Green University of Bangladesh
Department of Computer Science and Engineering (CSE)

Faculty of Sciences and Engineering
Semester: (Fall, 2022), B.Sc. in CSE (Day)

Course Title: Data Structure Lab Course Code: 106 Section: DG

Lab Project Name: Restaurant Management Software

Student Details

Name	ID
Md. Mahim Hossain	221902082

Submission Date : 30 Dec 2022

Course Teacher's Name : Sagufta Sabah Nakshi

[For Teachers use only: Don't Write Anything inside this box]

Lab Project Status	
Marks:	**Signature:**
Comments:	**Date:**

Table of contents

Chapter 1

1.1 Introduction

The name of my project is Restaurant managements software. As you know the population is increasing rapidly as population increasing their necessity is also growing among all the necessary food is one of them .among food thing a robust system that plays in the food chain is a restaurant. That's why I make software mainly for this restaurant business. I made this software primarily to save time for people and use more minor workers. So, the computerization of the restaurant management software will not only improve the efficiency of the restaurant workers and help the customers get all things at the right price.

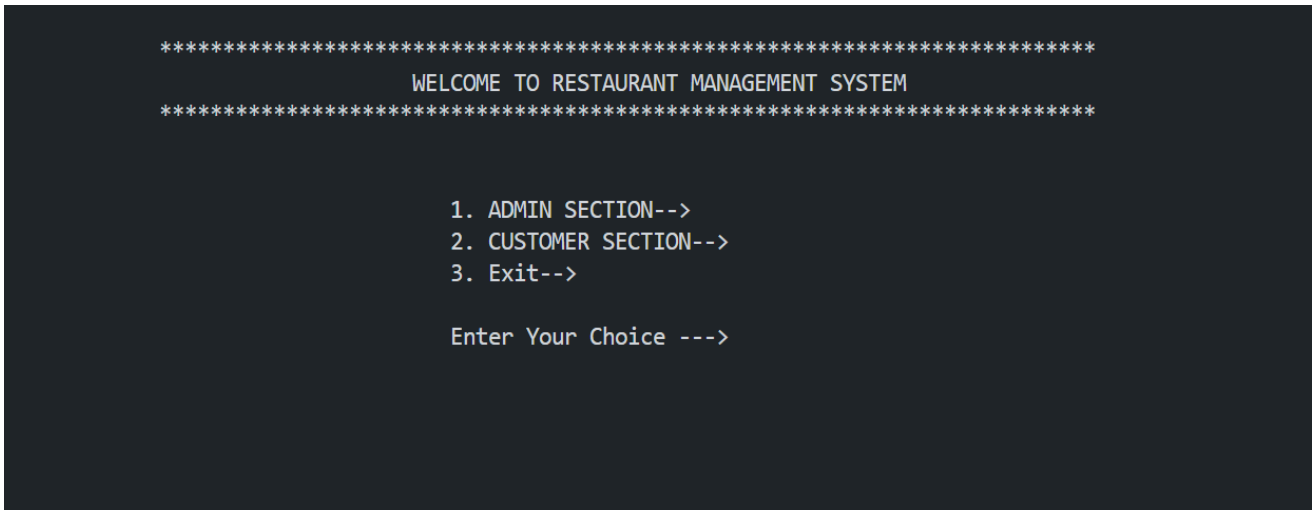
1.2 Goal of my project:

- Our main aim of the project is to get the correct information about particular things that customers Order.
- The main focus of this project is to lessen human efforts.
- To provide a user-friendly environment where users can get serviced better.

Chapter 2

2.1 restaurant management software design:

2.1.1 Interface:



2.2 Algorithm:

Step 1: Start.

Step 2: Create global struct pointers which are used throughout the program to create linked list and maintain it.

Step 3: Create admin menu function.

Step 3: Create option for admin function.

Step 4: Create function prints the options available for customer to choose.

Step 5: Creates a node for admin's linked list.

Step 6: Creates a node for customer's linked list

Step 7: Create function displays the respective entire Linked List whose head pointer is passed to it.

Step 8: Create a function for maintains the total value of sales done by maintaining another linked list which keeps a track of total sales made to each customer represented by each node.

Step 9: Create a function for performs take of calculating total sales for each customer.

Step 10: Delete an item from the order function use linked list.

Step 11: Create a function performs the task of deleting food item from admin's linked list.

Step 12: Create a function performs the task of deleting food item from customer's linked list (customer's ordered food item).

Step 13: Create a function displays the total bill of food items ordered by customer.

Step 14: Create a function perform the task of deleting entire linked list.

Step 15: Create a function opens up the admin section and provides its's interface and functionalities.

Step 16: Create a function opens up the customer section and provides its's interface and functionalities.

Step 17: Create a function prints the welcome interface and opens the main menu.

Step 18: End.

2.3 Source code:

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>

struct node
{
    char foodname[50];
    int quantity;
    float price;
    int data;
    struct node *prev;
    struct node *next;
};

struct node *headc = NULL,*newnode,*tailc = NULL;
struct node *heada = NULL, *taila = NULL;
struct node *head_s;

void adminmenu()
{
    printf("\n\t\t\t\t\t\t\t1. View total sales\n");
    printf("\t\t\t\t\t\t\t2. Add new items in the order menu\n");
    printf("\t\t\t\t\t\t\t3. Delete items from the order menu\n");
```

```
    printf("\t\t\t\t\t\t\t4. Display order menu\n");
    printf("\t\t\t\t\t\t\t5. Back To Main Menu \n\n");
    printf("\t\t\t\t\t\t\t   Enter Your Choice --->");
}

void customermenu()
{
    printf("\n\t\t\t\t\t\t\t1. Place your order\n");
    printf("\t\t\t\t\t\t\t2. View your ordered items\n");
    printf("\t\t\t\t\t\t\t3. Delete an item from order\n");
    printf("\t\t\t\t\t\t\t4. Display final bill\n");
    printf("\t\t\t\t\t\t\t5. Back To Main Menu \n\n");
    printf("\t\t\t\t\t\t\t   Enter Your Choice --->");
}

struct node* createadmin(struct node *head,int data, char foodname[25], float
price)
{
    newnode = (struct node*)malloc(sizeof(struct node));

    newnode->data = data;
    newnode->price = price;
    newnode-> quantity = 0;
    strcpy(newnode->foodname,foodname);
    newnode->next = NULL;
    newnode->prev = NULL;

    struct node *temp = head;

    if(temp==NULL)
        heada = taila = newnode;
    else
    {
        while(temp->next!=NULL)
            temp=temp->next;

        temp->next=newnode;
        newnode->prev = taila;
        taila = newnode;
    }

    return heada;
}

struct node* createcustomer(struct node *head,int data,int quantity)
{
    newnode = (struct node*)malloc(sizeof(struct node));
```

```
    struct node *temp1 = heada;
    int flag = 0;
    while(temp1!=NULL)
    {
        if(temp1->data==data)
        {
            flag = 1;
            break;
        }
        temp1 = temp1->next;
    }

    if(flag==1)
    {
        newnode->data = data;
        newnode->price = quantity*(temp1->price);
        newnode-> quantity = quantity;
        strcpy(newnode->foodname,temp1->foodname);
        newnode->next = NULL;
        newnode->prev = NULL;

        struct node *temp = head;

        if(temp==NULL)
            headc = tailc = newnode;
        else
        {
            while(temp->next!=NULL)
                temp=temp->next;

            temp->next=newnode;
            newnode->prev = tailc;
            tailc = newnode;
        }


    }
    else
    {
        printf("\n\t\t\t\t\t\t\tThis item is not present in the menu!\n");
    }
    return headc;
}

void displayList(struct node *head)
{
    struct node *temp1 = head;
    if(temp1==NULL)
```

```
    {
        printf("\n\t\t\t\t\t\t\t\tList is empty!!\n\n");
    }
    else
    {
        printf("\n");
        while(temp1!=NULL)
        {
            if(temp1->quantity==0)
                printf("\t\t\t\t\t\t%d\t%s\t%0.2f\n",temp1->data,temp1-
>foodname,temp1->price);
            else
            {
                printf("\t\t\t\t\t\t%d\t%s\t%d\t%0.2f\n",temp1->data,temp1-
>foodname,temp1->quantity,temp1->price);
            }

            temp1 = temp1->next;
        }
        printf("\n");
    }

}

struct node* totalsales(int data,int quantity)
{
    newnode = (struct node*)malloc(sizeof(struct node));
    int flag = 0;

    struct node *temp1 = heada;
    while(temp1->data!=data)
    {
        temp1 = temp1->next;
    }

    newnode->data = data;
    newnode->price = quantity*(temp1->price);
    newnode-> quantity = quantity;
    strcpy(newnode->foodname,temp1->foodname);
    newnode->next = NULL;
    newnode->prev = NULL;

    struct node *temp = head_s;

    if(temp==NULL)
        head_s = newnode;
    else
    {
```

```
        while(temp->next!=NULL)
        {
            if(temp->data==data)
            {
                flag = 1;
                break;
            }
            temp=temp->next;
        }

        if(flag==1)
        {
            temp->quantity += newnode-> quantity;
            temp->price += newnode->price;
        }
        else
        {
            temp->next=newnode;
        }
    }

    return head_s;
}

void calculatetotsales()
{
    struct node *temp = headc;
    while(temp!=NULL)
    {
        head_s = totalsales(temp->data, temp->quantity);
        temp=temp->next;
    }
}

struct node* delete(int data,struct node *head, struct node* tail)
{
    if(head==NULL)
    {
        printf("\n\t\t\t\t\t\t\tList is empty\n");
    }
    else
    {
        struct node* temp;
        if(data==head->data)
        {
            temp = head;
            head = head->next;
            if (head != NULL)
```

```
                head->prev = NULL;
            free(temp);
        }
        else if(data==tail->data)
        {
            temp = tail;
            tail = tail->prev;
            tail->next = NULL;
            free(temp);
        }
        else
        {
            temp = head;
            while(data!=temp->data)
            {
                temp = temp->next;
            }
            (temp->prev)->next = temp->next;
            (temp->next)->prev = temp->prev;
            free(temp);
        }
    }
    return head;
}

int deleteadmin()
{
    printf("\n\t\t\t\t\tEnter serial no. of the food item which is to be deleted:
");
    int num;
    scanf("%d",&num);

    struct node* temp=heada;
    while(temp!=NULL)
    {
        if (temp->data == num)
        {
            heada = delete(num, heada, taila);
            return 1;
        }
        temp=temp->next;
    }

    return 0;
}

int deletecustomer()
{
```

```
    printf("\n\t\t\t\t\tEnter serial no. of the food item which is to be deleted:
");
    int num;
    scanf("%d",&num);

    struct node* temp=headc;
    while(temp!=NULL)
    {
        if (temp->data == num)
        {
            headc = delete(num, headc, tailc);
            return 1;
        }
        temp=temp->next;
    }

    return 0;
}

void displaybill()
{
    displayList(headc);
    struct node *temp = headc;
    float total_price = 0;
    while (temp!=NULL)
    {
        total_price +=temp->price;
        temp = temp->next;
    }

    printf("\t\t\t\t\t\t\tTotal price: %0.02f\n",total_price);

}

struct node* deleteList(struct node* head)
{
    if(head==NULL)
    {
        return NULL;
    }
    else
    {
        struct node* temp = head;
        while(temp->next!=0)
        {
            temp = temp->next;
            free(temp->prev);
        }
```

```
        free(temp);
        head = NULL;
    }

    return head;
}

void admin()
{
    printf("\n\t\t\t\t\t   ----------------------------------------------\n");
    printf("\t\t\t\t\t\t\t    ADMIN SECTION\n");
    printf("\t\t\t\t\t   ----------------------------------------------\n");
    while(1)
    {
        adminmenu();

        int opt;
        scanf("%d",&opt);

        if(opt==5)
            break;

        switch (opt)
        {
            case 1:
                displayList(head_s);
                break;
            case 2:

                printf("\n\t\t\t\t\t\t\tEnter serial no. of the food item: ");
                int num,flag = 0;
                char name[50];
                float price;
                scanf("%d",&num);

                struct node *temp = heada;

                while(temp!=NULL)
                {
                    if(temp->data==num)
                    {
                        printf("\n\t\t\t\t\t\tFood item with given serial number
already exists!!\n\n");
                        flag = 1;
                        break;
                    }
                    temp = temp->next;
                }
```

```
                if(flag==1)
                    break;

                printf("\t\t\t\t\t\t\tEnter food item name: ");
                scanf("%s",name);
                printf("\t\t\t\t\t\t\tEnter price: ");
                scanf("%f",&price);
                heada = createadmin(heada, num, name, price);
                printf("\n\t\t\t\t\t\t\tNew food item added to the list!!\n\n");
                break;
            case 3:
                if(deleteadmin())
                {
                    printf("\n\t\t\t\t\t\t### Updated list of food items menu
###\n");
                    displayList(heada);
                }
                else
                    printf("\n\t\t\t\t\t\tFood item with given serial number
doesn't exist!\n\n");

                break;
            case 4:
                printf("\n\t\t\t\t\t\t   ### Order menu ###\n");
                displayList(heada);
                break;

            default:
                printf("\n\t\t\t\t\t\tWrong Input !! PLease choose valid
option\n");
                break;
        }
    }
}

void customer()
{
    int flag=0,j=1;
    char ch;
    printf("\n\t\t\t\t\t   ------------------------------------------\n");
    printf("\t\t\t\t\t\t    CUSTOMER SECTION\n");
    printf("\t\t\t\t\t   ------------------------------------------\n");
    while(1)
    {
        customermenu();

        int opt;
```

```
        scanf("%d",&opt);

        if(opt==5)
            break;

        switch (opt)
        {
            case 1:
                displayList(heada);
                printf("\n\t\t\t\t\t\tEnter number corresponding to the item you
want to order: ");
                int n;
                scanf("%d",&n);
                printf("\t\t\t\t\t\tEnter quantity: ");
                int quantity;
                scanf("%d",&quantity);
                headc = createcustomer(headc, n, quantity);
                break;
            case 2:
                printf("\n\t\t\t\t\t\t\t  ### List of ordered items ###\n");
                displayList(headc);
                break;
            case 3:
                if(deletecustomer())
                {
                    printf("\n\t\t\t\t\t\t### Updated list of your ordered food
items ###\n");
                    displayList(headc);
                }
                else
                    printf("\n\t\t\t\t\t\tFood item with given serial number
doesn't exist!!\n");
                break;
            case 4:
                calculatetotsales();
                printf("\n\t\t\t\t\t\t\t  ### Final Bill ###\n");
                displaybill();
                headc = deleteList(headc);
                printf("\n\t\t\t\t\t\tPress any key to return to main
menu:\n\t\t\t\t\t\t");
                fflush(stdin);
                ch=fgetc(stdin);
                flag=1;
                break;

            default:
                printf("\n\t\t\t\t\t\tWrong Input !! PLease choose valid
option\n");
```

```
                break;
        }
        if(flag==1)
            break;
    }
}

void mainnenu()
{
    printf("\n                                        ************************************
************************************\n");
    printf("                                                              WELCOME TO
RESTAURANT MANAGEMENT SYSTEM\n");
    printf("                                       ************************************
************************************\n\n\n");
    printf("\t\t\t\t\t\t\t1. ADMIN SECTION--> \n");
    printf("\t\t\t\t\t\t\t2. CUSTOMER SECTION--> \n");
    printf("\t\t\t\t\t\t\t3. Exit--> \n\n");
    printf("\t\t\t\t\t\t\tEnter Your Choice --->");
}

int main()
{
    heada = createadmin(heada,1,"Hot and Sour Soup",100);
    heada = createadmin(heada,2,"Manchow Soup",200);
    heada = createadmin(heada,3,"Manchurian Noodles",150);
    heada = createadmin(heada,4,"Fried Rice",180);
    heada = createadmin(heada,5,"Hakka Noodles",80);

    while(1)
    {
        mainnenu();
        int choice;
        scanf("%d",&choice);

        if(choice==3)
        {
            printf("\n\n\t\t\t\t\t\t\t**********Thank you!!**********\n");
            break;
        }

        switch (choice)
        {
            case 1:
                admin();
                break;
            case 2:
                customer();
```

```
                break;
            case 3:
                break;

            default:
                printf("\n\t\t\t\t\t\tWrong Input !! PLease choose valid
option\n");
                break;
        }
    }
}
```

Chapter3

Performance evaluation:

Front page of the project /main menu:

❖ Here you will see three options :

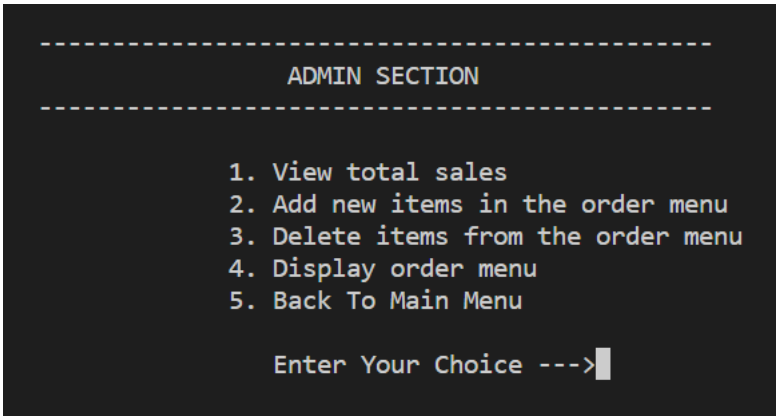
1. Admin section
2. Customer section and finally
3.Exit section.

```
**************************************************************************
                    WELCOME TO RESTAURANT MANAGEMENT SYSTEM
**************************************************************************


                    1. ADMIN SECTION-->
                    2. CUSTOMER SECTION-->
                    3. Exit-->

                    Enter Your Choice --->
```

Admin Section:

- ❖ When you select one as the admin section, you will see another page where you will give another six option:

1. View total sales.
2. Add new items in the order menu.
3. Delete items from the order menu.
4. Display order menu.
5. Back to Main Menu.



Add new items in the order menu option:

- ❖ This option is mainly for the restaurant owner or workers who will choose to add a new item to their order menu.

```
-------------------------------------------
              ADMIN SECTION
-------------------------------------------

            1. View total sales
            2. Add new items in the order menu
            3. Delete items from the order menu
            4. Display order menu
            5. Back To Main Menu

               Enter Your Choice --->2

            Enter serial no. of the food item: 6
            Enter food item name: Grill
            Enter price: 150

            New food item added to the list!!
```

Delete item in the order menu option:

- ❖ This option is also for the restaurant owner or workers who will choose to delete an item from their order menu.

```
Enter serial no. of the food item which is to be deleted: 5

        ### Updated list of food items menu ###

                1       Hot and Sour Soup       100.00
                2       Manchow Soup    200.00
                3       Manchurian Noodles      150.00
                4       Fried Rice      180.00
                6       Grill   150.00
```

Display Order menu option:

- ❖ This option will show their order menu.

```
      ### Order menu ###

   1        Hot and Sour Soup        100.00
   2        Manchow Soup    200.00
   3        Manchurian Noodles       150.00
   4        Fried Rice       180.00
   5        Hakka Noodles    80.00
   6        Grill    150.00
```

Customer section:

- ❖ This option is basically for the customer here, the customer will get 6 options such as:
  1. Place your order
  2. View your ordered items
  3. Delete an item from an order
  4. Display final bill
  5. Back to the main menu

```
-----------------------------------------------
                CUSTOMER SECTION
-----------------------------------------------

            1. Place your order
            2. View your ordered items
            3. Delete an item from order
            4. Display final bill
            5. Back To Main Menu

               Enter Your Choice --->
```

Over view output of all 6 option:

View your order items :

```
    Enter Your Choice --->1

1        Hot and Sour Soup        100.00
2        Manchow Soup    200.00
3        Manchurian Noodles       150.00
4        Fried Rice       180.00
5        Hakka Noodles    80.00
6        Grill    150.00
```

List of your order items:

```
Enter number corresponding to the item you want to order: 1
Enter quantity: 4

        1. Place your order
        2. View your ordered items
        3. Delete an item from order
        4. Display final bill
        5. Back To Main Menu

           Enter Your Choice --->2

         ### List of ordered items ###

        1       Hot and Sour Soup       4       400.00
```

Delete an item from your order:

```
Enter serial no. of the food item which is to be deleted: 1

        ### Updated list of your ordered food items ###

                        List is empty!!
```

Display final bill:

```
      ### Final Bill ###

              List is empty!!

    Total price: 0.00
```

Exit option:

When the customer is done their ordering work they can quickly get out from the system by choosing the exit option, and here is overview output of the exit function:

Chapter 4

Conclusion:

4.1 Introduction:

After completing the project, we are sure that this management software will help restaurant workers and customers order their desired food in time and without any Problem. It will reduce human errors and increase efficiency.

4.2 Practical implementation:

I already use this software in my friend's restaurant, which was not bad. The menu that I used was his restaurant menu, and he was pleased to use it.

4.3 Scope of future work:

Restaurant management software will be the most helpful software for all restaurant owners in the future. We can also use lots of new features like a food delivery system, customer record system, and we can also use it to save the record and collect information of the works .so I think this software reduces time and will serve as many ways it can.