2CSOE52 - DATA STRUCTURES
INNOVATIVE ASSIGNMENT

Prepared By :

Swapneel Bhatnagar (21BEE095)
Priyaj Gandhi (21BEE072)
Suryansh Tamra (21BEE094)
Kirti Rawal (21BEE043)

RESTAURANT MANAGEMENT SYSTEM

The project " Restaurant Management System" is implemented to reduce the manual work and enhance the accuracy of work in a restaurant . It is a type o f point - of - sale software specially designed for restaurants , bars, food trucks, and others in the food service industry .

Benefits of restaurant management system :

1) Tracks scales and orders
2) Access data easily and faster

Features:

1) Easy Handling : The restaurant management system i s designed to handle all the primary information required to calculate such as final bills, total sales during the entire day.
2) Interactive : The main purpose of the restaurant Management System is to reach a wider range of customers and to educate them about existing and new items offered by restaurants.
3) Linked List DataBase : As we will be using a linked list as a database in this restaurant management system, we will have all linked list advantages such as we can grow or shrink it any time as per our menu as it is a dynamic data structure.

CODE :

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

struct node
{
    char itemName[20];
    int quantity;
    float price;
    int sr_no;
    struct node *next;
    struct node *prev;
};

struct node *headCust = NULL;
struct node *headAdm = NULL;

void displayOrder()
{
    if (headCust == NULL)
    {
        printf("\t\t\tMenu is empty!\n");
        return;
    }
    else
    {
        printf("\t\t\tSr. No.\t\tItem Name\t\tQuantity\t\tPrice\n\n");
        struct node *temp = headCust;
        while (temp != NULL)
        {
            printf("\t\t\t%d\t\t%s\t\t%d\t\t%0.2f\n", temp->sr_no, temp->itemName, temp->quantity, (temp->price *
temp->quantity));
            temp = temp->next;
        }
    }
}

void displayMenu()
{
    if (headAdm == NULL)
    {
        printf("\t\t\tMenu is empty!\n");
        return;
    }
    else
```

```
    {
        printf("\t\t\t-------------------------------------------------------\n");
        printf("\t\t\tSr. No.\t\tItem Name\t\tPrice\n\n");
        printf("\t\t\t-------------------------------------------------------\n");
        struct node *temp = headAdm;
        while (temp != NULL)
        {
            printf("\t\t\t%d\t\t%s\t\t%0.2f\n", temp->sr_no, temp->itemName, temp->price);
            temp = temp->next;
        }
        printf("\n");
    }
}

void order()
{
    displayMenu();
    int srNo;;
    int quantity;
    printf("\t\t\tEnter the serial no. of the food item: ");
    scanf("%d", &srNo);
    printf("\t\t\tEnter the quantity: ");
    scanf("%d", &quantity);

    struct node *temp = headAdm;
    while (temp->sr_no != srNo && temp->next != NULL)
    {
        temp = temp->next;
    }
    if (temp->sr_no == srNo)
    {
        struct node *temp1 = headCust;
        while (temp1 != NULL)
        {
            if (temp1->sr_no == srNo)
            {
                temp1->quantity += quantity;
                return;
            }
            temp1 = temp1->next;
        }
        struct node *new = (struct node *)malloc(sizeof(struct node));
        new->quantity = quantity;
        new->price = temp->price;
        new->sr_no = temp->sr_no;
        strcpy(new->itemName, temp->itemName);
        if (headCust == NULL)
        {
            new->next = NULL;
```

```
            new->prev = NULL;
            headCust = new;
            printf("\t\t\tItem added!\n");
            return;
        }
        else
        {
            struct node *temp = headCust;
            while (temp->next != NULL)
            {
                temp = temp->next;
            }
            new->next = NULL;
            new->prev = temp;
            temp->next = new;
            printf("Item added!\n");
            return;
        }
    }
    else
    {
        printf("\t\t\tEnter correct serial no.!\n");
    }
}

void deleteItem()
{
    int srNo;
    printf("\t\t\tEnter the serial no. of the food item you want to delete: ");
    scanf("%d", &srNo);
    if (headCust == NULL)
    {
        printf("\t\t\tMenu is empty!\n");
        return;
    }
    if (headCust->sr_no == srNo)
    {
        struct node *temp = headCust;
        temp = headCust;
        headCust = headCust->next;
        headCust->prev = NULL;
        free(temp);
    }
    else
    {
        struct node *temp = headCust;
        while (temp->sr_no != srNo && temp->next != NULL)
        {
            temp = temp->next;
```

```
        }
        if (temp->sr_no != srNo)
        {
            printf("\t\t\tItem not present in the menu!\n");
            return;
        }
        if(temp->next == NULL)
        {
            (temp->prev)->next = NULL;
            free(temp);
        }
        else{
            (temp->prev)->next = temp->next;
            (temp->next)->prev = temp->prev;
            free(temp);
        }
    }
    printf("\t\t\tItem removed from the menu!\n");
}

void deleteList()
{
    if (headCust == NULL)
    {
        return;
    }
    else
    {
        struct node *temp = headCust;
        struct node *pred;
        while (temp->next != NULL)
        {
            pred = temp;
            temp = temp->next;
            free(pred);
        }
        free(temp);
        headCust = NULL;
    }
    return;
}

void displayBill()
{
    displayOrder();
    float result = 0;
    struct node *temp = headCust;
    while (temp != NULL)
    {
```

```
        result += (temp->quantity) * (temp->price);
        temp = temp->next;
    }
    printf("\t\t\tTotal Amount is : %0.2f\n", result);
    deleteList();
}

void addItem(int srNo, char name[], float mrp)
{
    if (headAdm == NULL)
    {
        struct node *newAdm = (struct node *)malloc(sizeof(struct node));
        strcpy(newAdm->itemName, name);
        newAdm->price = mrp;
        newAdm->sr_no = srNo;
        newAdm->quantity = 0;
        newAdm->next = NULL;
        newAdm->prev = NULL;
        headAdm = newAdm;
    }
    else
    {
        struct node *temp = headAdm;
        while (temp->next != NULL)
        {
            temp = temp->next;
        }
        struct node *newAdm = (struct node *)malloc(sizeof(struct node));
        strcpy(newAdm->itemName, name);
        newAdm->price = mrp;
        newAdm->sr_no = srNo;
        newAdm->quantity = 0;
        newAdm->prev = temp;
        temp->next = newAdm;
        newAdm->next = NULL;
    }
}

int isPresent(int srNo /*,char name[20]*/)
{
    struct node *temp = headAdm;
    while (temp != NULL)
    {
        if (temp->sr_no == srNo) //&& strcasecmp(temp->itemName, name) == 0)
        {
            return -1;
        }
        else
        {
```

```
            temp = temp->next;
        }
    }
    return 1;
}

void removeItem(int srNo)
{
    if (headAdm == NULL)
    {
        printf("\t\t\tMenu is empty!\n");
        return;
    }
    if (headAdm->sr_no == srNo)
    {
        struct node *temp = headAdm;
        temp = headAdm;
        headAdm = headAdm->next;
        headAdm->prev = NULL;
        free(temp);
    }
    else
    {
        struct node *temp = headAdm;
        while (temp->sr_no != srNo && temp->next != NULL)
        {
            temp = temp->next;
        }
        if (temp->sr_no != srNo)
        {
            printf("\t\t\tItem not present in the menu!\n");
            return;
        }
        if(temp->next == NULL)
        {
            (temp->prev)->next = NULL;
            free(temp);
        }
        else
        {
            (temp->prev)->next = temp->next;
            (temp->next)->prev = temp->prev;
            free(temp);
        }
    }
    printf("\t\t\tItem removed from the menu!\n");
}

void admin();
```

```
void customer();

void menu()
{
    int opt;
    printf("\t\t\t-------------------------------------------------------\n");
    printf("\t\t\tSelect Your Category\n");
    printf("\t\t\t1)ADMIN\n\t\t\t2)CUSTOMER\n");
    printf("\t\t\t->-> Enter 3 to EXIT: \n");
    printf("\t\t\t-------------------------------------------------------\n\t\t\t");
    scanf("%d", &opt);
    if (opt == 1)
    {
        admin();
    }
    else if (opt == 2)
    {
        customer();
    }
    else if (opt == 3)
    {
        printf("Program executed.\n");
        exit(0);
    }
    else
    {
        printf("Enter valid option.\n");
    }
}

int main()
{
    printf("\t\t\t=======================================================\n");
    printf("\t\t\t\t\tRESTAURANT MANAGEMENT\n");
    printf("\t\t\t=======================================================\n");
    addItem(1, "Munchow Soup",150);
    addItem(2, "Tomato Soup",130);
    addItem(3, "Coconut Soup",170);
    addItem(4, "Paneer Tikka",200);
    addItem(5, "Manchurian",190);
    addItem(6, "Hakka Noodles",180);
    addItem(7, "Tandoori Roti",30);
    addItem(8, "Dum Biryani",250);
    while (1)
    {
        menu();
    }
    return 0;
}
```

```
void admin()
{
   int adm;
   char food_nm[20];
   float price;
   int srNo;
   while (1)
   {
      printf("\t\t\t************************************************************\n");
      printf("\t\t\t\t\tADMIN SECTION\n");
      printf("\t\t\t************************************************************\n");
      // printf("\t\t\tEnter 1 to view the sale record: \n");
      printf("\t\t\tEnter 1 to add new item in the menu: \n");
      printf("\t\t\tEnter 2 to remove item in the menu: \n");
      printf("\t\t\tEnter 3 to display the received menu: \n");
      printf("\t\t\tEnter 4 to display the received order: \n");
      printf("\t\t\tEnter 5 to exit: \n");
      printf("\t\t\t-------------------------------------------------------\n");
      printf("\t\t\tEnter the service you want: \n\t\t\t");
      scanf("%d", &adm);

      switch (adm)
      {
      case 1:
         printf("\t\t\tEnter the serial no. of the food item: "); // assuming different serial no. has different food
item
         scanf("%d", &srNo);
         // printf("Enter the name of food item: ");
         // scanf("%s" ,food_nm);
         if (isPresent(srNo) == 1)
         {
            printf("\t\t\tEnter the name of food item: ");
            scanf("%s", food_nm);
            printf("\t\t\tEnter the price of the food item: ");
            scanf("%f", &price);
            addItem(srNo, food_nm, price);
            printf("\t\t\tItem added in the menu!\n");
         }
         else
         {
            printf("\t\t\tItem already exist!\n");
         }
         break;

      case 2:
         displayMenu();
         printf("\t\t\t-------------------------------------------------------\n");
         printf("\t\t\tEnter the serial no. of the food item you want to remove: ");
```

```
            scanf("%d", &srNo);
            removeItem(srNo);
            break;

        case 3:
            displayMenu();
            break;

        case 4:
            displayOrder();
            break;

        case 5:
            menu();

        default:
            printf("\t\t\tEnter the correct option: \n");
        }
    }
}

void customer()
{
    int cust;
    // int table_no = 1;
    while (1)
    {
        printf("\t\t\t**********************************************************\n");
        printf("\t\t\tCUSTOMER SECTION\n");
        printf("\t\t\t---------------------------------------------------------\n");
        printf("\t\t\tEnter 1 to place the order: \n");
        printf("\t\t\tEnter 2 to display the order: \n");
        printf("\t\t\tEnter 3 to delete item from the order: \n");
        printf("\t\t\tEnter 4 to display the bill: \n");
        printf("\t\t\tEnter 5 to exit: \n");
        printf("\t\t\t**********************************************************\n");
        printf("\t\t\tEnter the service you want: ");
        scanf(" %d", &cust);

        switch (cust)
        {
        case 1:
            order();
            break;

        case 2:
            displayOrder();
            break;
```

```
        case 3:
            displayOrder();
            deleteItem();
            break;

        case 4:
            displayBill();
            break;

        case 5:
            menu();

        default:
            printf("Enter the correct option: \n");
        }
    }
}
```

SAMPLE OUTPUT :

```
==========================================================
                RESTAURANT MANAGEMENT
==========================================================
----------------------------------------------------------
Select Your Category
1)ADMIN
2)CUSTOMER
->-> Enter 3 to EXIT:
----------------------------------------------------------
```

```
**********************************************************
CUSTOMER SECTION
----------------------------------------------------------
Enter 1 to place the order:
Enter 2 to display the order:
Enter 3 to delete item from the order:
Enter 4 to display the bill:
Enter 5 to exit:
**********************************************************
Enter the service you want: 1
----------------------------------------------------------
Sr. No.          Item Name                 Price

----------------------------------------------------------
1                Munchow Soup              150.00
2                Tomato Soup               130.00
3                Coconut Soup              170.00
4                Paneer Tikka              200.00
5                Manchurian                190.00
6                Hakka Noodles             180.00
7                Tandoori Roti             30.00
8                Dum Biryani               250.00

Enter the serial no. of the food item: 1
Enter the quantity: 2
Item added!
**********************************************************
CUSTOMER SECTION
----------------------------------------------------------
Enter 1 to place the order:
Enter 2 to display the order:
Enter 3 to delete item from the order:
Enter 4 to display the bill:
Enter 5 to exit:
**********************************************************
Enter the service you want: 1
----------------------------------------------------------
Sr. No.          Item Name                 Price

----------------------------------------------------------
1                Munchow Soup              150.00
2                Tomato Soup               130.00
```

```
Enter the service you want: 3
Sr. No.         Item Name               Quantity                Price

1               Munchow Soup            2               300.00
7               Tandoori Roti           2               60.00
Enter the serial no. of the food item you want to delete: 1
Item removed from the menu!
**********************************************************
CUSTOMER SECTION
----------------------------------------------------------
Enter 1 to place the order:
Enter 2 to display the order:
Enter 3 to delete item from the order:
Enter 4 to display the bill:
Enter 5 to exit:
**********************************************************
Enter the service you want: 4
Sr. No.         Item Name               Quantity                Price

7               Tandoori Roti           2               60.00
Total Amount is : 60.00
**********************************************************
```

```
**************************************************************
                    ADMIN SECTION
**************************************************************
Enter 1 to view the sale record:
Enter 2 to add new item in the menu:
Enter 3 to remove item in the menu:
Enter 4 to display the recieved menu:
Enter 5 to display the recieved order:
Enter 6 to exit:
--------------------------------------------------------------
Enter the service you want:

Enter the serial no. of the food item: 9
Enter the name of food item: Burger
Enter the price of the food item: 80
Item added in the menu!
**************************************************************
```