Assignment No: 6

```
#include<iostream>
#include<queue>
#include<stack>
using namespace std;
class Graph
  string city[10];
  int a[10][10];
  int n;
  public:
  void input();
  void display();
  void BFS();
  void DFS();
};
void Graph::input()
  cout<<"\n Enter No. of cities :";</pre>
  cin>>n;
  cout<<"\n Enter the names of cities :";</pre>
  for(int i=0; i<n; i++)
     cin>>city[i];
  cout<<"\nEnter the distances :";</pre>
  for(int i=0; i<n; i++)
     for(int j=0; j< n; j++)
     {
        if(i==j)
           a[i][j] = 0;
           continue;
        }
        cout<<"\n Enter the dinstance between"<<city[i]<<"\t"<<"and"<<"\t"<<city[j]<<":";
        cin>>a[i][j];
        a[j][i] = a[i][j];
     }
}
void Graph::display()
  for(int i=0; i<n; i++)
     cout<<"\n";
     for(int j=0; j< n; j++)
        cout << a[i][j] << "\backslash t";
}
void Graph::BFS()
   cout<<"\n\n BFS Traversal :";</pre>
```

```
queue<int>q;
   int visit[n];
   for(int i=0; i<n; i++)
      visit[i] = 0;
   string start;
   int index;
   cout<<"\n Enter starting city :";</pre>
   cin>>start;
   for(int i=0; i<n; i++)
      if(start == city[i])
           index = i;
   visit[index] = 1;
   cout<<city[index]<<"->";
   int current = index;
   while(1)
   {
      for(int i=0; i<n; i++)
        if(a[current][i] != 0 && visit[i] ==0)
            visit[i] = 1;
            q.push(i);
            cout<<city[i]<<"->";
         }
      }
      if(q.empty()!=0)
         break;
      else
        current = q.front();
        q.pop();
  }
}
void Graph::DFS()
  cout<<"\n\nDFS Traversal :";</pre>
  stack<int>s;
  int visit[n];
  for(int i=0; i<n; i++)
      visit[i] = 0;
  string start;
  int index;
  cout<<"\n Enter starting city :";</pre>
  cin>>start;
  for(int i=0; i<n; i++)
      if(start == city[i])
          index = i;
  s.push(index);
  visit[index] = 1;
  int current = index;
  cout<<city[index]<<"->";
  while(1)
  {
     for(int i=0; i<n; i++)
      {
```

```
if(a[current][i] != 0 && visit[i] ==0)
           s.push(i);
            cout << city[i] << "-> ";
           visit[i] = 1;
            current = i;
           i=0;
         }
      if(s.empty()!=0)
         break;
      else
         current = s.top();
         s.pop();
  }
}
int main()
  Graph g1;
  int choice;
  MENU:
  cout<<"\n\nGRAPH TRAVERSAL...";</pre>
  cout<<"\n1. Input data";</pre>
  cout << "\n2. Display data";
  cout<<"\n3. DFS Traversal";</pre>
  cout<<"\n4. BFS Traversal";</pre>
  cout << "\n5. Exit";
  cout<<"\nEnter your choice :";</pre>
  cin>>choice;
  switch(choice)
  {
      case 1:
          g1.input();
          break;
      case 2:
          g1.display();
          break;
      case 3:
          g1.DFS();
          break;
      case 4:
          g1.BFS();
          break;
      case 5:
          return 0;
      default:
          cout<<"\nInvalid choice. Try again!";</pre>
  }
  if(choice !=5)
      goto MENU;
  return 0;
}
```

```
#include<queue
#include<queue
#include<stack
cout"\nS. Exit";
using namespac
student@student-OptiPlex-3010:~/Desktop/Nikita$ g++ Ass6.cpp
student@student-OptiPlex-3010:~/Desktop/Nikita$ ,/a.out
      string citGRAPH TRAVERSAL...
int a[10][1. Input data
int n; 2. Display data
public: 3. DFS Traversal
void input 4. BFS Traversal
void displemer your choice:1
void BFS()
void DFS() Enter No. of cities:4
cout<<"\nE Enter the dinstance betweenpuneandlatur:654
        for(in i=
for(in Enter the dinstance betweenmumbalandpune:123

{
    Enter the dinstance betweenmumbalandpashik:3
                      if Enter the dinstance betweenmumbalandnashik:3456
                      Enter the dinstance betweenmumbaiandlatur:234
                          Enter the dinstance betweennashikandnune:656
                    Enter the dinstance betweennashikandmumbai:134
                     ci Enter the dinstance betweennashikandlatur:5566 al Enter the dinstance betweenlaturandpune:345
                         Enter the dinstance betweenlaturandmumbai:5667
 oid Graph::display()
                               🕒 📵 student@student-OptiPlex-3010: ~/Desktop/Nikita
#include<queue
#include<queue
#include<queue
#include<stack 3. DFS Traversal
using namespac 5. Extt
Enter your choice :2
     ss Graph

string cit123 0 134
int a[10][656 134 0
int n; 345 5667 4556
public:
coid input1. Input data
void displ2. Display data
void BFS()3. DFS Traversal
void DFS()4. BFS Traversal
5. Extt
Enter your choice :3
d Graph::in
                                                                      345
5667
4556
0
oid Graph::in
     { BFS Traversal :
Enter starting city :pune
pune->mumbai->nashik->latur->
                          GRAPH TRAVERSAL...
                     Col. Input data
ci. Display data
ci. Display data
al. DFS Traversal
4. BFS Traversal
5. Exit
Enter your choice :5
```