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 – 5

- **Aim:**

  Implement a navigation system for a delivery service using a circular linked list to represent routes. The navigation system should support the following functionalities:
  a. Add route
  b. Remove route
  c. Display route

- **Source Code :**

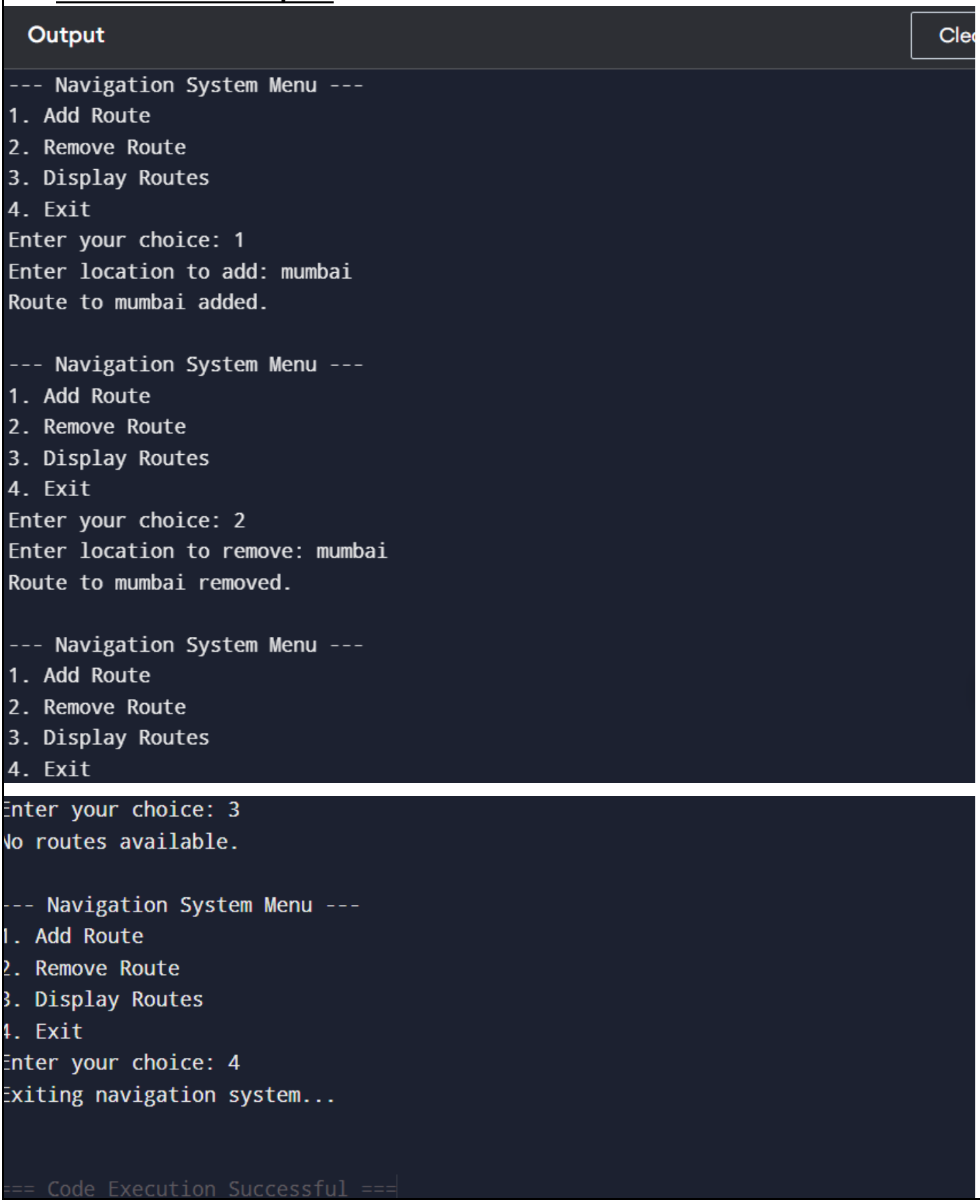
```
#include <iostream>
#include <string>
using namespace std;

class node {
public:
   string location;
   node *next;

   node(string loc) {
      location = loc;
      next = NULL;
   }
};

class NavigationSystem {
   node *head;
public:
   NavigationSystem() {
```

```
    head = NULL;
}

void addRoute(string loc) {
    node *newNode = new node(loc);
    if (!head) {
        head = newNode;
        head->next = head;
    } else {
        node *temp = head;
        while (temp->next != head) {
            temp = temp->next;
        }
        temp->next = newNode;
        newNode->next = head;
    }
    cout << "Route to " << loc << " added." << endl;
}

void removeRoute(string loc) {
    if (!head) {
        cout << "No routes to remove." << endl;
        return;
    }

    node *curr = head;
    node *prev = NULL;

    do {
        if (curr->location == loc) {
            if (prev) {
                prev->next = curr->next;
            } else {
                if (head->next == head) {
                    head = NULL;
                } else {
                    node *temp = head;
```

```
                while (temp->next != head) {
                    temp = temp->next;
                }
                head = head->next;
                temp->next = head;
            }
        }
        delete curr;
        cout << "Route to " << loc << " removed." << endl;
        return;
      }
      prev = curr;
      curr = curr->next;
    } while (curr != head);

    cout << "Route to " << loc << " not found." << endl;
  }

  void displayRoutes(){
    if (head == NULL) {
      cout << "No routes available." << endl;
      return;
    }

    node *temp = head;
    cout << "curr Routes:" << endl;
    do {
      cout << temp->location << " -> ";
      temp = temp->next;
    } while (temp != head);
    cout << "(back to start)" << endl;
  }
};
int main() {
  NavigationSystem navigation;
  int choice;
  string location;
```

```
    do {
        cout <<endl<< "--- Navigation System Menu ---"<<endl;
        cout << "1. Add Route"<<endl;
        cout << "2. Remove Route"<<endl;
        cout << "3. Display Routes"<<endl;
        cout << "4. Exit"<<endl;
        cout << "Enter your choice: ";
        cin >> choice;
        cin.ignore();

        switch (choice) {
            case 1:
                cout << "Enter location to add: ";
                getline(cin, location);
                navigation.addRoute(location);
                break;

            case 2:
                cout << "Enter location to remove: ";
                getline(cin, location);
                navigation.removeRoute(location);
                break;

            case 3:
                navigation.displayRoutes();
                break;

            case 4:
                cout << "Exiting navigation system..." << endl;
                break;

            default:
                cout << "Invalid choice! Please try again." << endl;
        }
    } while (choice != 4);
    return 0;
}
```

- **Screen Shot of Output :**



- **Conclusion:**

  Hence, we studied about application of Circular Linked List as Navigation System by Adding Routes, Removing Routes, and Displaying Routes with their algorithm and programs.