UIT UNIVERSITY
DEPARTMENT OF ENGINEERING TECHNOLOGY

OBJECT-ORIENTED PROGRAMMING

HST-121

Spring 2024

ASSIGNMENT # 01-02

Group Members Name and roll numbers:

1. Student Name (Roll number)
2. Student Name (Roll number)
3. Student Name (Roll number)
4. Student Name (Roll number)

Section : A

Semester : 2nd

Date of submission: 23-MAY-2024

Marks Obtained: _____

(Based on Viva & Presentation)

Submitted to Engr.RABIA BILAL

Title: Restaurant Management Program

Introduction:

Welcome to the Restaurant Management Program! This program aims to streamline the operations of a restaurant by providing a comprehensive solution for managing various tasks efficiently. Whether it's managing the menu, handling reservations, generating bills, or collecting feedback from customers, this program is designed to simplify the day-to-day operations of a restaurant.

Key Features:

1. Menu Management:

- The program allows the restaurant staff to manage the menu efficiently. This includes adding, updating, and removing items from the menu.

2. Reservation Handling:

- Customers can make reservations through the program, specifying the date, time, and number of guests. Staff can view and manage reservations to ensure smooth operations.

3. Billing System:

- The program calculates bills based on the items ordered by the customers, including taxes and additional charges. It supports multiple payment methods for customer convenience.

4. Feedback Collection:

- After their dining experience, customers can provide feedback through the program. This helps the restaurant management understand customer preferences and areas for improvement.

5. Interactive Interface:

- The program features an intuitive and user-friendly interface, making it easy for both staff and customers to navigate and use the various functionalities.

Modules:

Implementation:

The program will be implemented using object-oriented programming principles in a language such as C++, Java, or Python. It will utilize data structures and algorithms to efficiently manage restaurant data and operations. Additionally, graphical user interface (GUI) libraries may be used to create an interactive and visually appealing interface for the program.

SOURCE CODE:

```
#include <iostream>

#include <windows.h>

#include <mmsystem.h>
```

```
#include <cstdlib> // Required for rand() function
#include <ctime>   // Required for seeding rand()
#pragma comment(lib, "winmm.lib") // Link with winmm.lib to resolve PlaySound function
using namespace std;
// Class representing the welcome message
class Welcome {
public:
   Welcome() {
      cout << "Hello" << endl;
      cout << "WELCOME TO OUR RESTAURANT RANGOLI BUFFET ARENA:" << endl;
      PlaySound(TEXT("s.wav"), NULL, SND_FILENAME); // Play welcome sound
      cout << "How can I help you, sir?" << endl;
      cout << "HERE IS THE MENU OF OUR RESTAURANT" << endl;
   }
};// Class representing the dinner menu
class DinnerMenu {
public:
   void display() {
      cout << "DINNER MENU" << endl;
      cout<<"MONDAY TO SUNDAY \t\t\t\t Slot A 07:30 pm - 09:30 pm Slot B 09:30 pm - 11:30 pm";
   }
void soup() {
      cout <<endl<< "SOUP:" << endl;
      cout << "\t Soup Of The Day (Chinese)" << endl;
      cout << "\t Soup Of The Day (Continental)" << endl;
   }
   void saladbar() {
      cout << endl << "SALAD BAR:" << endl;  string dish_2[13] = {"Russian Salad", "Beetroot Salad", "Sea Food Salad", "German Potato Salad", "Macaroni Salad", "Apple Cabbage Salad", "Fresh Green Salad", "Beans Salad", "Fresh Carrot", "Fresh Onion", "Fresh Cucumber", "Fresh Tomato", "Fresh Lettuce"};
```

```
        const int num_columns = 3; // Number of columns in the table
        int rows = 13 / num_columns; // Number of full rows
        int remainder = 13 % num_columns; // Items left after forming full rows

        // Print full rows
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < num_columns; j++) {
                cout << dish_2[i * num_columns + j] << "\t\t";
            }
            cout << endl;
    }// Print remaining items in the last row
        for (int i = 0; i < remainder; i++) {
            cout << dish_2[rows * num_columns + i] << "\t\t";
        }
        cout << endl;
    }

    void dressing() {
        cout << "DRESSING:" << endl;
        string dish_3[3] = {"Mustard Sauce", "Thousand Dressing", "French Dressing"};
        for (int i = 0; i < 3; i++) {
            cout << "\t" << dish_3[i] ;
        }
    }
};Class representing the lunch menu
class LunchMenu {
public:
    void bbq() {
        cout<<endl<<"LUNCH MENU  \t\t\t Monday to Saturday\t\t 01:00 pm - 03:00 pm";
        cout << endl << "BBQ:" << endl;
        string lunch_1[2] = {"Chicken Boti", "Chandan Kabab"};
```

```
        for (int i = 0; i < 2; i++) {
            cout << "\t" << lunch_1[i];
        }
    }

    void desi() {
        cout << endl << "LIVE DESI:" << endl;
        string lunch_2[4] = {"Halwa", "Puri", "Kachori", "Tarkari"};
        for (int i = 0; i < 4; i++) {
            cout << "\t" << lunch_2[i];
        }
    }
};
```

Class representing the hi-tea menu

```
class HiTeaMenu : public DinnerMenu {
public:
    void mainCourse() {
        cout << endl << "HI-TEA Menu\t\t Monday to Sunday\t\t 04:30 pm - 06:30 pm" ;
        string tea_1p[17] = {"Finger Fish", "Fried Prawns", "Flip Rice", "Mini Pizza", "Chicken Chilli", "Nuts", "Aaloo Samoosa", "Chowmein", "Shami Kabab", "Chicken Patties", "Mutton Haleem", "Pasta", "Chicken Wonton", "Mix Pakora", "Batata Wara", "Chicken Sandwich", "French Fries"};
        for (int i = 0; i < 17; i++) {
            cout << "\t" << tea_1p[i] ;
        }
    }
};
// Class representing the brunch menu
class BrunchMenu:public DinnerMenu {
public:
    void breaddisplay() {
        cout << endl << "  BRUNCH MENU:\t\t\t Sunday\t12:00 pm - 03:00 pm" ;
```

```
        cout <<endl<< "Breaddisplay:" << endl;
        string brunch_1[8] = {"Assorted Croissant", "Danish Pastry", "Bread Slice", "Brown Bread", "Jam", "Butter", "Honey", "Muffin"};
        for (int i = 0; i < 8; i++) {
            cout << "\t" << brunch_1[i] ;
        }
    }
};
// Class representing booking
class Booking {
public:
    void people() {
        cout << endl << "ARE YOU INTERESTED IN IT" << endl;
        cout << "SELECT THE MENU WHICH YOU WANT FOR  BOOKING" << endl;
        cout << "1. DINNER MENU" << endl;
        cout << "2. LUNCH MENU" << endl;
        cout << "3. Hi Tea MENU" << endl;
        cout << "4. BRUNCH MENU" << endl;
        PlaySound(TEXT("L.wav"), NULL, SND_FILENAME);
        int n;
        cout << "ENTER THE NUMBER OF MENU YOU WANT TO SELECT: ";
        cin >> n;
        if (n == 1) {
            cout << "YOU HAVE SELECTED DINNER" << endl;
        } else if (n == 2) {
            cout << "YOU HAVE SELECTED LUNCH" << endl;
        } else if (n == 3) {
            cout << "YOU HAVE SELECTED HI TEA MENU" << endl;
        } else if (n == 4) {
            cout << "YOU HAVE SELECTED BRUNCH MENU" << endl;
        }      srand(time(0)); // Generate and print a random number between 1 and 100
```

```
        int randomNumber = rand() % 100 + 1;
        cout << "TABLE number: " << randomNumber << endl;
    }
};// Class representing a timer
class Timer {
public:
    void startTimer(int minutes) {
        cout << "Timer started for " << minutes << " minutes." << endl;
        int seconds = minutes * 60;

        while (seconds > 0) {
            cout << "Time remaining: " << seconds / 60 << " minutes " << seconds % 60 << " seconds." << endl;
            Sleep(1000); // Sleep for 1 second (1000 milliseconds)
            seconds--;
        }

        cout << "Time's up!" << endl;
    }

    void startShortTimer() {
        cout << "Short Timer started for 2 seconds." << endl;
        Sleep(2000); // Sleep for 2 seconds
        cout << "Short Timer's up!" << endl;
    }
};
// Class representing billing
class Billing {
public:
    void generateBill(int n) {
        if (n == 1) {
```

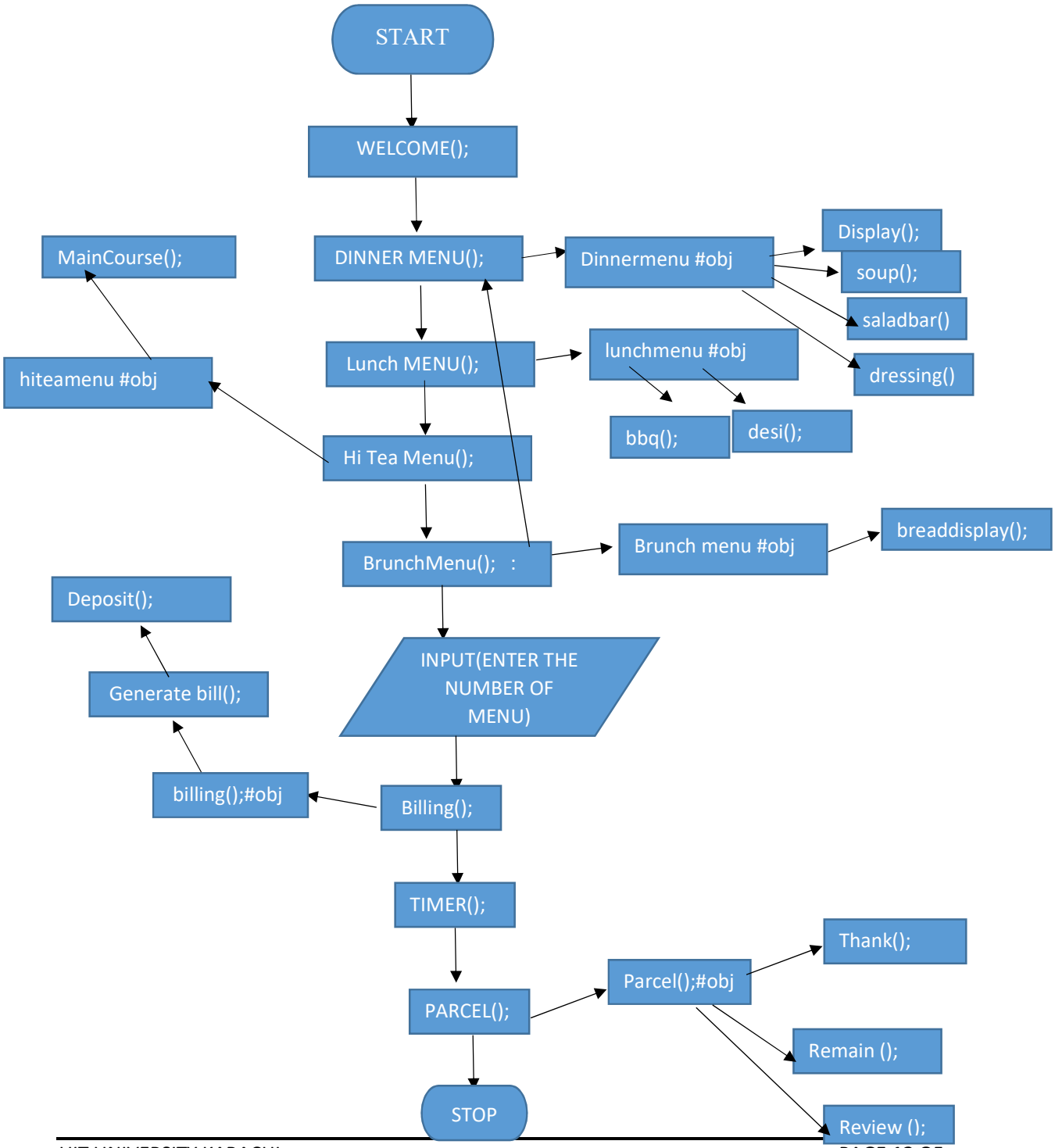
```
        cout << " Rs 3715 per member FOR DINNER" << endl;
        int members;
        cout<<"ENTER NUMBER OF MEMBERS:"<<endl;
        cin>>members;
        cout<<"TOTAL BILL AMOUNT: " <<members*3715<<endl;
    } else if (n == 2) {
        cout << " Rs  2020 per member FOR LUNCH" << endl;
        int members;
        cout<<"ENTER NUMBER OF MEMBERS:"<<endl;
        cin>>members;
        cout<<"TOTAL BILL AMOUNT: " <<members*2020<<endl;
    } else if (n == 3) {
        cout << " Rs 1907 per member FOR HI TEA MENU" << endl;
        int members;
        cout<<"ENTER NUMBER OF MEMBERS:"<<endl;
        cin>>members;
        cout<<"TOTAL BILL AMOUNT: " <<members*1907<<endl;
    } else if (n == 4) {
        cout << "Rs 2359 per member FOR BRUNCH MENU" << endl; // Replace XXXXX
with actual brunch bill
        int members;
        cout<<"ENTER NUMBER OF MEMBERS:"<<endl;
        cin>>members;
        cout<<"TOTAL BILL AMOUNT: " <<members*2359<<endl;
    }
  }

  void deposit() {
    cout << "ENTER YOUR BILL AMOUNT CASH OR CREDIT CARD NO" << endl;
    int x;
    cout << "ENTER YOUR CHOICE 1 OR 2 WHICH METHOD" << endl;
```

```
 cin >> x;
      if (x == 1) {
         cout << "ENTER YOUR CASH AMOUNT" << endl;
         int mon;
         cin>>mon;



      } else if (x == 2) {
         cout << "ENTER YOUR CREDIT CARD NUMBER" << endl;
         string id;
         cin >> id;
      }
   }
};
class Parcel {
public:
   void thank() {
      cout << "THANKS FOR COMING TO OUR RESTAURANT" << endl;
   } void remain() {
      cout << "SIR, ANY REMAINING ITEMS SHOULD BE PARCELED?" << endl;
      char option;
      cout << "ENTER YES OR NO: ";
      cin >> option;
      if (option == 'y' || option == 'Y') {
         cout << "OK SIR, SURE I WILL PARCEL IT" << endl;
         PlaySound(TEXT("P.wav"), NULL, SND_FILENAME);
      } else if (option == 'n' || option == 'N') {
         cout << "OK SIR, TAKE CARE BYE!!" << endl;
         cout << "INVALID OPTION" << endl;
      }
   }
```

```
    void review() {
        string r;
        cout << "SHARE YOUR REVIEW ABOUT THIS RESTAURANT: ";
        cin >> r;
        cout << "THANKS FOR SHARING IT!!" << endl;
    }
};
int main() {
    Welcome welcome; // Create an object of the Welcome class
    DinnerMenu dinnerMenu;
    dinnerMenu.display();
    dinnerMenu.soup();
    dinnerMenu.saladbar();
    dinnerMenu.dressing();
    LunchMenu lunchMenu;
    lunchMenu.bbq();
    lunchMenu.desi();
    HiTeaMenu hiTeaMenu;
    hiTeaMenu.mainCourse();
   BrunchMenu brunchMenu;
    brunchMenu.breaddisplay();
    brunchMenu.soup();
    Sleep(150);
    system("cls");
    Booking booking;
    booking.people();

    int menuChoice;
    cout << "ENTER THE NUMBER OF MENU YOU WANT TO GENERATE BILL FOR:
";
```

```
    cin >> menuChoice;
  Billing billing;
      billing.generateBill(menuChoice);
              billing.deposit();
              Sleep(10000);
              system("cls");
    Timer timer;
    timer.startTimer(02);
    Parcel parcel;
    parcel.thank();
    parcel.remain();
    parcel.review();
    // Start a short timer
    timer.startShortTimer();
    return 0;
}
```

FLOW-CHART:


START
WELCOME();
DINNER MENU();
Dinnermenu #obj
Display();
soup();
saladbar()
dressing()
MainCourse();
Lunch MENU();
lunchmenu #obj
bbq();
desi();
hiteamenu #obj
Hi Tea Menu();
BrunchMenu(); :
Brunch menu #obj
breaddisplay();
Deposit();
INPUT(ENTER THE NUMBER OF MENU)
Generate bill();
billing();#obj
Billing();
TIMER();
Thank();
Parcel();#obj
PARCEL();
Remain ();
STOP
Review ();