**Experiment -7**

**There are flight paths between cities. If there is a flight between city A and city B then there is an edge between the cities. The cost of the edge can be the time that flight take to reach city B from A, or the amount of fuel used for the journey. Represent this as a graph. The node can be represented by airport name or name of the city. Use adjacency list representation of the graph or use adjacency matrix representation of the graph. Check whether the graph is connected or not. Justify the storage representation used.**

#include<iostream>

#include<string.h>

using namespace std;

class flight

{

public:

int am[10][10]; //Amount array

char city\_index[10][10]; // City name array

flight(); //Constructor

int create(); // To function create

void display(int city\_count); //To display adj matrix

};

flight::flight() // Constructor of class flight

{

int i,j;

for(i=0;i<10;i++)

{

strcpy(city\_index[i],"xx");

}

for(i=0;i<10;i++)

{

for(j=0;j<10;j++)

{

am[i][j]=0;

}

}

}

int flight::create() // To create matrix

{

int city\_count=0,j,si,di,wt; //Declaration and initialization of variable

char s[10],d[10],c;

do

{

cout<<"\n\tEnter Source City : "; //Ask user source city:s

cin>>s;

cout<<"\n\tEnter Destination City : "; //Ask user destination city:d

cin>>d;

for(j=0;j<10;j++)

{

if(strcmp(city\_index[j],s)==0) //if source city already available then break

break;

}

if(j==10)

{ // if source city is not available the copy source city at current index .

strcpy(city\_index[city\_count],s);

city\_count++;

}

for(j=0;j<10;j++) //Initially city-count=0

{

if(strcmp(city\_index[j],d)==0)//if destination city already available then break.

break;

}

if(j==10)

{// if destination city is not available the copy destination city at current index .

strcpy(city\_index[city\_count],d);

city\_count++;

}

//Ask user for distance from source to destination.

cout<<"\n\t Enter Distance From "<<s<<" And "<<d<<": ";

cin>>wt;

for(j=0;j<10;j++)

{

//search that source and destination index in matrix and copy them in si and di respectively.

if(strcmp(city\_index[j],s)==0)

si=j;

if(strcmp(city\_index[j],d)==0)

di=j;

}

am[si][di]=wt; // Insert that wt in new index si di in array of amount.

cout<<"\n\t Do you want to add more cities.....(y/n) : ";

cin>>c;

}while(c=='y'||c=='Y');

return(city\_count);

}

void flight::display(int city\_count) //Display citi count

{

int i,j;

cout<<"\n\t Displaying Adjacency Matrix :\n\t";

for(i=0;i<city\_count;i++)

cout<<"\t"<<city\_index[i]; // Display city name

cout<<"\n";

for(i=0;i<city\_count;i++)

{

cout<<"\t"<<city\_index[i];

for(j=0;j<city\_count;j++)

{

cout<<"\t"<<am[i][j]; // Display city name with amount.

}

cout<<"\n";

}

}

int main()

{

flight f;

int n,city\_count; // variable declaration as per requirement

char c;

do

{

cout<<"\n\t\*\*\*\*\* Flight Main Menu \*\*\*\*\*";

cout<<"\n\t1. Create \n\t2. Adjacency Matrix\n\t3. Exit"; //Menu for the user

cout<<"\n\t.....Enter your choice : ";

cin>>n;

switch(n)

{

case 1:

city\_count=f.create(); // To call create function which return city-count

break;

case 2:

f.display(city\_count); //To call Display function which represent adj matrix

break;

case 3:

return 0;

}

cout<<"\n\t Do you Want to Continue in Main Menu....(y/n) : ";

cin>>c;

}while(c=='y'||c=='Y');

return 0;

}

\*/OUTPUT:

\*\*\*\*\* Flight Main Menu \*\*\*\*\*

1. Create

2. Adjacency Matrix

3. Exit

.....Enter your choice : 1

Enter Source City : M

Enter Destination City : P

Enter Distance From M And P: 200

Do you want to add more cities.....(y/n) : Y

Enter Source City : NS

Enter Destination City : P

Enter Distance From NS And P: 150

Do you want to add more cities.....(y/n) : Y

Enter Source City : NG

Enter Destination City : M

Enter Distance From NG And M: 500

Do you want to add more cities.....(y/n) : Y

Enter Source City : NG

Enter Destination City : NA

Enter Distance From NG And NA: 600

Do you want to add more cities.....(y/n) : Y

Enter Source City : S

Enter Destination City : P

Enter Distance From S And P: 100

Do you want to add more cities.....(y/n) : N

Do you Want to Continue in Main Menu....(y/n) : Y

\*\*\*\*\* Flight Main Menu \*\*\*\*\*

1. Create

2. Adjacency Matrix

3. Exit

.....Enter your choice : 2

Displaying Adjacency Matrix :

M P NS NG NA S

M 0 200 0 0 0 0

P 0 0 0 0 0 0

NS 0 150 0 0 0 0

NG 500 0 0 0 600 0

NA 0 0 0 0 0 0

S 0 100 0 0 0 0

Do you Want to Continue in Main Menu....(y/n) :