7. Graph: Minimum Spanning Tree

TITLE: Represent a graph of your college campus using adjacency list /adjacency matrix. Nodes should represent the various departments/institutes and links should represent the distance between them. Find minimum spanning tree-

a) Using Kruskal's algorithm.

b) Using Prim's algorithm.

OBJECTIVE:

1. Learn the concepts of graph as a data structure and their applications in everyday life.
2. Understand graph representation (adjacency matrix, adjacency list, adjacency multi list)

THEORY:

1. What is a graph? Various terminologies and its applications. Explain in brief.

Definition : A graph is a triple G = (V,E, φ) where

• V is a finite set, called the vertices of G,

• E is a finite set, called the edges of G, and

• φ is a function with domain E and codomain P2(V).

Loops: A loop is an edge that connects a vertex to itself.

Degrees of vertices: Let G = (V,E, φ) be a graph and v ∈ V a vertex.

Define the degree of v, d(v) to be the number of e ∈ E such that v ∈ φ(e); i.e., e is Incident on v.

Directed graph: A directed graph (or digraph) is a triple D = (V,E, φ) where V and E are finite sets and φ is a function with domain E and codomain V × V . We call E the set of edges of the digraph D and call V the set of vertices of D.

Path: Let G = (V,E, φ) be a graph.

Let e1, e2, . . . , en−1 be a sequence of elements of E (edges of G) for which there is a sequence a1, a2, . . . , an of distinct elements of V (vertices of G) such that φ(ei) = {ai, ai+1} for i = 1, 2, . . . , n − 1. The sequence of edges e1, e2, . . . , en−1 is called a path in G. The sequence of vertices a1, a2, . . . , an is called the vertex sequence of the path. **Circuit and Cycle**: Let G = (V,E, φ) be a graph and let e1, . . . , en be a trail with vertex sequence a1, . . . , an, a1. (It returns to its starting point.) The subgraph G′ of G induced by the set of edges {e1, . . . , en} is called a circuit of G. The length of the circuit is n. 2. Different representations of graph.

Adjacency matrix:

Graphs $G = (V, E)$ can be represented by **adjacency matrices** $G[v1..v|V|, v1..v|V|]$, where the rows and columns are indexed by the nodes, and the entries $G[vi, vj]$ represent the edges. In Incident on v.

Directed graph: A directed graph (or digraph) is a triple D = (V,E, φ) where V and E are finite sets and φ is a function with domain E and codomain V × V . We call E the set of edges of the digraph D and call V the set of vertices of D.

Path: Let G = (V,E, φ) be a graph.

Let e1, e2, . . . , en−1 be a sequence of elements of E (edges of G) for which there is a sequence a1, a2, . . . , an of distinct elements of V (vertices of G) such that φ(ei) = {ai, ai+1} for i = 1, 2, . . . , n − 1. The sequence of edges e1, e2, . . . , en−1 is called a path in G. The sequence of vertices a1, a2, . . . , an is called the vertex sequence of the path. **Circuit and Cycle**: Let G = (V,E, φ) be a graph and let e1, . . . , en be a trail with vertex sequence a1, . . . , an, a1. (It returns to its starting point.) The subgraph G′ of G induced by the set of edges {e1, . . . , en} is called a circuit of G. The length of the circuit is n. 2. Different representations of graph.

Adjacency matrix:

Graphs $G = (V, E)$ can be represented by **adjacency matrices** $G[v1..v|V|, v1..v|V|]$, where the rows and columns are indexed by the nodes, and the entries $G[vi, vj]$ represent the edges. In

In computer science, Prim's algorithm is a greedy algorithm that finds a minimum spanning tree for a connected weighted undirected graph. This means it finds a subset of the edges that forms a tree that includes every vertex, where the total weight of all the edges in the tree is minimized. Other algorithms for this problem include Kruskal's algorithm and Borůvka's algorithm. However, these other algorithms can also find minimum spanning forests of disconnected graphs, while Prim's algorithm requires the graph to be connected. Kruskal‘s Algorithm may also be used to find a minimum spanning tree, but this considers the weights themselves rather than the connecting points

Algorithm:

1. **Prim's Algorithm:**

   Step 1: Select any node to be the first node of T.

   Step 2: Consider the arcs which connect nodes in T to nodes outside T. Pick the one with minimum weight. Add this arc and the extra node to T. (If there are two or more arcs of minimum weight, choose any one of them.)

   Step 3: Repeat Step 2 until T contains every node of the graph.

2. **Kruskal's Algorithm:**

   Step 1: Choose the arc of least weight.

   Step 2: Choose from those arcs remaining the arc of least weight which does not form a cycle with already chosen arcs. (If there are several such arcs, choose one arbitrarily.)

   Step 3: Repeat Step 2 until $n – 1$ arcs have been chosen.

INPUT:

Enter the no. of nodes in graph. Create the adjacency LIST.

OUTPUT:

Display result of each operation with error checking.

FAQ:

1. What is graph?
2. Application of Prim's & Kruskal's algorithm.
3. What are the traversal techniques?
4. What are the graph representation techniques?
5. What is adjacency Matrix?
6. What is adjacency list?
7. What is adjacency Multi-list?

CONCLUSION: Thus, we have studied and implement Kruskal‘s algorithm and Prim‘s algorithm.