty.\n");
        return;
    }
    printf("\n🛒 Your Cart:\n");
    for (int i = 0; i < cartCount; i++) {
        printf("%d. %s x %d  (unit $%.2f)\n",
               cart[i].product.id,
               cart[i].product.name,
               cart[i].quantity,
               cart[i].product.price);
    }
}
```

Display Cart

```
🛒 Cart contents:
1  Milk     2   2.00   4.00
2  Bread   12   1.00  12.00
3  Eggs     1   3.00   3.00
4  Apple    1   5.00   5.00
5  Coconut  1   2.00   2.00

Total units now: 17
Has all products? Yes
Qualifies Super Offer? Yes

Original total: $26.00
Discount (super offer): $5.20
Total after discounts: $20.80
```

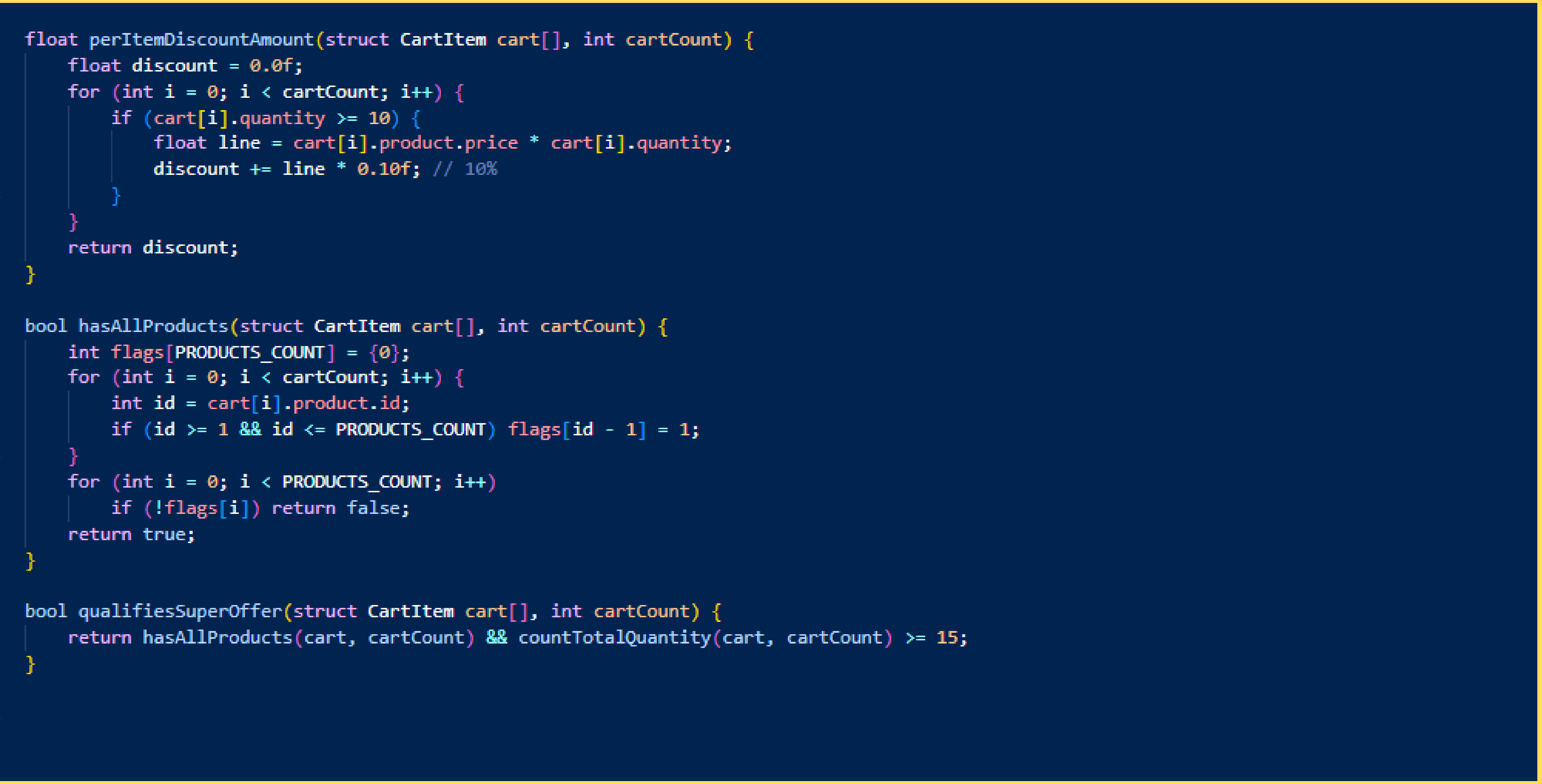
Edit Cart



```c
void editCart(struct CartItem cart[], int *cartCount) {
    char choice = 'y';
    while ((choice == 'y' || choice == 'Y') && *cartCount > 0) {
        displayCart(cart, *cartCount);
        printf("Enter Product ID to update (0 to stop): ");
        int pid; scanf("%d", &pid);
        if (pid == 0) break;
        int idx = findInCart(cart, *cartCount, pid);
        if (idx < 0) {
            printf("Product not in cart.\n");
        } else {
            printf("Current qty of %s: %d\n", cart[idx].product.name, cart[idx].quantity);
            printf("Enter quantity change (+ to add, negative to reduce): ");
            int delta; scanf("%d", &delta);
            cart[idx].quantity += delta;
            if (cart[idx].quantity <= 0) { // remove item
                for (int j = idx; j < *cartCount - 1; j++) cart[j] = cart[j+1];
                (*cartCount)--;
                printf("Item removed.\n");
            } else {
                printf("Updated qty: %d\n", cart[idx].quantity);
            }
        }
        printf("Edit another? (y/n): ");
        scanf(" %c", &choice);
    }
}
```

```
EDIT CART
(reduce Bread by 3 units)
🛒 Cart contents:
ID  Name     Qty  Unit($)  LineTotal($)
----------------------------------------
 1  Milk      2    2.00      4.00
 2  Bread     9    1.00      9.00
 3  Eggs      1    3.00      3.00
 4  Apple     1    5.00      5.00
 5  Coconut   1    2.00      2.00

After edit - total units: 14
Qualifies Super Offer? No
Original: $23.00  Discount: $0.90  Final: $22.10
```

Per-item Discount



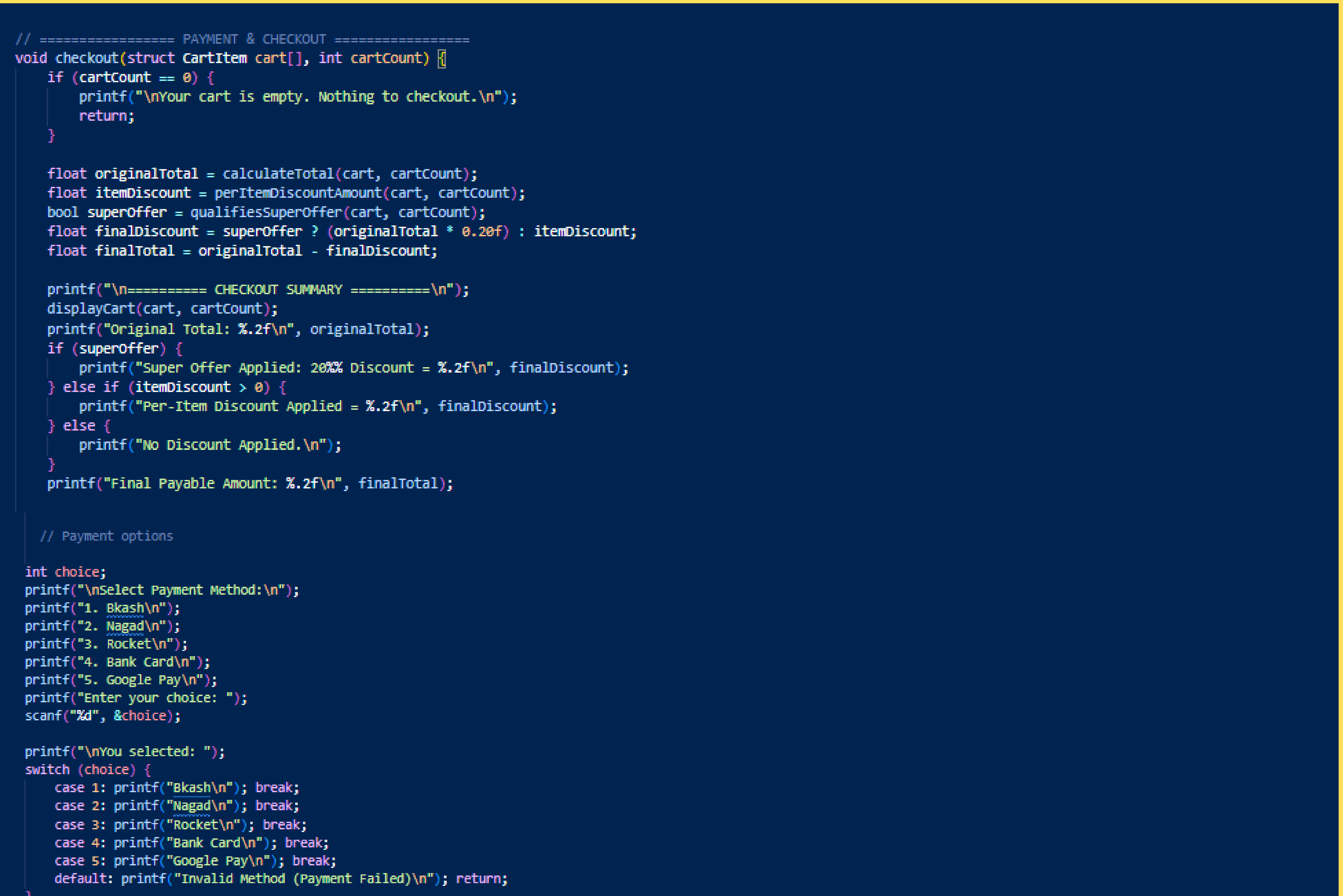
```c
float perItemDiscountAmount(struct CartItem cart[], int cartCount) {
    float discount = 0.0f;
    for (int i = 0; i < cartCount; i++) {
        if (cart[i].quantity >= 10) {
            float line = cart[i].product.price * cart[i].quantity;
            discount += line * 0.10f; // 10%
        }
    }
    return discount;
}

bool hasAllProducts(struct CartItem cart[], int cartCount) {
    int flags[PRODUCTS_COUNT] = {0};
    for (int i = 0; i < cartCount; i++) {
        int id = cart[i].product.id;
        if (id >= 1 && id <= PRODUCTS_COUNT) flags[id - 1] = 1;
    }
    for (int i = 0; i < PRODUCTS_COUNT; i++)
        if (!flags[i]) return false;
    return true;
}

bool qualifiesSuperOffer(struct CartItem cart[], int cartCount) {
    return hasAllProducts(cart, cartCount) && countTotalQuantity(cart, cartCount) >= 15;
}
```

🎁 Per-item discounts

applied where eligible
(10% on 10+ units)

Original total: $23.00
Discount: $0.90
Amount to pay: $22.10

PAYMENT & CHECKOUT



```c
// ================= PAYMENT & CHECKOUT =================
void checkout(struct CartItem cart[], int cartCount) {
    if (cartCount == 0) {
        printf("\nYour cart is empty. Nothing to checkout.\n");
        return;
    }

    float originalTotal = calculateTotal(cart, cartCount);
    float itemDiscount = perItemDiscountAmount(cart, cartCount);
    bool superOffer = qualifiesSuperOffer(cart, cartCount);
    float finalDiscount = superOffer ? (originalTotal * 0.20f) : itemDiscount;
    float finalTotal = originalTotal - finalDiscount;

    printf("\n========== CHECKOUT SUMMARY ==========\n");
    displayCart(cart, cartCount);
    printf("Original Total: %.2f\n", originalTotal);
    if (superOffer) {
        printf("Super Offer Applied: 20%% Discount = %.2f\n", finalDiscount);
    } else if (itemDiscount > 0) {
        printf("Per-Item Discount Applied = %.2f\n", finalDiscount);
    } else {
        printf("No Discount Applied.\n");
    }
    printf("Final Payable Amount: %.2f\n", finalTotal);

   // Payment options

  int choice;
  printf("\nSelect Payment Method:\n");
  printf("1. Bkash\n");
  printf("2. Nagad\n");
  printf("3. Rocket\n");
  printf("4. Bank Card\n");
  printf("5. Google Pay\n");
  printf("Enter your choice: ");
  scanf("%d", &choice);

  printf("\nYou selected: ");
  switch (choice) {
      case 1: printf("Bkash\n"); break;
      case 2: printf("Nagad\n"); break;
      case 3: printf("Rocket\n"); break;
      case 4: printf("Bank Card\n"); break;
      case 5: printf("Google Pay\n"); break;
      default: printf("Invalid Method (Payment Failed)\n"); return;
```

PAYMENT & CHECKOUT

```
=== Checkout Summary ===
🛒 Cart contents:
ID  Name     Qty  Unit($)  LineTotal($)
----------------------------------------
 1  Milk      2    2.00      4.00
 2  Bread     9    1.00      9.00
 3  Eggs      1    3.00      3.00
 4  Apple     1    5.00      5.00
 5  Coconut   1    2.00      2.00

Select Payment Method:
1. Bkash
2. Nagad
3. Rocket
4. Bank Card
5. Google Pay
Enter your choice: 1
Thank you for shopping with us!
```

SUPERSHOP
TAGLINE HERE

THANK YOU EVERYONE