Project report

on

Restaurant Billing system

Submitted to:

Touhid Ahmed

Lecturer

Department of Computer

Science & Engineering

Submitted by:

1. **Redown Ahmed** (2022-1-60-159)
2. **Md. Yousuf Hozaifa** (2022-1-60-162)
3. **Mahin Hasan** (2022-1-60-180)

Date of submission: 8 May 2022

Contents

Summary

The Restaurant billing system application refers to a complete restaurant billing system that can allow you to perform a smooth transaction as well as maintain all additional activities such as order management, production system, inventory tracking, and so on. Moreover, this system is mainly a centralized control system that records all transactional history, customer information, purchase, and sale information.

Chapter - 1 Introduction to Project

1.1 Chapter Overview

This chapter provides an overview of the project by explaining the challenge that restaurants face, the main objectives that the system expects to achieve, and a brief introduction to existing solutions.

1.2 The Problem

The restaurant billing system is specially designed for the purpose of adding items and calculating the total bill in a restaurant. This system elaborates basic concept for storing and generating ordered item's detail.

- First, show the list of foods with their associated prices and availability.
- Then, take orders from the customer as much as he/she wants.
- If any food is unavailable, notify the customer.
- Finally, prepare a bill of ordered items. Add 10% service charge and 15% value added tax to the total bill.

1.3 Project Objective

The objective of this project is to build a restaurant billing system using all the skills and techniques from the field ensuring that no common development mistakes are reproduced. Project management is critical to all software engineering projects and keeping to a project plan will be of similar importance. Any business's major goal is to maximize profit by increasing efficiency and lowering overheads while maintaining customer

happiness. Many restaurants now employ a paper-based system to interact with the kitchen, which has been demonstrated to be one of the least efficient methods. Even though this strategy is used in profitable restaurants, various issues may be considered as diminishing the restaurant's efficiency:

- Miscommunication caused by handwriting.
- Inefficient restaurant-kitchen communication.
- Difficult order tracking and time management.
- Difficult stock management.
- Limited statistical output.

These issues can be avoided or solved by using a restaurant billing system, resulting in increased earnings.

1.4 **Project Proposal**

The goal of this project is to develop a restaurant billing system that combines the advantages of all existing systems while eliminating their disadvantages and adding many new features. A list of proposed features can be found in table 1.

Table 1: A table showing the proposed features of the system and the motivation behind the features.

Feature	Motivation
Menu display	Viewing of all active meals
Meal option and preference selection	Flexible meal options available for the customer

Wireless order system	Waiters are no longer required to walk to take order

1.5 **Summary of Chapters**

The rest of this report consists of the following chapters:

- **Project Design:** Several diagrammatic approaches are used to design the project.
- **Implementation:** Uses diagrams and pseudocode to discuss the Program's implementation.
- **Outcomes:** Uses screenshots to demonstrate the system.
- **Testing:** Describes how the system was put to the test.
- **Conclusion:** Project conclusion with future development ideas.

Chapter - 2 Project Design

When we begin working on the project design, we encountered a variety of issues. However, we followed some phases:

Design Objectives –

The primary goal of the design is to deliver the requirements as specified already.

Practically –

The system should be stable and can be operated by people on average.

Cost –

It is desirable to aim for a system with a minimum cost subject to the condition that it must satisfy all requirements & maintain maintenance.

Flexibility –

The system should be modifiable depending on the changing needs of the user.

Design Process –

We need to create an application that is engaging for the user and the restaurant owner. That's we need to take some step. Below the show of the step of the process:

Algorithm

1. Firstly, the program stored all the menu names as a string literal.

2. Secondly, we used a file pointer for interactivity, and in that file, we wrote the menu name exactly as it will appear in the display. If a menu item is unavailable, the owner will simply write (U/A) beside the menu name in the file.

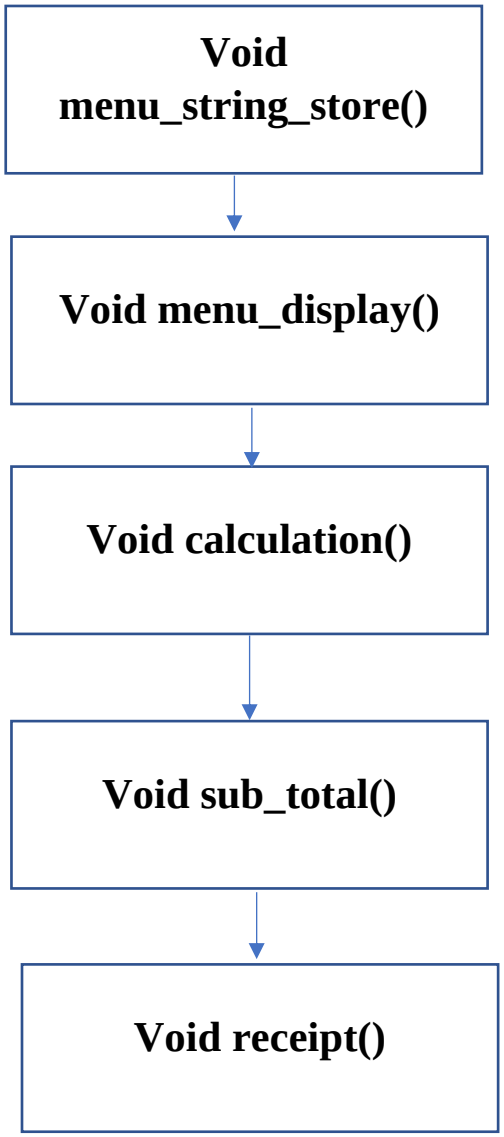
3. Then, stored this menu name in a file as a string array. We stored this because we compared string literals with string arrays. If this comparison ends in 0, then the food is available; else the food is unavailable, which will show in the display.

4. After that, A user can see what food is available and which is not. He/she can order food items based on availability as much as he/she wants. He/she just has to input the menu number and quantity.

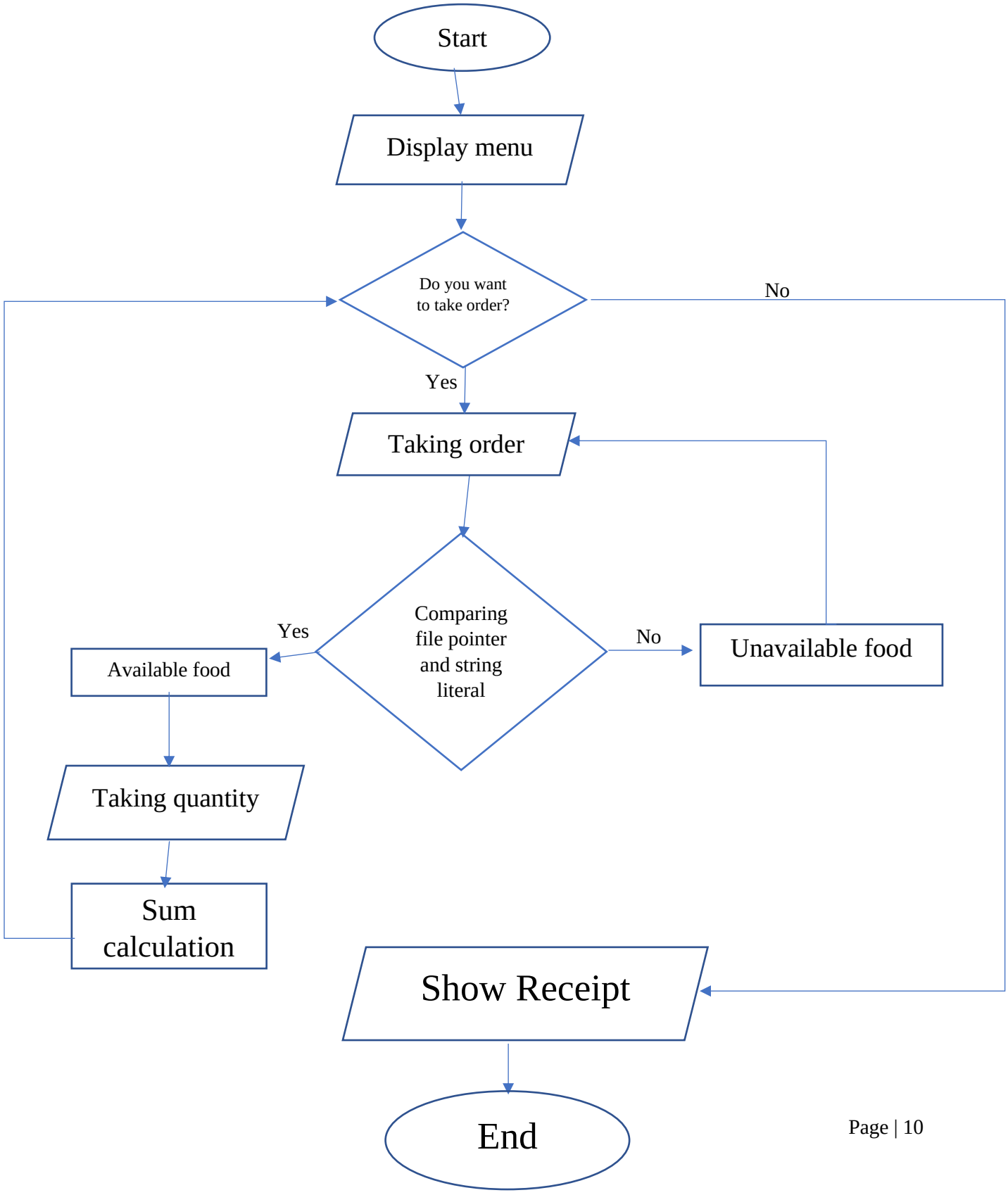
5. The receipt will show what the user ordered from the menu and the food's price. Then sub-total, vat, and the net total will be calculated.

6. Lastly, this receipt will be stored in a new file pointer as just a reminder for the owner of what the user has ordered.

User-Defined functions flow:



Flowchart


Start
Display menu
Do you want to take order?
No
Yes
Taking order
Comparing file pointer and string literal
Yes
No
Available food
Unavailable food
Taking quantity
Sum calculation
Show Receipt
End

Source Code

```c
#include <stdio.h>
#include <string.h> // used this to perform string library function
#include <ctype.h>  // used this to lowercase a character
#include <conio.h>  // used this header for getch()
#include <time.h>   // used this to generate date and time

struct menu
{
    int compare;
    char *string_from_menu;         // String literal. can't change any
element between any index. Stored here menu name.
    char string_from_database[200]; // stored here menu name from txt file.
} menu_string[50];                  // menu data type array. can store upto
50 stack of items here. But we stored 25 stack of items here.

struct calculation_part
{
    double sum, total;
    int quantity, input, limit;
    char string[200], name[100];
} sum_calculation, sum[50], receipt_string[50], loop, name; // loop.limit is
used in "for loop" limit for void calculation() function, it stored the
input limit from user
// sum[50].sum = stored all the available foods individual sum [ which
consists of (sum[50].sum * sum[50].quantity)]   sum[50].sum--->array type
// name.name[50] = stored customers name
// sum_calculation.input = takes input the number given in menu
// sum_calculation.total = all sum of sum[50].sum

FILE *ptr1, *ptr2; // declared FILE pointer globally

time_t currentTime; // special types of data_type "time_h" for time header,
declared currentTime variable

void menu_string_store(); // function prototype
```

```
void menu_display();      // for menu display
void calculation();       // for calculation
void receipt();           // generating reciept
void sub_total();         // for total sum without vat and service charge

int main()
{
    char input;
    printf("**********************************************************************
**********************************************************************
**********\n");
    printf("*\t\t\t\t\t\t\t\tWelcome to \"CHILLS
RESTAURANT\"\t\t\t\t\t\t\t\t*\n");
    printf("**********************************************************************
**********************************************************************
**********\n");
    printf("\n\n\n\t\t\t*** Press 'Y' to display the menu to progress
further and press 'N' to close the application. ***\n\n");
start:
    printf("Press (Y/N):\t");
    scanf("%c", &input);
    fflush(stdin);
    if (tolower(input) == 'y') // can input either y or Y
    {
        menu_string_store();
        menu_display();
        calculation();

        if (loop.limit == 0)
        {
            ;
        }
        else
        {
            printf("\t\t\t\t\t\t\t\t*** Receipt is given below: ***");
            receipt();
        }
        printf("\n\n\n\t\t\t\t\t\t\t\t*** Have a nice day!
***\n\n\t\t\t\t\t\t\t\t(Press any button to exit)");
    }
    else if (tolower(input) == 'n') // can input either n or N
    {
```

```
        printf("\n\n\n\t\t\t\t\t\t\t\t*** Have a nice day!
***\n\n\t\t\t\t\t\t\t\t(Press any button to exit)");
    }
    else
    {
        printf("*** Invalid keyword! Press 'Y' to display the menu to
progress further and press 'N' to close the application. ***\n\n");
        goto start; // if user press any keyword except y or n..goto will
send the code in start lebel(in 41 no line)
    }
    getch(); // it will give some time
    return 0;
}

void sub_total()
{
    for (int i = 0; i < loop.limit; i++)
    {
        sum_calculation.total += sum[i].sum; // simply summation of all
sum[50].sum....see calculation()
    }
}

void receipt()
{
    time(&currentTime); // passing address through time library function of
time.h
    int i;
    double service_charge, vat;
    printf("\n\n\n------------------------------------------------------------
------------------------------------------------------------------------
----------------------\n\n");
    printf("\t\t\t\t\t\t\t\tChills Restaurant\n");
    printf("\t\t\t\t\t\t\t\t-----------------\n");
    printf("\t\t\t\t\t\t%s", ctime(&currentTime)); // ctime library function
prints current time and date in a string with a newline.
    printf("\t\t\t\t\t\tCustomer Name: %s\n", name.name);
    printf("\t\t\t\t\t\t-------------------------------------------------------
-------\n");
    printf("\t\t\t\t\t\tItems\t\t\tQty\t\tTotal\n");
    printf("\t\t\t\t\t\t-------------------------------------------------------
-------\n\n");
    for (i = 0; i < loop.limit; i++)
```

```
    {
        if (strlen(receipt_string[i].string) <= 15)
        {
            if (sum[i].sum == 0)
                continue;
            else
                printf("\t\t\t\t\t\t%s\t\t\t%d\t\t%.2lf\n",
receipt_string[i].string, sum[i].quantity, sum[i].sum);
        }
        else if (strlen(receipt_string[i].string) > 15 &&
strlen(receipt_string[i].string) <= 22)
        {
            if (sum[i].sum == 0)
                continue;
            else
                printf("\t\t\t\t\t\t%s\t\t%d\t\t%.2lf\n",
receipt_string[i].string, sum[i].quantity, sum[i].sum);
        }

        else
        {
            if (sum[i].sum == 0)
                continue;
            else
                printf("\t\t\t\t\t\t%s\t%d\t\t%.2lf\n",
receipt_string[i].string, sum[i].quantity, sum[i].sum);
        }
    }
    printf("\n\t\t\t\t\t\t-------------------------------------------------
---------\n");

    printf("\t\t\t\t\t\tSub Total\t\t\t\t\t");
    sub_total();
    printf("%.2lf\n", sum_calculation.total);
    printf("\t\t\t\t\t\t---------\n");
    service_charge = .1 * sum_calculation.total;
    printf("\t\t\t\t\t\tService Charge (10%)\t\t\t\t%.2lf\n",
service_charge);
    vat = .15 * sum_calculation.total;
    printf("\t\t\t\t\t\tValue Added Tax (15%)\t\t\t\t%.2lf\n", vat);
    printf("\t\t\t\t\t\t-------------------------------------------------
-------\n");
```

```
    printf("\t\t\t\t\t\tGrand Total\t\t\t\t\t%.2lf\n", sum_calculation.total
+ service_charge + vat);
    printf("\t\t\t\t\t\t-------------------------------------------------
-------\n");

    // storing all receipts in file which is for the owners

    ptr2 = fopen("C:\\Users\\Rifat\\Desktop\\Project\\Owners
Database\\Receipt List.txt", "a");

    fprintf(ptr2, "\n\n\n-------------------------------------------------
-------------------------------------------------\n");
    fprintf(ptr2, "\t\t\tChills Restaurant\n");
    fprintf(ptr2, "\t\t\t-----------------\n");
    fprintf(ptr2, "%s", ctime(&currentTime)); // ctime library function
prints current time and date in a string with a newline.
    fprintf(ptr2, "Customer Name: %s\n", name.name);
    fprintf(ptr2, "-------------------------------------------------------
--------------------\n");
    fprintf(ptr2, "Items\t\t\t\tQty\t\tTotal\n");
    fprintf(ptr2, "-------------------------------------------------------
--------------------\n\n");
    for (i = 0; i < loop.limit; i++)
    {
        if (strlen(receipt_string[i].string) <= 15)
        {
            if (sum[i].sum == 0)
                continue;
            else
                fprintf(ptr2, "%s\t\t\t%d\t\t%.2lf\n",
receipt_string[i].string, sum[i].quantity, sum[i].sum);
        }
        else if (strlen(receipt_string[i].string) > 15 &&
strlen(receipt_string[i].string) <= 22)
        {
            if (sum[i].sum == 0)
                continue;
            else
                fprintf(ptr2, "%s\t\t%d\t\t%.2lf\n",
receipt_string[i].string, sum[i].quantity, sum[i].sum);
        }

        else
```

```
        {
            if (sum[i].sum == 0)
                continue;
            else
                fprintf(ptr2, "%s\t%d\t\t%.2lf\n", receipt_string[i].string,
sum[i].quantity, sum[i].sum);
        }
    }
    fprintf(ptr2, "\n-------------------------------------------------------
--------------------\n");

    fprintf(ptr2, "Sub Total\t\t\t\t\t");
    fprintf(ptr2, "%.2lf\n", sum_calculation.total);
    fprintf(ptr2, "---------\n");
    fprintf(ptr2, "Service Charge (10%)\t\t\t\t%.2lf\n", service_charge);
    fprintf(ptr2, "Value Added Tax (15%)\t\t\t\t%.2lf\n", vat);
    fprintf(ptr2, "-------------------------------------------------------
---------------------\n");
    fprintf(ptr2, "Grand Total\t\t\t\t\t%.2lf\n", sum_calculation.total +
service_charge + vat);
    fprintf(ptr2, "-------------------------------------------------------
---------------------\n");
    fclose(ptr2);
}

void calculation()
{
    char input; // local variable,takes only single character

    printf("-------------------------------------------------------------
-----------------------------------------------------------------------
---------------\n\n");
    printf("\t\t\t**** Press 'Y' to continue ordering items from the menu
and press 'N' to finish ordering from the menu. ****\n\n");
    printf("Please enter the customer name:\t");
    fflush(stdin);   // if there is any newline character comes through this
library function converts this '\n' to NULL to avoid buffer
    gets(name.name); // customer name
    printf("\n\n");
    for (loop.limit = 0; loop.limit < 50; loop.limit++)
    {
    again:
        printf("Press (Y/N):\t");
```

```
        fflush(stdin);
        scanf("%c", &input);
        printf("\n");
        if (tolower(input) == 'y')
        {
        again_menu:
            printf("Please enter the item number:\t");
            fflush(stdin);
            scanf("%d", &sum_calculation.input);
            switch (sum_calculation.input)
            {
            case 1:
                if (menu_string[0].compare == 0) // see in void
menu_display() function
                {
                quantity_again1:
                    printf("Please enter the quantity:\t");
                    fflush(stdin); // again if \n exists it terminates with
or converts with NULL
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0) // if greater than 0
then it will run normally
                    {
                        fflush(stdin);
                        sum[loop.limit].sum = 180.0 *
sum[loop.limit].quantity;                    // normal calculation with
price and stored in sum[50].sum
                        strcpy(receipt_string[loop.limit].string,
menu_string[0].string_from_menu); // copied our string menu no 1 to
receipt_string[50].string[200],see in void menu_string_store() function
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again1; // if not greeater than 0,
code will jump here from 155 no line and run its code again
                    }
                }
                else
                {
```

```
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again; // if the menu is unavailable code will jump
to line no 139
                }

            case 2:
                if (menu_string[1].compare == 0)
                {
                quantity_again2:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 200.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[1].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again2;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 3:
                if (menu_string[2].compare == 0)
                {
                quantity_again3:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
```

```
                    {
                        sum[loop.limit].sum = 225.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[2].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again3;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 4:
                if (menu_string[3].compare == 0)
                {
                quantity_again4:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 230.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[3].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again4;
                    }
```

```
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 5:
                if (menu_string[4].compare == 0)
                {
                quantity_again5:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 250.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[4].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again5;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 6:
                if (menu_string[5].compare == 0)
                {
                quantity_again6:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
```

```
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 320.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[5].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again6;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 7:
                if (menu_string[6].compare == 0)
                {
                quantity_again7:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 380.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[6].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again7;
```

```
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 8:
                if (menu_string[7].compare == 0)
                {
                quantity_again8:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 430.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[7].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again8;
                    }
                }
                else
                {
                    printf("***This item is not available. Please select
another available item from the menu.***\n\n");
                    goto again;
                }
            case 9:
                if (menu_string[8].compare == 0)
                {
                quantity_again9:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
```

```
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 680.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[8].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again9;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 10:
                if (menu_string[9].compare == 0)
                {
                quantity_again10:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 180.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[9].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
```

```
                        goto quantity_again10;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 11:
                if (menu_string[10].compare == 0)
                {
                quantity_again11:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 200.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[10].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again11;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 12:
                if (menu_string[11].compare == 0)
                {
                quantity_again12:
                    printf("Please enter the quantity:\t");
```

```
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 225.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[11].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again12;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 13:
                if (menu_string[12].compare == 0)
                {
                quantity_again13:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 230.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[12].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
```

```
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again13;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 14:
                if (menu_string[13].compare == 0)
                {
                quantity_again14:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 250.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[13].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again14;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 15:
                if (menu_string[14].compare == 0)
                {
```

```
                quantity_again15:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 320.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[14].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again15;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 16:
                if (menu_string[15].compare == 0)
                {
                quantity_again16:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 380.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[15].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
```

```
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again16;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 17:
                if (menu_string[16].compare == 0)
                {
                quantity_again17:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 430.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[16].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again17;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 18:
                if (menu_string[17].compare == 0)
```

```
                {
                quantity_again18:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 680.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[17].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again18;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 19:
                if (menu_string[18].compare == 0)
                {
                quantity_again19:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 120.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[18].string_from_menu);
                        printf("\n\n");
                        break;
                    }
```

```
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again19;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu.*** \n\n");
                    goto again;
                }
            case 20:
                if (menu_string[19].compare == 0)
                {
                quantity_again20:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 130.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[19].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again20;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 21:
```

```
                if (menu_string[20].compare == 0)
                {
                quantity_again21:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 180.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[20].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again21;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 22:
                if (menu_string[21].compare == 0)
                {
                quantity_again22:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 140.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[21].string_from_menu);
                        printf("\n\n");
                        break;
```

```
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again22;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 23:
                if (menu_string[22].compare == 0)
                {
                quantity_again23:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 140.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[22].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again23;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu.*** \n\n");
                    goto again;
                }
```

```
            case 24:
                if (menu_string[23].compare == 0)
                {
                quantity_again24:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 160.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[23].string_from_menu);
                        printf("\n\n");
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again24;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            case 25:
                if (menu_string[24].compare == 0)
                {
                quantity_again25:
                    printf("Please enter the quantity:\t");
                    fflush(stdin);
                    scanf("%d", &sum[loop.limit].quantity);
                    if (sum[loop.limit].quantity > 0)
                    {
                        sum[loop.limit].sum = 170.0 *
sum[loop.limit].quantity;
                        strcpy(receipt_string[loop.limit].string,
menu_string[24].string_from_menu);
                        printf("\n\n");
```

```
                        break;
                    }
                    else
                    {
                        printf("*** Invalid quantity! Please enter quantity
greater than 0. ***\n\n");
                        goto quantity_again25;
                    }
                }
                else
                {
                    printf("*** This item is not available. Please select
another available item from the menu. ***\n\n");
                    goto again;
                }
            default:
                printf("*** Invalid menu number! Please enter valid menu
number. ***\n\n");
                goto again_menu;
            }
        }
        else if (tolower(input) == 'n')
        {
            printf("------------------------------------------------------
------------------------------------------------------------------------
-----------------------\n\n");
            break;
        }
        else
        {
            printf("*** Invalid keyword! Please press 'Y' to continue
ordering items from the menu and press 'N' to finish ordering from the menu.
***\n\n");
            fflush(stdin);
        }
    }
}

void menu_string_store()
{ // in this function we stored serially menu items, these are string
literals
    menu_string[0].string_from_menu = "1. BEEF BURGER";
    menu_string[1].string_from_menu = "2. BEEF BURGER WITH CHEESE";
```

```
    menu_string[2].string_from_menu = "3. BEEF SMOKY BBQ CHEESE";
    menu_string[3].string_from_menu = "4. BEEF WITH BACON";
    menu_string[4].string_from_menu = "5. BEEF WITH SAUSAGE";
    menu_string[5].string_from_menu = "6. BEEF CHEESE BLAST";
    menu_string[6].string_from_menu = "7. BEEF SIGNATURE";
    menu_string[7].string_from_menu = "8. GIGANTO BEEF";
    menu_string[8].string_from_menu = "9. BINGE BEEF";
    menu_string[9].string_from_menu = "10. CHICKEN BURGER";
    menu_string[10].string_from_menu = "11. CHICKEN BURGER WITH CHEESE";
    menu_string[11].string_from_menu = "12. CHICKEN SMOKY BBQ CHEESE";
    menu_string[12].string_from_menu = "13. CHICKEN WITH BACON";
    menu_string[13].string_from_menu = "14. CHICKEN WITH SAUSAGE";
    menu_string[14].string_from_menu = "15. CHICKEN CHEESE BLAST";
    menu_string[15].string_from_menu = "16. CHICKEN SIGNATURE";
    menu_string[16].string_from_menu = "17. GIGANTO CHICKEN";
    menu_string[17].string_from_menu = "18. BINGE CHICKEN";
    menu_string[18].string_from_menu = "19. FRENCH FRIES";
    menu_string[19].string_from_menu = "20. CHICKEN FINGERS (10 PCS)";
    menu_string[20].string_from_menu = "21. NAGA DRUMS (3 PCS)";
    menu_string[21].string_from_menu = "22. COLD COFFEE";
    menu_string[22].string_from_menu = "23. MUNCH";
    menu_string[23].string_from_menu = "24. OREO";
    menu_string[24].string_from_menu = "25. NUTELLA";
}

void menu_display()
{
    int i = 0, j = 0, len;
    ptr1 = fopen("C:\\Users\\Rifat\\Desktop\\Project\\Owners Database\\Food
Availibility.txt", "r"); // file location (change it in your computer)
    char store[200];
    while (fgets(menu_string[i].string_from_database,
sizeof(menu_string[i].string_from_database), ptr1))
    { // fgets reads strings with a new line and to terminate the new line ,
used loop and replaces the last element with NULL
        len = strlen(menu_string[i].string_from_database);
        menu_string[i].string_from_database[len - 1] = 0;
        i++;
    }
    menu_string[i - 1].string_from_database[len - 1] = 'A'; // but fgets
doesnt read new line for the last line string of file, so here i assigned
the last element as A ,because 25. NUTELLA, here added A in last element
```

```
    for (j = 0; j < i; j++)
    { // compares the string of menu and string from database and stored in
menu_string[j].compare
        menu_string[j].compare = strcmp(menu_string[j].string_from_menu,
menu_string[j].string_from_database);
    }

    printf("*********************************************************************
*********************************************************************************
**************\n");
    printf("*\t\t\t\t\t\t\t\t\tCHILLS RESTAURANT\t\t\t\t\t\t\t\t  *\n");
    printf("*********************************************************************
*********************************************************************************
**************\n");
    printf("\n");

    printf("\t\tBEEF BURGERS\t\t\t");
    printf("|\t\t  CHICKEN BURGERS\t\t\t|\t\t\tSIDES\n");
    printf("\t\t------------\t\t\t|\t\t  ---------------\t\t\t|\t\t\t-----
\n");

    // for display purposes
    if (menu_string[0].compare == 0)
        printf("%s\t\t\t| 180 TK |", menu_string[0].string_from_menu);
    else
        printf("%s(U/A)\t\t| 180 TK |", menu_string[0].string_from_menu);

    printf("\t");

    if (menu_string[9].compare == 0)
        printf("|%s\t\t\t| 180 TK |", menu_string[9].string_from_menu);
    else
        printf("|%s(U/A)\t\t| 180 TK |", menu_string[9].string_from_menu);

    printf("\t");

    if (menu_string[18].compare == 0)
        printf("|%s\t\t\t| 120 TK |\n", menu_string[18].string_from_menu);
    else
        printf("|%s(U/A)\t\t\t| 120 TK |\n",
menu_string[18].string_from_menu);
```

```
    printf("   (Beef Patty, Special Sauce)\t\t\t|   (Chicken Patty, Special
Sauce)\t\t\t|\n\t\t\t\t\t\t|\t\t\t\t\t\t\t");

    if (menu_string[19].compare == 0)
        printf("|%s\t\t| 130 TK |\n", menu_string[19].string_from_menu);
    else
        printf("|%s(U/A)\t| 130 TK |\n", menu_string[19].string_from_menu);

    if (menu_string[1].compare == 0)
        printf("%s\t| 200 TK |", menu_string[1].string_from_menu);
    else
        printf("%s(U/A)\t| 200 TK |", menu_string[1].string_from_menu);

    printf("\t");

    if (menu_string[10].compare == 0)
        printf("|%s\t\t| 200 TK |", menu_string[10].string_from_menu);
    else
        printf("|%s(U/A)\t| 200 TK |", menu_string[10].string_from_menu);

    printf("\t|\n");
    printf("   (Beef Patty, Cheese, Special \t\t|   (Chicken Patty, Cheese,
Special\t\t\t");

    if (menu_string[20].compare == 0)
        printf("|%s\t\t\t| 180 TK |", menu_string[20].string_from_menu);
    else
        printf("|%s(U/A)\t\t| 180 TK |", menu_string[20].string_from_menu);

    printf("\n    Sauce)\t\t\t\t\t|    Sauce)\t\t\t\t\t\t|\n\t\t\t\t\t\t|\t\
t\t\t\t\t\t|\n");

    if (menu_string[2].compare == 0)
        printf("%s\t| 225 TK |", menu_string[2].string_from_menu);
    else
        printf("%s(U/A)\t| 225 TK |", menu_string[2].string_from_menu);

    printf("\t");

    if (menu_string[11].compare == 0)
        printf("|%s\t\t| 225 TK |", menu_string[11].string_from_menu);
    else
        printf("|%s(U/A)\t| 225 TK |", menu_string[11].string_from_menu);
```

```
    printf("\t|\t\t\tSHAKES\n");
    printf("   (BBQ Sauce Cooked Beef Patty,\t\t|   (BBQ Sauce Cooked Beef
Patty,\t\t\t|\t\t\t------\n    Cheese)\t\t\t\t\t|    Cheese)\t\t\t\t\t\t");

    if (menu_string[21].compare == 0)
        printf("|%s\t\t\t| 140 TK |", menu_string[21].string_from_menu);
    else
        printf("|%s(U/A)\t\t\t| 140 TK |",
menu_string[21].string_from_menu);

    printf("\n\t\t\t\t\t\t\t|\t\t\t\t\t\t\t|\n");

    if (menu_string[3].compare == 0)
        printf("%s\t\t| 230 TK |", menu_string[3].string_from_menu);
    else
        printf("%s(U/A)\t\t| 230 TK |", menu_string[3].string_from_menu);

    printf("\t");

    if (menu_string[3].compare == 0)
        printf("|%s\t\t\t| 230 TK |", menu_string[12].string_from_menu);
    else
        printf("|%s(U/A)\t\t| 230 TK |", menu_string[12].string_from_menu);

    printf("\t");

    if (menu_string[22].compare == 0)
        printf("|%s\t\t\t\t| 140 TK |\n", menu_string[22].string_from_menu);
    else
        printf("|%s(U/A)\t\t\t\t| 140 TK |\n",
menu_string[22].string_from_menu);

    printf("   (Beef Patty, Beef Bacon)\t\t\t|   (Chicken Patty, Beef
Bacon)\t\t\t\t|\n\t\t\t\t\t\t\t|\t\t\t\t\t\t\t");

    if (menu_string[23].compare == 0)
        printf("|%s\t\t\t\t| 160 TK |\n", menu_string[23].string_from_menu);
    else
        printf("|%s(U/A)\t\t\t\t| 160 TK |\n",
menu_string[23].string_from_menu);

    if (menu_string[4].compare == 0)
```

```
        printf("%s\t\t| 250 TK |", menu_string[4].string_from_menu);
    else
        printf("%s(U/A)\t| 250 TK |", menu_string[4].string_from_menu);

    printf("\t");

    if (menu_string[13].compare == 0)
        printf("|%s\t\t| 250 TK |", menu_string[13].string_from_menu);
    else
        printf("|%s(U/A)\t\t| 250 TK |", menu_string[13].string_from_menu);

    printf("\t|\n");
    printf("   (Beef Patty, 2x Chicken \t\t\t|   (Chicken Patty, 2x Chicken
\t\t\t\t");

    if (menu_string[24].compare == 0)
        printf("|%s\t\t\t\t| 170 TK |", menu_string[24].string_from_menu);
    else
        printf("|%s(U/A)\t\t\t| 170 TK |",
menu_string[24].string_from_menu);

    printf("\n    Sausage, Cheese)\t\t\t\t|    Sausage,
Cheese)\n\t\t\t\t\t\t|\n");

    if (menu_string[5].compare == 0)
        printf("%s\t\t| 320 TK |", menu_string[5].string_from_menu);
    else
        printf("%s(U/A)\t| 320 TK |", menu_string[5].string_from_menu);

    printf("\t");

    if (menu_string[14].compare == 0)
        printf("|%s\t\t| 320 TK |\n", menu_string[14].string_from_menu);
    else
        printf("|%s(U/A)\t\t| 320 TK |\n",
menu_string[14].string_from_menu);

    printf("   (2x Melted Cheese inside a \t\t\t|   (2x Melted Cheese inside
a \n    Double Sized Beef Patty & \t\t\t|    Double Sized Chicken Patty &
\n    Cheese outside)\t\t\t\t|    Cheese outside)\n\t\t\t\t\t\t|\n");

    if (menu_string[6].compare == 0)
        printf("%s\t\t| 380 TK |", menu_string[6].string_from_menu);
```

```
    else
        printf("%s(U/A)\t\t| 380 TK |", menu_string[6].string_from_menu);

    printf("\t");

    if (menu_string[15].compare == 0)
        printf("|%s\t\t\t| 380 TK |\n", menu_string[15].string_from_menu);
    else
        printf("|%s(U/A)\t\t| 380 TK |\n",
menu_string[15].string_from_menu);

    printf("   (2x Beef Patty, 2x Cheese, \t\t\t|   (2x Chicken Patty, 2x
Cheese, \n    Beef Pastrami, Poached Egg)\t\t\t|    Chicken Pastrami,
Poached Egg)\n\t\t\t\t\t\t|\n");

    if (menu_string[7].compare == 0)
        printf("%s\t\t\t| 430 TK |", menu_string[7].string_from_menu);
    else
        printf("%s(U/A)\t\t| 430 TK |", menu_string[7].string_from_menu);

    printf("\t");

    if (menu_string[16].compare == 0)
        printf("|%s\t\t\t| 430 TK |\n", menu_string[16].string_from_menu);
    else
        printf("|%s(U/A)\t\t| 430 TK |\n",
menu_string[16].string_from_menu);

    printf("   (2x Beef Patty, 2x Cheese, \t\t\t|   (2x Chicken Patty, 2x
Cheese, \n    Double Beef Bacon,BBQ Sauce)\t\t|    Double Beef Bacon,BBQ
Sauce)\n\t\t\t\t\t\t|\n");

    if (menu_string[8].compare == 0)
        printf("%s\t\t\t| 680 TK |", menu_string[8].string_from_menu);
    else
        printf("%s(U/A)\t\t| 680 TK |", menu_string[8].string_from_menu);

    printf("\t");

    if (menu_string[17].compare == 0)
        printf("|%s\t\t\t| 680 TK |", menu_string[17].string_from_menu);
    else
```

```
        printf("|%s(U/A)\t\t\t| 680 TK |",
menu_string[17].string_from_menu);

    printf("\t|* If any food item is unavailable, it will show up\n");

    printf("   (2x Giant Beef Patty, Smoked \t\t|   (2x Giant Chicken Patty,
Smoked\t\t\t|\tnext to the food item (U/A)\n    Chicken, Chicken Ham, 3x
\t\t\t|    Chicken, Chicken Ham, 3x
\t\t\t\t|\n    Cheese)\t\t\t\t\t|    Cheese)\t\t\t\t\t\t|* Service
Charge(10%) and VAT(15%) excluded\n\n");
    fclose(ptr1);
}
```

Sample Output

1. If all the foods are available:

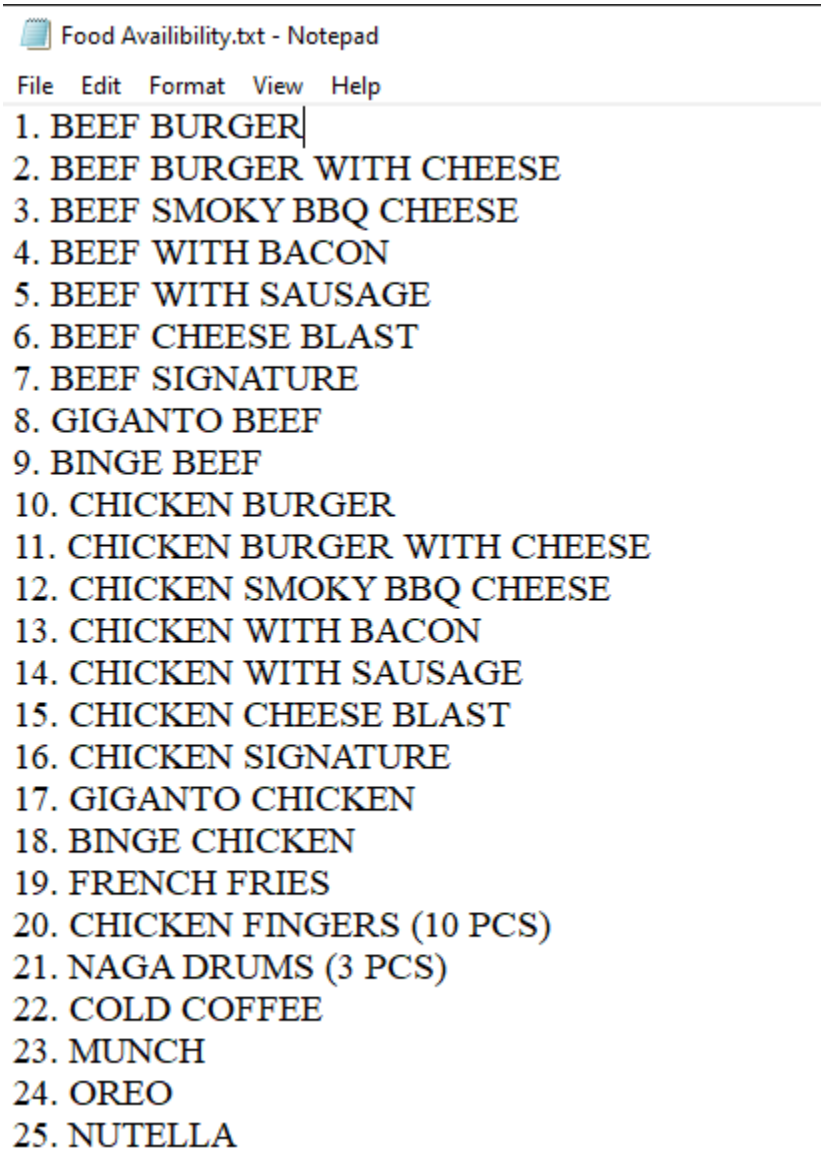
```
**********************************************************************************************************************************************************
*                                                    Welcome to "CHILLS RESTAURANT"                                                                      *
**********************************************************************************************************************************************************



                    *** Press 'Y' to display the menu to progress further and press 'N' to close the application. ***

Press (Y/N):    Y
**********************************************************************************************************************************************************
*                                                          CHILLS RESTAURANT                                                                             *
**********************************************************************************************************************************************************

              BEEF BURGERS                                |               CHICKEN BURGERS                                |                    SIDES
              ------------                                |               ---------------                                |                    -----
1. BEEF BURGER                    | 180 TK |            |10. CHICKEN BURGER                 | 180 TK |                |
   (Beef Patty, Special Sauce)                           |   (Chicken Patty, Special Sauce)                             |
                                                         |                                                              |19. FRENCH FRIES                | 120 TK |
2. BEEF BURGER WITH CHEESE        | 200 TK |            |11. CHICKEN BURGER WITH CHEESE     | 200 TK |                |
   (Beef Patty, Cheese, Special                          |   (Chicken Patty, Cheese, Special                            |20. CHICKEN FINGERS (10 PCS)    | 130 TK |
    Sauce)                                               |    Sauce)                                                    |21. NAGA DRUMS (3 PCS)          | 180 TK |
                                                         |                                                              |
3. BEEF SMOKY BBQ CHEESE          | 225 TK |            |12. CHICKEN SMOKY BBQ CHEESE       | 225 TK |                |                    SHAKES
   (BBQ Sauce Cooked Beef Patty,                         |   (BBQ Sauce Cooked Beef Patty,                              |                    ------
    Cheese)                                              |    Cheese)                                                   |22. COLD COFFEE                 | 140 TK |
                                                         |                                                              |
4. BEEF WITH BACON                | 230 TK |            |13. CHICKEN WITH BACON             | 230 TK |                |23. MUNCH                       | 140 TK |
   (Beef Patty, Beef Bacon)                              |   (Chicken Patty, Beef Bacon)                                |
                                                         |                                                              |24. OREO                        | 160 TK |
5. BEEF WITH SAUSAGE              | 250 TK |            |14. CHICKEN WITH SAUSAGE           | 250 TK |                |
   (Beef Patty, 2x Chicken                               |   (Chicken Patty, 2x Chicken                                 |25. NUTELLA                     | 170 TK |
    Sausage, Cheese)                                     |    Sausage, Cheese)                                          |
                                                         |                                                              |
6. BEEF CHEESE BLAST              | 320 TK |            |15. CHICKEN CHEESE BLAST           | 320 TK |
   (2x Melted Cheese inside a                            |   (2x Melted Cheese inside a                                 |
    Double Sized Beef Patty &                            |    Double Sized Chicken Patty &                              |
    Cheese outside)                                      |    Cheese outside)                                           |
```

```
7. BEEF SIGNATURE                | 380 TK |     |16. CHICKEN SIGNATURE                | 380 TK |
   (2x Beef Patty, 2x Cheese,                    |   (2x Chicken Patty, 2x Cheese,
    Beef Pastrami, Poached Egg)                  |    Chicken Pastrami, Poached Egg)
                                                 |
8. GIGANTO BEEF                  | 430 TK |     |17. GIGANTO CHICKEN                  | 430 TK |
   (2x Beef Patty, 2x Cheese,                    |   (2x Chicken Patty, 2x Cheese,
    Double Beef Bacon,BBQ Sauce)                 |    Double Beef Bacon,BBQ Sauce)
                                                 |
9. BINGE BEEF                    | 680 TK |     |18. BINGE CHICKEN                    | 680 TK |     |* If any food item is unavailable, it will show up
   (2x Giant Beef Patty, Smoked                  |   (2x Giant Chicken Patty, Smoked                |       next to the food item (U/A)
    Chicken, Chicken Ham, 3x                     |    Chicken, Chicken Ham, 3x                      |
    Cheese)                                      |    Cheese)                                       |* Service Charge(10%) and VAT(15%) excluded

------------------------------------------------------------------------------------------------------------------------------------------------------------

                    **** Press 'Y' to continue ordering items from the menu and press 'N' to finish ordering from the menu. ****

Please enter the customer name: Redown Ahmed


Press (Y/N):    y

Please enter the item number:   1
Please enter the quantity:      3


Press (Y/N):    a

*** Invalid keyword! Please press 'Y' to continue ordering items from the menu and press 'N' to finish ordering from the menu. ***

Press (Y/N):    Y

Please enter the item number:   29
*** Invalid menu number! Please enter valid menu number. ***
```

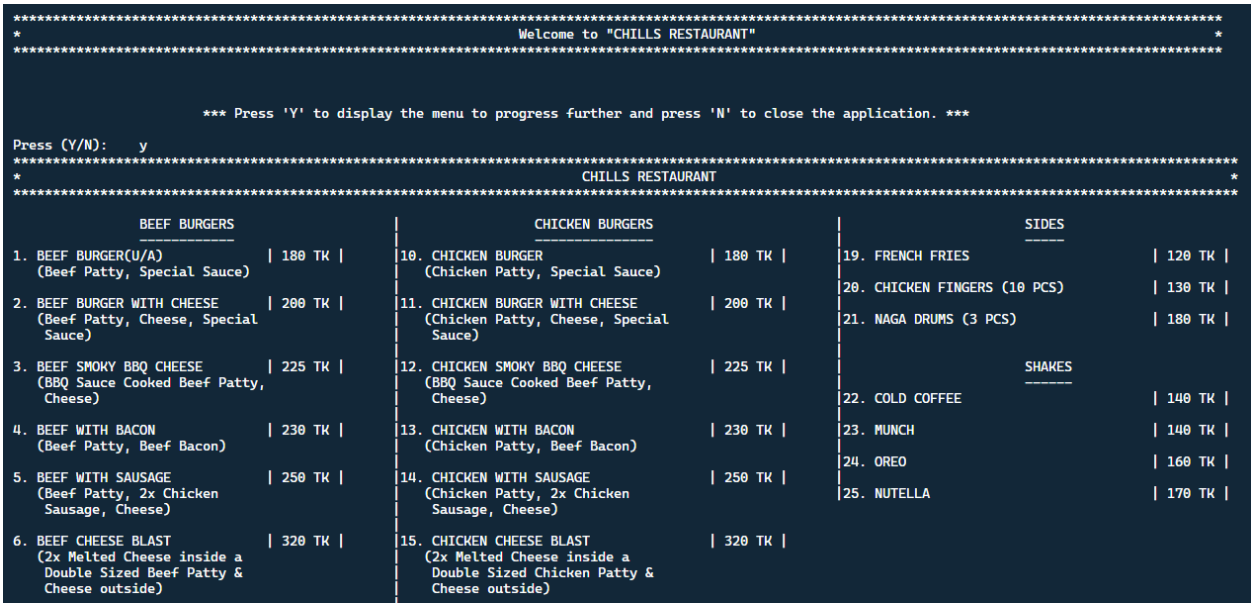
```
Please enter the item number:   25
Please enter the quantity:      0
*** Invalid quantity! Please enter quantity greater than 0. ***

Please enter the quantity:      -1
*** Invalid quantity! Please enter quantity greater than 0. ***

Please enter the quantity:      3


Press (Y/N):    n

------------------------------------------------------------------------------------------------------------------------------------------------------------

                                        *** Receipt is given below: ***

------------------------------------------------------------------------------------------------------------------------------------------------------------
```

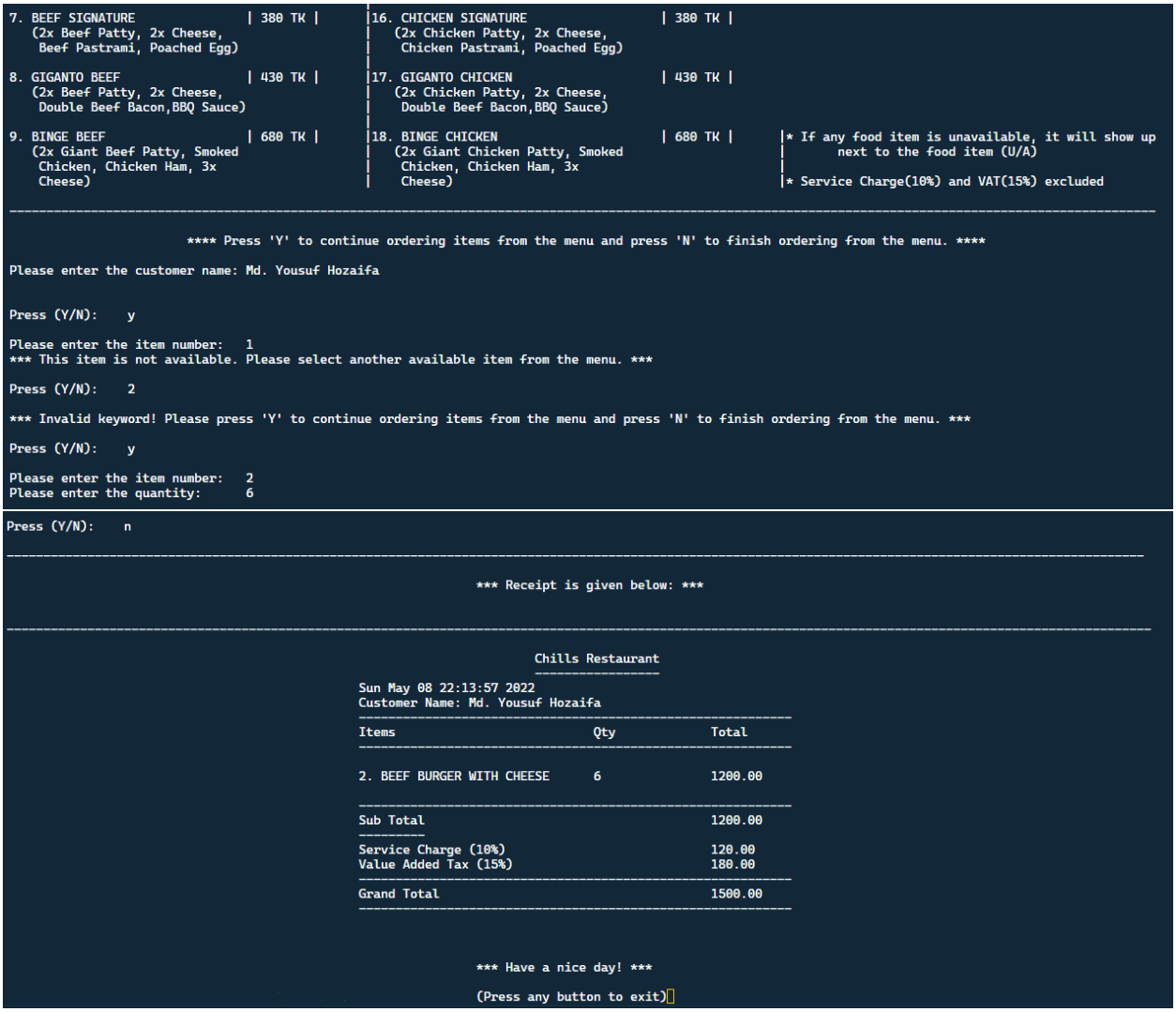
```
                                Chills Restaurant
                                -----------------
Sun May 08 22:11:28 2022
Customer Name: Redown Ahmed
-------------------------------------------------------------
Items                           Qty             Total
-------------------------------------------------------------

1. BEEF BURGER                  3               540.00
25. NUTELLA                     3               510.00

-------------------------------------------------------------
Sub Total                                       1050.00
---------
Service Charge (10%)                            105.00
Value Added Tax (15%)                           157.50
-------------------------------------------------------------
Grand Total                                     1312.50
-------------------------------------------------------------


                            *** Have a nice day! ***

                            (Press any button to exit)
```

In “Food Availibility.txt” :

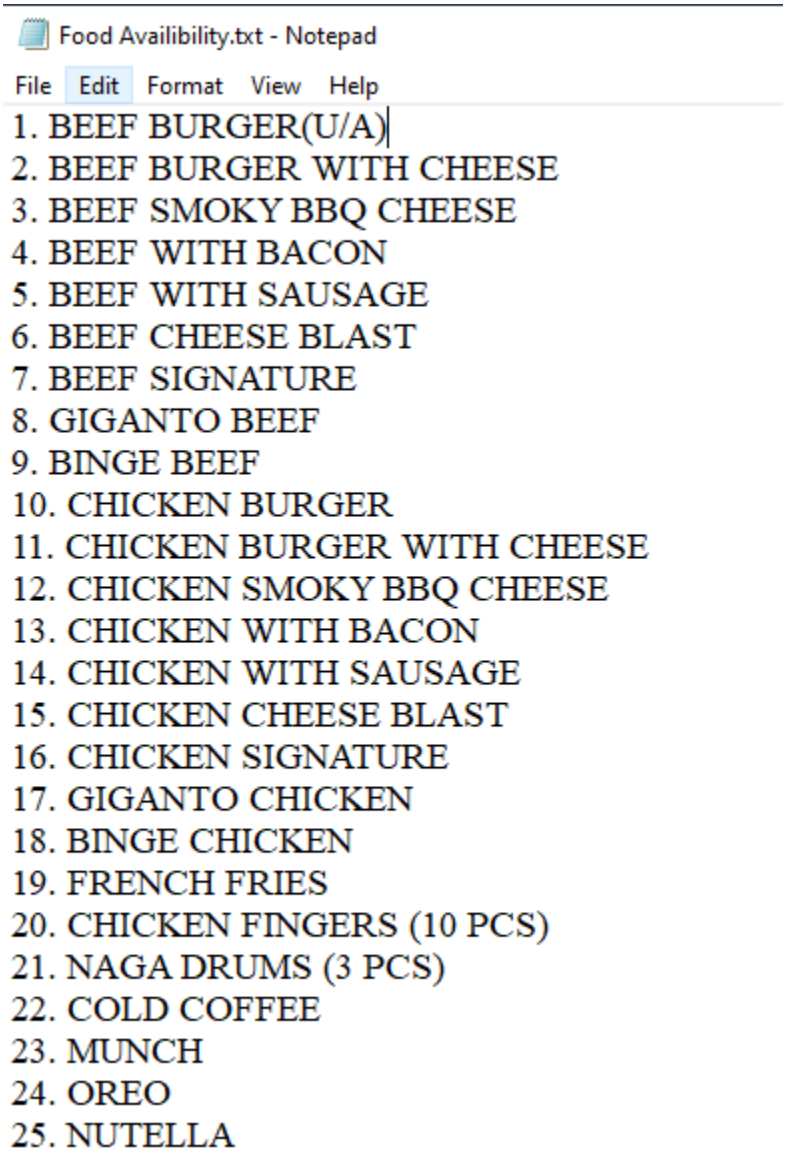


2. If foods are unavailable:




7. BEEF SIGNATURE | 380 TK |
(2x Beef Patty, 2x Cheese, Beef Pastrami, Poached Egg)
8. GIGANTO BEEF | 430 TK |
(2x Beef Patty, 2x Cheese, Double Beef Bacon,BBQ Sauce)
9. BINGE BEEF | 680 TK |
(2x Giant Beef Patty, Smoked Chicken, Chicken Ham, 3x Cheese)
16. CHICKEN SIGNATURE | 380 TK |
(2x Chicken Patty, 2x Cheese, Chicken Pastrami, Poached Egg)
17. GIGANTO CHICKEN | 430 TK |
(2x Chicken Patty, 2x Cheese, Double Beef Bacon,BBQ Sauce)
18. BINGE CHICKEN | 680 TK |
(2x Giant Chicken Patty, Smoked Chicken, Chicken Ham, 3x Cheese)
* If any food item is unavailable, it will show up next to the food item (U/A)
* Service Charge(10%) and VAT(15%) excluded
**** Press 'Y' to continue ordering items from the menu and press 'N' to finish ordering from the menu. ****
Please enter the customer name: Md. Yousuf Hozaifa
Press (Y/N): y
Please enter the item number: 1
*** This item is not available. Please select another available item from the menu. ***
Press (Y/N): 2
*** Invalid keyword! Please press 'Y' to continue ordering items from the menu and press 'N' to finish ordering from the menu. ***
Press (Y/N): y
Please enter the item number: 2
Please enter the quantity: 6
Press (Y/N): n
*** Receipt is given below: ***
Chills Restaurant
Sun May 08 22:13:57 2022
Customer Name: Md. Yousuf Hozaifa
Items Qty Total
2. BEEF BURGER WITH CHEESE 6 1200.00
Sub Total 1200.00
Service Charge (10%) 120.00
Value Added Tax (15%) 180.00
Grand Total 1500.00
*** Have a nice day! ***
(Press any button to exit)

In “Food Availibility.txt” :



3. Others:

```
**********************************************************************************************************************************************************
*                                                          Welcome to "CHILLS RESTAURANT"                                                                *
**********************************************************************************************************************************************************


                        *** Press 'Y' to display the menu to progress further and press 'N' to close the application. ***

Press (Y/N):    y
**********************************************************************************************************************************************************
*                                                                CHILLS RESTAURANT                                                                       *
**********************************************************************************************************************************************************

              BEEF BURGERS                    |                CHICKEN BURGERS                    |                        SIDES
              ------------                    |                ---------------                    |                        -----
1. BEEF BURGER(U/A)             | 180 TK |    |10. CHICKEN BURGER                 | 180 TK |    |19. FRENCH FRIES                  | 120 TK |
   (Beef Patty, Special Sauce)                 |   (Chicken Patty, Special Sauce)                 |                                                  |
                                               |                                                  |20. CHICKEN FINGERS (10 PCS)      | 130 TK |
2. BEEF BURGER WITH CHEESE      | 200 TK |    |11. CHICKEN BURGER WITH CHEESE     | 200 TK |    |
   (Beef Patty, Cheese, Special                |   (Chicken Patty, Cheese, Special                |21. NAGA DRUMS (3 PCS)            | 180 TK |
    Sauce)                                     |    Sauce)                                        |
                                               |                                                  |                        SHAKES
3. BEEF SMOKY BBQ CHEESE        | 225 TK |    |12. CHICKEN SMOKY BBQ CHEESE       | 225 TK |    |                        ------
   (BBQ Sauce Cooked Beef Patty,               |   (BBQ Sauce Cooked Beef Patty,                  |22. COLD COFFEE                   | 140 TK |
    Cheese)                                    |    Cheese)                                       |
                                               |                                                  |23. MUNCH                         | 140 TK |
4. BEEF WITH BACON              | 230 TK |    |13. CHICKEN WITH BACON             | 230 TK |    |
   (Beef Patty, Beef Bacon)                    |   (Chicken Patty, Beef Bacon)                    |24. OREO                          | 160 TK |
                                               |                                                  |
5. BEEF WITH SAUSAGE            | 250 TK |    |14. CHICKEN WITH SAUSAGE           | 250 TK |    |25. NUTELLA                       | 170 TK |
   (Beef Patty, 2x Chicken                     |   (Chicken Patty, 2x Chicken                     |
    Sausage, Cheese)                           |    Sausage, Cheese)                              |
                                               |                                                  |
6. BEEF CHEESE BLAST            | 320 TK |    |15. CHICKEN CHEESE BLAST           | 320 TK |    |
   (2x Melted Cheese inside a                  |   (2x Melted Cheese inside a                     |
    Double Sized Beef Patty &                  |    Double Sized Chicken Patty &                  |
    Cheese outside)                            |    Cheese outside)                               |
                                               |                                                  |
8. GIGANTO BEEF                 | 430 TK |    |17. GIGANTO CHICKEN                | 430 TK |    |
   (2x Beef Patty, 2x Cheese,                  |   (2x Chicken Patty, 2x Cheese,                  |
    Double Beef Bacon,BBQ Sauce)               |    Double Beef Bacon,BBQ Sauce)                  |
                                               |                                                  |
9. BINGE BEEF                   | 680 TK |    |18. BINGE CHICKEN                  | 680 TK |    |* If any food item is unavailable, it will show up
   (2x Giant Beef Patty, Smoked                |   (2x Giant Chicken Patty, Smoked                |       next to the food item (U/A)
    Chicken, Chicken Ham, 3x                   |    Chicken, Chicken Ham, 3x                      |
    Cheese)                                    |    Cheese)                                       |* Service Charge(10%) and VAT(15%) excluded

----------------------------------------------------------------------------------------------------------------------------------------------------------

                        **** Press 'Y' to continue ordering items from the menu and press 'N' to finish ordering from the menu. ****

Please enter the customer name: Mahin Hasan


Press (Y/N):    N

----------------------------------------------------------------------------------------------------------------------------------------------------------



                                                        *** Have a nice day! ***

                                                        (Press any button to exit)
```

```
**********************************************************************************************************************************************************
*                                                          Welcome to "CHILLS RESTAURANT"                                                                *
**********************************************************************************************************************************************************


                        *** Press 'Y' to display the menu to progress further and press 'N' to close the application. ***

Press (Y/N):    1
*** Invalid keyword! Press 'Y' to display the menu to progress further and press 'N' to close the application. ***

Press (Y/N):    N



                                                        *** Have a nice day! ***

                                                        (Press any button to exit)
```

Chapter - 3

Database Development

The maintenance of the records is made efficient as all the records are stored in the Access Database, through which data can be retrieved easily. The navigation control is provided in all the forms to navigate through a large number of records. If the number of records is very large, then the user must just type in the search string and get the results immediately.

As a result, the owner of the restaurant monitors all the processes through which orders are being placed.

Here we are showing our stored receipt in database:

Receipt List.txt - Notepad

File Edit Format View Help

```
--------------------------------------------------------------------------------------------------------
                        Chills Restaurant
                        -----------------
Sun May 08 22:11:28 2022
Customer Name: Redown Ahmed
-------------------------------------------------------------------------------
Items                           Qty             Total
-------------------------------------------------------------------------------

1. BEEF BURGER                  3               540.00
25. NUTELLA                     3               510.00

-------------------------------------------------------------------------
Sub Total                                       1050.00
---------
Service Charge (10%)                            105.00
Value Added Tax (15%)                           157.50
-------------------------------------------------------------------------------
Grand Total                                     1312.50
-------------------------------------------------------------------------------




--------------------------------------------------------------------------------------------------------
                        Chills Restaurant
                        -----------------
Sun May 08 22:13:57 2022
Customer Name: Md. Yousuf Hozaifa
-------------------------------------------------------------------------------
Items                           Qty             Total
-------------------------------------------------------------------------------

2. BEEF BURGER WITH CHEESE      6               1200.00

-------------------------------------------------------------------------
Sub Total                                       1200.00
---------
Service Charge (10%)                            120.00
Value Added Tax (15%)                           180.00
-------------------------------------------------------------------------------
Grand Total                                     1500.00
-------------------------------------------------------------------------------
```

Chapter – 4
Conclusion

Finally, in the restaurant billing system source code, the outcome of all the time and hard work is here. This application is mechanized to decrease human mistakes and boost productivity. The main goal of this project is to require less human effort. This system takes the necessary choices from the customer according to the various filters like price, category of the food, and popularity. Then he can place the order accordingly, and then the system calculates the total of the order with taxes and service charges, and then it can dispatch the bill that is handed over to the customer.

Hence, all the procedures are flawless and fully fulfill the requirements.