SRM, Andhra Pradesh

Introduction to Programming Using C
CSE 105 L
Project Report on

"Customer Billing System"

submitted in partial fulfillment for the award of the degree in

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Shivansh Goel (AP21110011078)
Venkatesh Jasthi (AP21110011077)
Ajay Jagadale (AP21110011076)
Keventh Nandala (AP21110011075)

Under the guidance of

Mrs. Vidya V
Project Mentor

ABSTRACT

"Customer Billing System" is a simple Command-Line based billing and customer database program that can help businesses manage their customer data and assist with the billing process in a very cost and time-efficient manner. It uses C Language as the front end and Generic Data Files to store the data in the backend. This program lets the businesses store customer accounts along with their basic details and their wallet balance.

CONTENTS

CHAPTER 1

INTRODUCTION

This program is intended to be used by Business Managers and the Billing Team. It works by providing the user with a menu-based interface. Users can create accounts for their customers by entering basic details which are then stored in Generic Data Files which are located in the same Directory as the Program Code. The Program also offers the user the functionality to fetch the Customer Details and update their wallet balance according to the Sale Bill or Wallet Top-Up.

The whole program has been written in C-Language which is a high-level and general-purpose procedural computer programming language that supports structured programming, lexical variable scope, and recursion with a static type system.

CHAPTER 2

OBJECTIVE

This program aims to solve the problem of long billing queues at the checkout counter and ease out the customer details and bill management process for all types of businesses all while increasing time efficiency and reducing human error which can result in huge losses for high-value transaction businesses.

CHAPTER 3

SYSTEM REQUIREMENTS

3.1 SOFTWARE REQUIREMENTS:

1. Language Used: C (C18)

3.2 HARDWARE REQUIREMENTS:

1. Operating System: Windows 7 or Later
2. Processor: Dual-Core and above
3. RAM: 1GB or More

CHAPTER 4

SYSTEM DESIGN

ALGORITHM

MENU: This contains all the options that can be carried out by the program
OPTION: Option Number according to the menu
NAME: Name of the Customer entered by the user
BALANCE: Money Available in Customer's Account

Step 1: START
Step 2: Print MENU
Step 3: Input OPTION
Step 4: IF OPTION==1:
    Input Customer Details
    Dump customer details in NAME.dat file
    Print "Details of NAME Saved Successfully"
    GOTO Step 2
Step 5: IF OPTION==2:
    Input Customer Name to be Searched
    Search for the NAME.dat file
    Output Customer Details
    GOTO Step 2
Step 6: IF OPTION==3:
    Input Customer Name and Bill Details
    Search and Update for the NAME.dat file
    Print "Updated Customer Balance: Rs. BALANCE"
    GOTO Step 2
Step 7: IF OPTION==4:
    Input Customer Name and Amount to be added
    Search and Update the NAME.dat file
    Print "Updated Customer Balance: Rs. BALANCE"
    GOTO Step 2

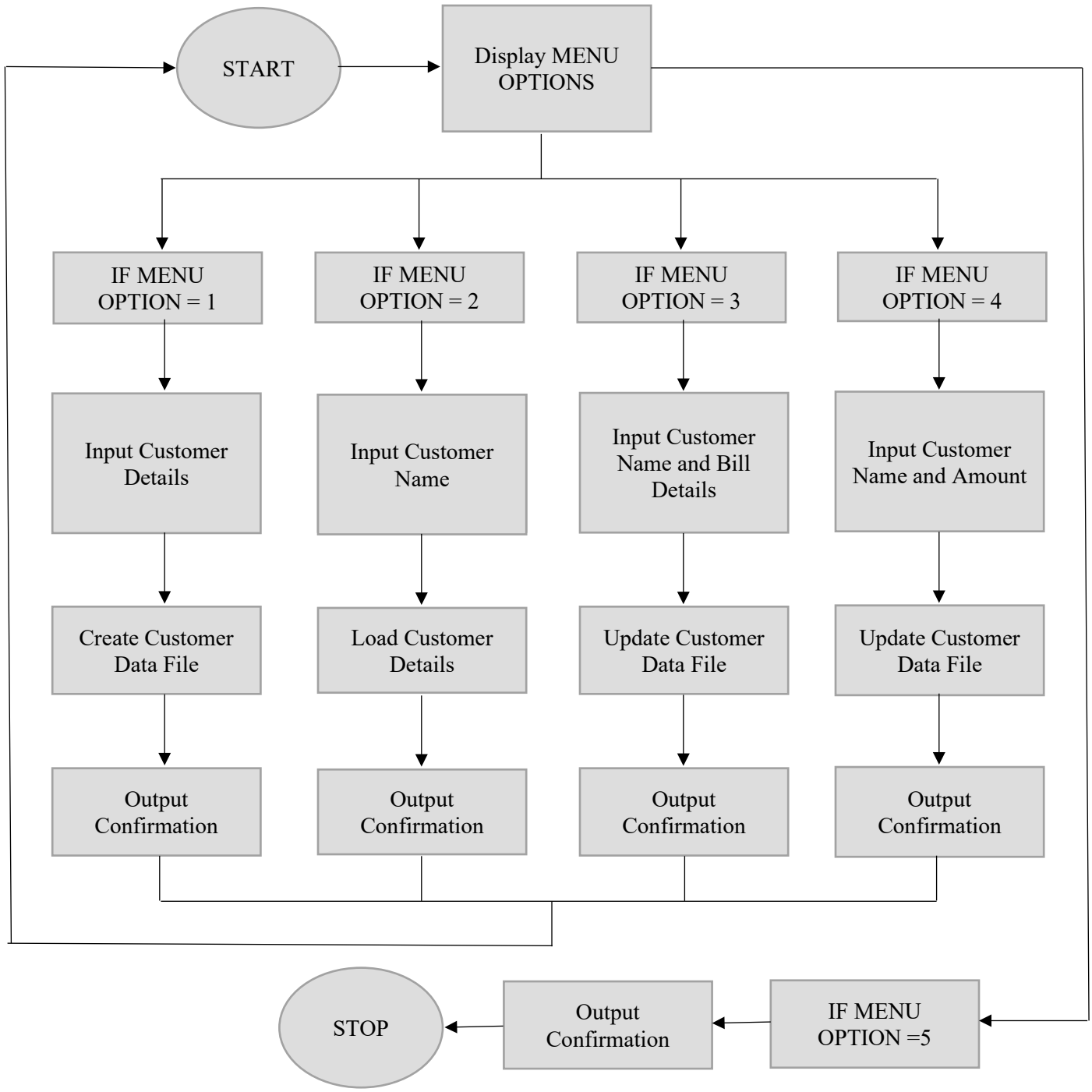
Step 8: IF OPTION==5:

Print “Program Terminated”

STOP

BLOCK DIAGRAM

Here is the representation of the program in the form of a block diagram:



CHAPTER 5

SYSTEM IMPLEMENTATION

The program is an amalgamation of 5 Modules that are accessed by the users using the Main Module:

MAIN MODULE

Main Module has an infinite loop that Executes the User Command as per the value returned by the Option Module. The loop can only be exited when the Value returned by the Option Module is “5”.

OPTION MODULE

```c
int menu()
{
    int option;
    printf("\n\n=================================================================
    printf("Select an Option:\n");
    printf("1. Add a New Customer\n");
    printf("2. Fetch Customer Details\n");
    printf("3. Create a New Bill\n");
    printf("4. Add Money to Customer's Wallet\n");
    printf("5. Exit Program\n");
    printf(">> ");
    scanf("%d", &option);
    printf("\n");
    return option;
}
```

Option Module prints the available functions that can be executed by the program and then inputs a number from the user which is then returned to the function call.

WRITEFILE MODULE

```c
void writefile()
{
    FILE *fp;
    sprintf(buffer, "%s.dat", customer.name);
    fp = fopen(buffer, "wb");
    fwrite(&customer, sizeof(customer), 1, fp);
    fclose(fp);
    printf("\nDetails of %s Saved Successfully", customer.name);
}
```

Write File Module dumps the Account Structure named “Customer” in a Generic Data File in the form of “CustomerName.dat” and then outputs a confirmation message.

SEARCH MODULE

```c
void search(char name[50])
{
    FILE *fp;
    sprintf(buffer, "%s.dat", name);
    fp = fopen(buffer, "rb");
    fread(&customer, sizeof(customer), 1, fp);
    printf("\nCustomer Details:");
    printf("\nName: %s\n", customer.name);
    printf("Mobile Number: %llu\n", customer.mobile);
    printf("Age: %d\n", customer.age);
    printf("Balance: Rs. %.2f", customer.balance);
    fclose(fp);
}
```

Search Module takes Customer Name as an input and then reads the “CustomerName.dat” file after which it outputs the Customer’s Details in a Structured Format.

READFILE MODULE

```
void readfile(char name[50])
{
    FILE *fp;
    sprintf(buffer, "%s.dat", name);
    fp = fopen(buffer, "rb");
    fread(&customer, sizeof(customer), 1, fp);
    fclose(fp);
}
```

Read File Module takes Customer Name as an input and then reads the “CustomerName.dat” file after which it loads the read data into the Account Structure named “Customer”.

CHAPTER 6

RESULTS

INSTANCE 1

```
================================================================================

Select an Option:
1. Add a New Customer
2. Fetch Customer Details
3. Create a New Bill
4. Add Money to Customer's Wallet
5. Exit Program
>> 1

Enter Customer's Name: Shivansh
Enter Customer's Mobile Number: 9318335215
Enter Customer's Age: 18
Enter Customer's Opening Balance: Rs. 1000

Details of Shivansh Saved Successfully

================================================================================
```

INSTANCE 2

```
================================================================================

Select an Option:
1. Add a New Customer
2. Fetch Customer Details
3. Create a New Bill
4. Add Money to Customer's Wallet
5. Exit Program
>> 2

Enter Customer's Name: Shivansh

Customer Details:
Name: Shivansh
Mobile Number: 9318335215
Age: 18
Balance: Rs. 1000.00

================================================================================
```

INSTANCE 3

```
================================================================================

Select an Option:
1. Add a New Customer
2. Fetch Customer Details
3. Create a New Bill
4. Add Money to Customer's Wallet
5. Exit Program
>> 3

Enter Customer's Name: Shivansh
Enter Billing Amount: Rs. 2000
Enter Amount Payed by the Customer: Rs. 500

Updated Customer Balance: Rs. -500.00
Details of Shivansh Saved Successfully

================================================================================
```

INSTANCE 4

```
================================================================================

Select an Option:
1. Add a New Customer
2. Fetch Customer Details
3. Create a New Bill
4. Add Money to Customer's Wallet
5. Exit Program
>> 4

Enter Customer's Name: Shivansh
Enter Amount to be added: Rs. 1000

Rs. 1000.00 Added Successfully
Updated Customer Balance: Rs. 500.00
Details of Shivansh Saved Successfully

================================================================================
```

INSTANCE 5

```
================================================================================

Select an Option:
1. Add a New Customer
2. Fetch Customer Details
3. Create a New Bill
4. Add Money to Customer's Wallet
5. Exit Program
>> 5

Program Terminated

================================================================================
```