CCP REPORT

(CT-175)

Mini Shopping Cart in C

Submitted By:

1. Alishba Habib (153)
2. Daniya Ali (159)
3. Amber Tariq (163)

Course Instructor:

Muhammad Abdullah

CONTENTS

1. Project Title: Mini Shopping Cart

2. Abstract:

The Mini Shopping Cart project is a console-based application developed in the C programming language that simulates the basic operations of a real-world shopping system. The program allows users to browse through multiple product categories, select items, specify quantities, and automatically calculate totals and discounts. It demonstrates the application of core programming concepts such as loops, conditional statements, switch-case structures, and user-defined functions to design an interactive and functional system. The project emphasizes user experience through features like category-wise selection, error handling for invalid inputs, and a structured billing system. Testing across various input cases confirmed the program's reliability and accuracy in computing totals and applying appropriate discounts. Overall, the Mini Shopping Cart serves as an effective example of how procedural programming in C can be used to create simple yet practical applications for retail and inventory systems.

3. Objectives:

The objective of this project is to develop a console-based Mini Shopping Cart system in C language. It allows users to select categories, choose items, specify quantities, and calculate totals automatically with discounts applied based on the purchase amount.

4. Tools and Technologies:

• Programming Language: C

• Compiler: GCC (Dev-C++)

• Platform: Windows

5. Project Description:

This Mini Shopping Cart program is a text-based system created in the C programming language. It mimics the behavior of a real shopping cart where users can select different product categories such as Dairy, Meat, and Fruits, choose items, and enter quantities. The program automatically calculates the subtotal, applies a discount according to the total, and displays the final bill. This project demonstrates practical usage of loops, conditions, switch cases, and user-defined functions in C.

6. Features:

• Category-wise item selection

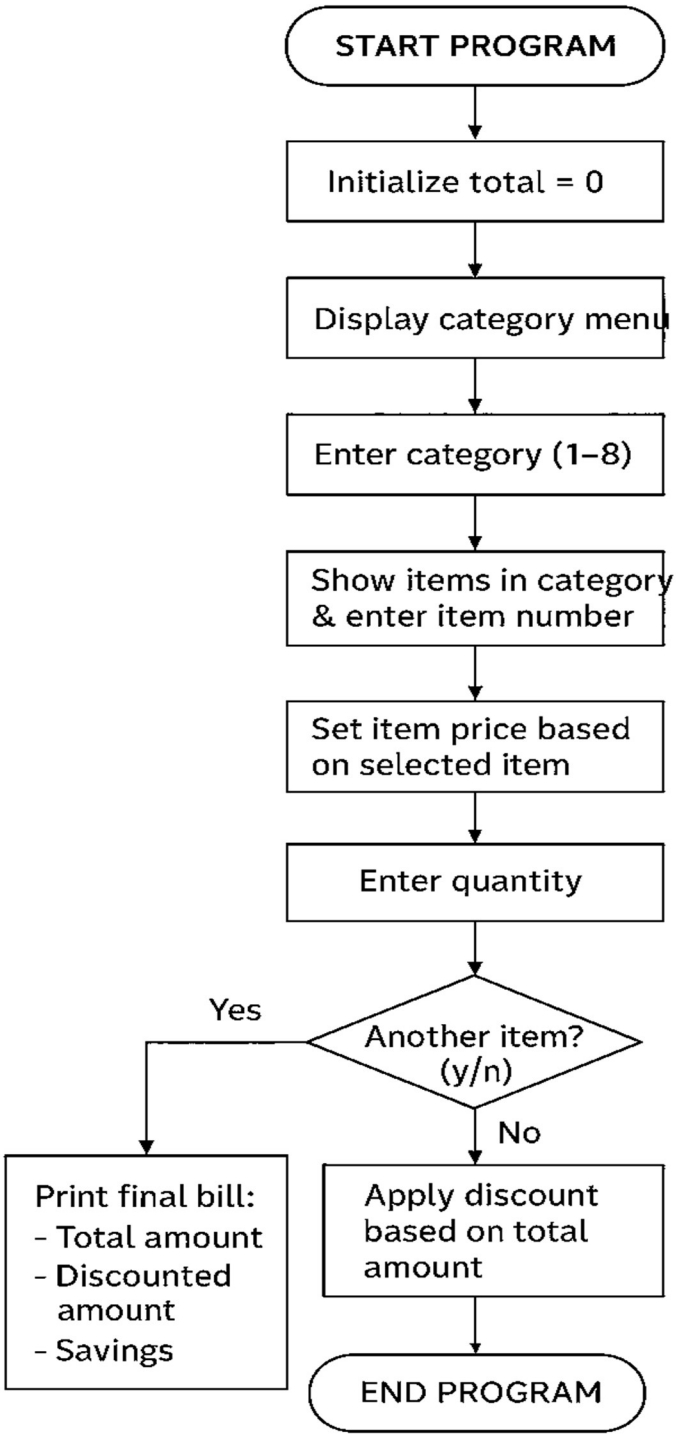
• Dynamic total and subtotal calculation

• Discount system based on total purchase

• Final bill with total, discount, and savings

• Error handling for invalid selections

7. Flowchart:


START PROGRAM
Initialize total = 0
Display category menu
Enter category (1–8)
Show items in category & enter item number
Set item price based on selected item
Enter quantity
Another item? (y/n)
Yes
No
Print final bill:
- Total amount
- Discounted amount
- Savings
Apply discount based on total amount
END PROGRAM

8. Source Code:

Header files:

- <stdio.h>
- <string.h>

Functions:

- float discount_calc(float total);
- void bill(float total, float discountedTotal);
- void showCategories();

Variables:

- int category, item, quantity;
- float price = 0, itemTotal, total = 0, discountedTotal;
- char choice;

Conditional Statements:

- Switch statement
- If statement
- Nested if else statement

Iterative statement:

- Do while loop

Jump Statements:

- Return statement
- Continue statement
- Break statement

Function Body:

```
// Function to display the categories
void showCategories()
{
   printf("\n================== CATEGORIES ==================");
   printf("\n1. Dairy");
   printf("\n2. Meat");
   printf("\n3. Cooking Stuff");
   printf("\n4. Ready Made");
   printf("\n5. Laundry");
   printf("\n6. Fruits & Vegetables");
   printf("\n7. Canned Stuff");
   printf("\n8. Bath");
   printf("\n================================================\n");
}



// Function to apply discount
float discount_calc(float total)
{
   float discount = 0;
```

```
    if (total >= 10000)
        discount = 0.40 * total;
    else if (total >= 5000)
        discount = 0.25 * total;
    else if (total >= 3000)
        discount = 0.15 * total;

    return total - discount;
}


// Function to print the final bill
void bill(float total, float discountedTotal)
{
    printf("\n=========================================\n");
    printf("--------------- FINAL BILL ---------------\n");
    printf("=========================================\n");
    printf("Total Amount:        Rs. %.2f\n", total);
    printf("Discounted Amount:   Rs. %.2f\n", discountedTotal);
    printf("You Saved:           Rs. %.2f\n", total - discountedTotal);
    printf("-----------------------------------------\n");
    printf("    Thank you for shopping with us!\n");
    printf("-----------------------------------------\n");
}
```

Switch statement:

```
switch (category)
  {
      case 1: // Dairy section
         printf("\n1. Milk (Rs. 180 per litre)");
         printf("\n2. Cheese (Rs. 500 per pack)");
         printf("\n3. Butter (Rs. 350 per pack)");
         printf("\n4. Yoghurt (Rs. 250 per kg)");
         printf("\nChoose item: ");
         scanf("%d", &item);
         if (item == 1)
            price = 180;
         else if (item == 2)
            price = 500;
         else if (item == 3)
            price = 350;
         else if (item == 4)
            price = 250;
         else{
           printf("\nInvalid item selected! Please try again.\n");
           continue;}
         break;

      case 2: // Meat section
         printf("\n1. Chicken (Rs. 650 per kg)");
```

```
    printf("\n2. Beef (Rs. 900 per kg)");
    printf("\n3. Mutton (Rs. 1300 per kg)");
    printf("\n4. Fish (Rs. 2000 per kg)");
    printf("\nChoose item: ");
    scanf("%d", &item);
    if (item == 1)
       price = 650;
    else if (item == 2)
       price = 900;
    else if (item == 3)
       price = 1300;
    else if (item == 4)
       price = 2000;
    else{
       printf("\nInvalid item selected! Please try again.\n");
       continue;}
    break;

case 3: // Cooking stuff
    printf("\n1. Oil (Rs. 650 per litre)");
    printf("\n2. Flour (Rs. 180 per kg)");
    printf("\n3. Rice (Rs. 300 per kg)");
    printf("\n4. Salt (Rs.50 per pack)");
    printf("\n5. Sugar (Rs.100 per kg)");
    printf("\n6. Beans (Rs.200 per kg)");
    printf("\n7. Tea leaves (Rs.265 per pack)");
```

```
    printf("\nChoose item: ");
    scanf("%d", &item);
    if (item == 1)
       price = 650;
    else if (item == 2)
       price = 180;
    else if (item == 3)
       price = 300;
    else if (item == 4)
       price = 50;
    else if (item == 5)
       price = 100;
    else if (item == 6)
       price = 200;
    else if (item == 7)
       price = 265;
    else{
        printf("\nInvalid item selected! Please try again.\n");
         continue;}

    break;

case 4: // Ready made
    printf("\n1. Bread (Rs. 180 per pack)");
    printf("\n2. Biscuits (Rs. 120 per pack)");
    printf("\n3. Cake (Rs. 500 per cake)");
```

```
    printf("\nChoose item: ");
    scanf("%d", &item);
    if (item == 1)
       price = 180;
    else if (item == 2)
       price = 120;
    else if (item == 3)
       price = 500;
    else{
        printf("\nInvalid item selected! Please try again.\n");
        continue;}

    break;

case 5: // Laundry
    printf("\n1. Surf Excel (Rs. 550 per kg)");
    printf("\n2. Ariel (Rs. 600 per kg)");
    printf("\n3. Comfort (Rs. 250 per bottle)");
    printf("\nChoose item: ");
    scanf("%d", &item);
    if (item == 1)
       price = 550;
    else if (item == 2)
       price = 600;
    else if (item == 3)
       price = 250;
```

```
    else{
      printf("\nInvalid item selected! Please try again.\n");
      continue;}

    break;

case 6: // Fruits and Veggies
    printf("\n1. Apples (Rs. 300 per kg)");
    printf("\n2. Tomatoes (Rs. 150 per kg)");
    printf("\n3. Potatoes (Rs. 120 per kg)");
    printf("\nChoose item: ");
    scanf("%d", &item);
    if (item == 1)
       price = 300;
    else if (item == 2)
       price = 150;
    else if (item == 3)
       price = 120;
    else{
        printf("\nInvalid item selected! Please try again.\n");
      continue;}

    break;

case 7: // Canned stuff
    printf("\n1. Baked Beans (Rs. 250 per can)");
```

```
    printf("\n2. Tuna (Rs. 350 per can)");
    printf("\n3. Sweet Corn (Rs. 200 per can)");
    printf("\nChoose item: ");
    scanf("%d", &item);
    if (item == 1)
       price = 250;
    else if (item == 2)
       price = 350;
    else if (item == 3)
       price = 200;
    else{
         printf("\nInvalid item selected! Please try again.\n");
         continue;}

    break;

case 8: // Bath
    printf("\n1. Soap (Rs. 180 per bar)");
    printf("\n2. Shampoo (Rs. 400 per bottle)");
    printf("\n3. Toothpaste (Rs. 250 per pack)");
    printf("\nChoose item: ");
    scanf("%d", &item);
    if (item == 1)
       price = 180;
    else if (item == 2)
       price = 400;
```

```
    else if (item == 3)
        price = 250;
    else{
        printf("\nInvalid item selected! Please try again.\n");
        continue;}
        break;
}
```

9. Testing and Results:

We tested the program under various input cases including:
• Normal purchase flow across multiple categories
• Invalid inputs and error handling
• Boundary cases for discount thresholds at Rs. 3000, 5000, and 10000

All tests passed successfully, and the output matched the expected results.

10. Future Enhancements:

• Add file handling to save and load shopping sessions
• Introduce arrays or structures to handle item data efficiently
• Add search, remove, or edit options in the cart
• Improve interface and implement advanced error handling

11. Conclusion:

Through this Mini Shopping Cart project, we learned how to apply fundamental C programming concepts to solve real-world problems. We gained hands-on experience in using loops, conditional statements, switch cases, and functions to design an interactive and user-friendly program. This project enhanced our problem-solving and logical thinking skills.