JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA

B. TECH 4TH SEMESTER

REPORT FOR MINOR PROJECT IN ALGORITHMS

TITLE OF PROJECT

'METRO APPLICATION'

Supervision of:

Dr. Taj Alam

Assistant Professor

Department Of Computer

Science and Engineering

JIIT, SECTOR 62,NOIDA

Submitted by:

Name	Enroll No.
SABEEH AHSAN	22803006
TANMAY BUTTA	22803014
LAKSHYAVEER SINGH	22803015

PROBLEM STATEMENT:-

Design and implement a Metro Application that facilitates efficient route planning and navigation within a metro system. The application should utilize Dijkstra's algorithm to find the shortest path between two stations, considering both distance and time factors.

MOTIVATION:-

The idea for the Metro Application using Dijkstra's algorithm in C++ is to make traveling on metro trains easier. We want to help people find the fastest and cheapest routes to their destinations. By using this app, users can save time and money while navigating through the metro system. It's like having a personal guide to help you get around town quickly and efficiently. Our goal is to make urban transportation smoother and more convenient for everyone.

OBJECTIVE:-

The objective of this project is to help the users find the shortest path between their source and destination station, So that the time and distance can be minimized so that efficient route planning is achieved.

CONTRIBUTION

SABEEH AHSAN	Class graph and its functions
TANMAY BUTTA	Class dijkstra and its functions
LAKSHYAVEER SINGH	Class dijkstra pair and its functions

Features of the Project:

1) LIST ALL THE STATIONS IN THE MAP.
2) SHOW THE METRO MAP.
3) GET SHORTEST DISTANCE FROM A 'SOURCE' STATION TO 'DESTINATION'.
4)GET SHORTEST TIME TO REACH FROM A 'SOURCE' STATION TO 'DESTINATION' STATION.

Code Snippets:-

```
int main()
{

    Graph_M g;
    Graph_M::Create_Metro_Map(g);

     cout << "\n\t\t\t****WELCOME TO THE METRO APP*****" << endl;

    while (true)
    {
        cout << "\t\t\t\t~~LIST OF ACTIONS~~\n\n";
        cout << "1. LIST ALL THE STATIONS IN THE MAP" << endl;
        cout << "2. SHOW THE METRO MAP" << endl;
        cout << "3. GET SHORTEST DISTANCE FROM A 'SOURCE' STATION TO 'DESTINATION' STATION" << endl;
        cout << "4. GET SHORTEST TIME TO REACH FROM A 'SOURCE' STATION TO 'DESTINATION' STATION" << endl;
        cout << "5. GET SHORTEST PATH (DISTANCE WISE) TO REACH FROM A 'SOURCE' STATION TO 'DESTINATION' STATION" << endl;
        cout << "6. GET SHORTEST PATH (TIME WISE) TO REACH FROM A 'SOURCE' STATION TO 'DESTINATION' STATION" << endl;
        cout << "7. EXIT THE MENU" << endl;
        cout << "\nENTER YOUR CHOICE FROM THE ABOVE LIST (1 to 7) : ";
        int choice = -1;
        cin >> choice;
        cout << "\n************************************************************\n";
        if (choice == 7)
        {
            break;
        }
        switch (choice)
        {
        case 1:
            g.display_Stations();
            break;

        case 2:
            g.display_Map();
            break;

        case 3:
        {
            string *keys = printCodelist();
            cout << "\n1. TO ENTER SERIAL NO. OF STATIONS\n2. TO ENTER CODE OF STATIONS\n3. TO ENTER NAME OF STATIONS\n";
```

TEST CASES:-

```
                        ****WELCOME TO THE METRO APP*****
                                ~~LIST OF ACTIONS~~

1. LIST ALL THE STATIONS IN THE MAP
2. SHOW THE METRO MAP
3. GET SHORTEST DISTANCE FROM A 'SOURCE' STATION TO 'DESTINATION' STATION
4. GET SHORTEST TIME TO REACH FROM A 'SOURCE' STATION TO 'DESTINATION' STATION
5. GET SHORTEST PATH (DISTANCE WISE) TO REACH FROM A 'SOURCE' STATION TO 'DESTINATION' STATION
6. GET SHORTEST PATH (TIME WISE) TO REACH FROM A 'SOURCE' STATION TO 'DESTINATION' STATION
7. EXIT THE MENU

ENTER YOUR CHOICE FROM THE ABOVE LIST (1 to 7) :
```

```
1. Rajouri Garden~BP
2. IGI Airport~O
3. Netaji Subhash Place~PR
4. Moti Nagar~B
5. Rajiv Chowk~BY
6. AIIMS~Y
7. Yamuna Bank~B
8. Botanical Garden~B
9. Janak Puri West~BO
10. Huda City Center~Y
11. Vishwavidyalaya~Y
12. New Delhi~YO
13. Shivaji Stadium~O
14. Vaishali~B
15. Saket~Y
16. Dwarka Sector 21~B
17. Chandni Chowk~Y
18. Punjabi Bagh West~P
19. Noida Sector 62~B
20. DDS Campus~O
```

```
         Delhi Metro Map
         ---------------
----------------------------------------

Rajouri Garden~BP =>
        Punjabi Bagh West~P     2
        Moti Nagar~B            2

IGI Airport~O =>
        DDS Campus~O            8

Netaji Subhash Place~PR =>
        Punjabi Bagh West~P     3

Moti Nagar~B =>
        Rajouri Garden~BP       2
        Rajiv Chowk~BY          9
        Janak Puri West~BO      7

Rajiv Chowk~BY =>
        New Delhi~YO            1
        AIIMS~Y                 7
        Yamuna Bank~B           6
        Moti Nagar~B            9

AIIMS~Y =>
        Rajiv Chowk~BY          7
        Saket~Y                 6
```

```
1. TO ENTER SERIAL NO. OF STATIONS
2. TO ENTER CODE OF STATIONS
3. TO ENTER NAME OF STATIONS
ENTER YOUR CHOICE:3
ENTER THE SOURCE AND DESTINATION STATIONS
New Delhi~YO
Saket~Y
SHORTEST DISTANCE FROM New Delhi~YO TO Saket~Y IS 14KM
```

```
ENTER YOUR CHOICE FROM THE ABOVE LIST (1 to 7) : 4

**********************************************************
ENTER THE SOURCE STATION: New Delhi~YO
ENTER THE DESTINATION STATION: Saket~Y
SHORTEST TIME FROM (New Delhi~YO) TO (Saket~Y) IS 15 MINUTES
                                    ~~LIST OF ACTIONS~~
```

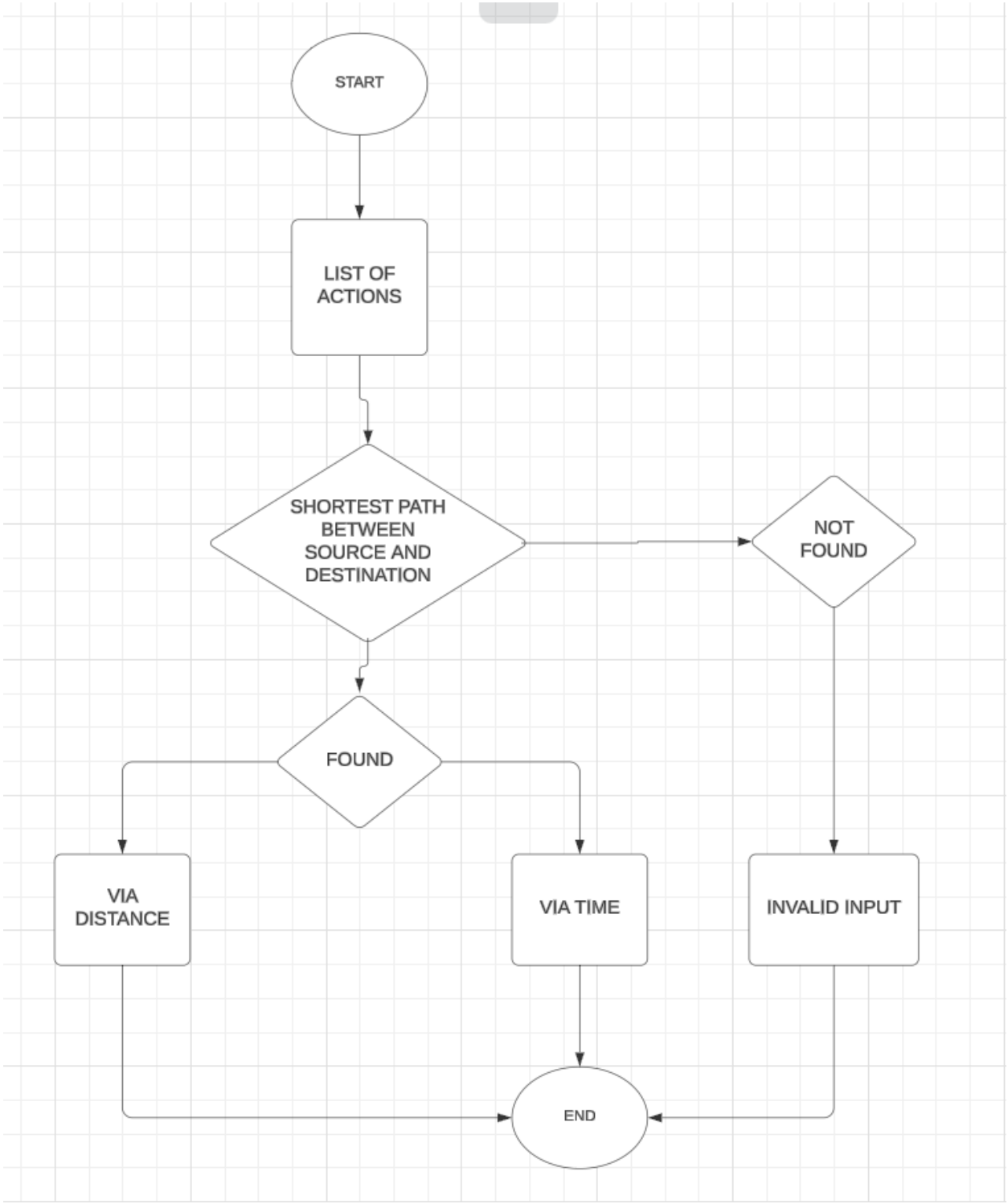
Algorithm Description

Dijkstra's algorithm is a popular algorithm used to find the shortest paths from a single source node to all other nodes in a weighted graph. It was conceived by Dutch computer scientist Edsger W. Dijkstra in 1956 and is widely used in various applications, including network routing protocols and transportation systems.

Operations used:

1. Weighted Graphs
2. Unordered map
3. Vectors
4. Pairs
5. Stack

DATA FLOW DIAGRAM



FUTURE SCOPES:-

1) Making a fare calculator in this application
2) Creating a GUI for this application
3) Taking feedback from users to improve more

Conclusion:-

In conclusion, the Metro Application utilizing Dijkstra's algorithm in C++ presents a valuable solution for improving urban transportation by providing users with efficient route planning, real-time updates, and personalized travel experiences. With its user-friendly interface, helpful features, and focus on accessibility, the application aims to simplify metro travel and promote sustainable transportation options for urban commuters. As technology continues to evolve, there is immense potential for further enhancements and innovations to enhance the functionality and reach of the Metro Application, ultimately contributing to the improvement of urban mobility and quality of life in cities worldwide.

References:-

https://www.geeksforgeeks.org/dijkstras-shortest-path-algorithm-greedy-algo-7/
https://www.geeksforgeeks.org/introduction-to-dijkstras-shortest-path-algorithm/