PIMPRI CHINCHWAD EDUCATION TRUST's.

PIMPRI CHINCHWAD COLLEGE OF ENGINEERING

(An Autonomous Institute)

S.Y. B. TECH **Year:** 2024 – 25 **Semester:** I

Name: Abhishek Joshi **PRN: 123B1B150**

Department: Computer Engineering **Division:** C (C1)

Course: Data Structures Laboratory **Course Code:** BCE23PC02

Date:

Assignment – 9A

- **Aim:**

Implement a restaurant waitlist system using the queue data structure. Restaurant waitlist provide following facility:

a. Add Party to Waitlist

b. Seat Party

c. Display Waitlist

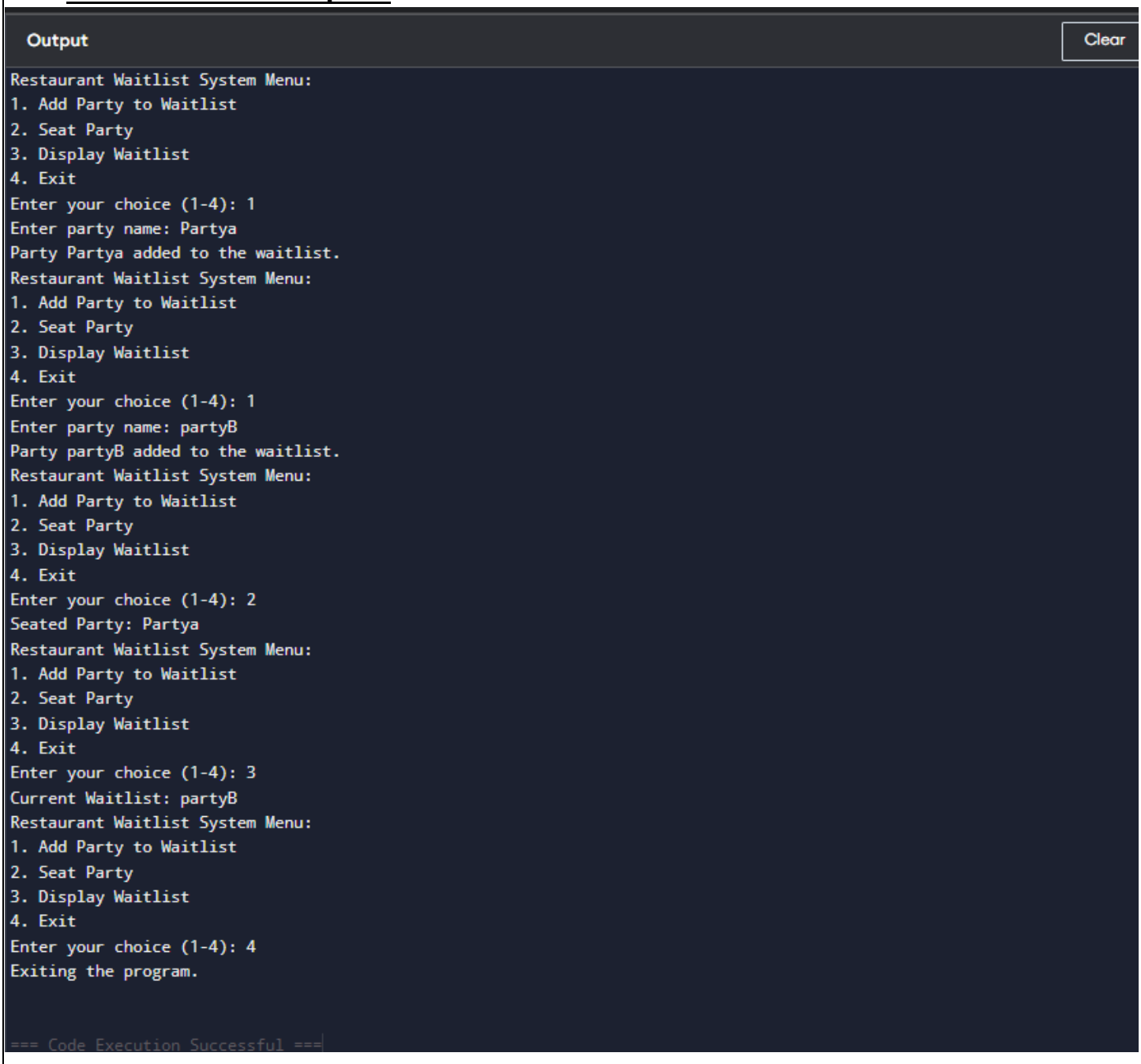
- **Source Code :**

```
#include <iostream>
#include <string>
using namespace std;

class Queue {
public:
  int front;
  int rear;
  string arr[5];

  Queue() {
    front = -1;
    rear = -1;
  }

  void addParty(string partyName) {
    if (rear == 5 - 1) {
      cout << "Waitlist is FULL: Party " << partyName << " cannot be added." << endl;
    } else {
      if (front == -1) {
```

```
            front = 0;
        }
        rear++;
        arr[rear] = partyName;
        cout << "Party " << partyName << " added to the waitlist." << endl;
    }
}

string seatParty() {
    if (front == -1 || front > rear) {
        cout << "Waitlist is empty." << endl;
        return "";
    }
    string partyName = arr[front];
    front++;

    if (front > rear) {
        front = rear = -1;
    }
    return partyName;
}

void displayWaitlist() {
    if (front == -1 || front > rear) {
        cout << "Waitlist is empty." << endl;
        return;
    }
    cout << "Current Waitlist: ";
    for (int i = front; i <= rear; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
}
};

int main() {
    Queue waitlist;
    int choice;
    string partyName;

    do {
```

```
        cout << "Restaurant Waitlist System Menu:" << endl;
        cout << "1. Add Party to Waitlist" << endl;
        cout << "2. Seat Party" << endl;
        cout << "3. Display Waitlist" << endl;
        cout << "4. Exit" << endl;
        cout << "Enter your choice (1-4): ";
        cin >> choice;

        switch (choice) {
            case 1:
                cout << "Enter party name: ";
                cin >> partyName;
                waitlist.addParty(partyName);
                break;

            case 2:
                partyName = waitlist.seatParty();
                if (!partyName.empty()) {
                    cout << "Seated Party: " << partyName << endl;
                }
                break;

            case 3:
                waitlist.displayWaitlist();
                break;

            case 4:
                cout << "Exiting the program." << endl;
                break;

            default:
                cout << "Invalid choice. Please enter a number between 1 and 4." << endl;
                break;
        }
    } while (choice != 4);
    return 0;
}
```

- **Screen Shot of Output :**



- **Conclusion:**

  In this assignment, we implemented a restaurant waitlist system using arrays to demonstrate queue operations for adding parties, seating them, and displaying the current waitlist. The program effectively handles basic queue functionalities while providing a user-friendly menu interface.