**GRAPHS**

Trees are limited in that a data structure can only have one parent, graphs overcame this

Graph algorithms run

* large communication networks
* the software that makes the internet function
* programs to determine optimal placement of components on a silicon chip

Graphs describe

* roads maps
* airline routes
* course prerequisites

**Graph Terminology**

Graph is a data structure that consists of:

* set of vertices (or nodes)
* set of edges (relations) between pairs of vertices

Edges represent relations or connections between vertices

Both the set of vertices and the set of edges must be finite

If the set of vertices is finite, set of edges may not be finite because 2 vertices may have several edges. So edge number may be infinite when vertices number is finite, it is not possible.

Either set may be empty (if the set of vertices is empty, the set of edges also must be empty)

We restrict our discussion to simple graphs in which there is at most one edge from a given vertex to another vertex (and there is no self edge from a vertex to itself for our discussion)



Graph is a tuple of vertex set (V) and edge set (E).

A picture containing diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated with medium confidenceIf there is an edge between vertices x and y, there is a connection from x to y and vice versa

If there is a connection from x to y but not from y to x, then representation would be as tuple (x,y) in edge set because there is no hierarchy in set

A graph with directed edges is called a directed graph or digraph

A graph with undirected edges is an undirected graph, or simply a graph

Directed edges are represented by ordered pairs of vertices (source, destination); the edges for the digraph on this slide are:

{(A, B), (B, A), (B, E), (D, A), (E, A), (E, C), (E, D)}

A is adjacent to D, but D is not adjacent to A

The edges in a graph may have associated values known as their weights

A graph with weighted edges is known as a weighted graph

Chart, radar chart

Description automatically generated

In directed graph, weight for to go from A to B may be 5 and weight for to go from B to A may be 10. It may not be the same. In undirected graph, both ways are same.

A vertex is adjacent to another vertex if there is an edge to it from that other vertex

**Paths and Cycles**

In a path, we have list of vertices which two consecutive vertices are adjacent

A path is a sequence of vertices in which each successive vertex is adjacent to its predecessor

In a simple path, the vertices and edges are distinct except that the first and last vertex may be the same

Chart, radar chart

Description automatically generated

Chart, radar chart

Description automatically generated

This cycle (starting from Columbus and end in Columbus) is a cycle as well but it is not a simple cycle since there is another cycle in it.

Chart, radar chart

Description automatically generated

In an undirected graph a cycle must contain at least 3 distinct vertices

* Pittsburgh 🡪 Columbus 🡪 Pittsburgh : is not a a cycle

An undirected graph is called connected graph if there is a path from every vertex to every other vertex

If a graph is not connected, it is considered unconnected but still consists of connected components

Diagram

Description automatically generatedThis graph is unconnected but still consists of connected components

{4, 5} & {{4, 5}} 🡪 connected component

{6, 7, 8, 9} & {{6, 7}, {6, 8}, {6, 9}, {8, 9}} 🡪 connected component

Chart

Description automatically generated with low confidence

**Relationship between Graphs and Trees**

A tree is a special case of a graph

Any graph that is

* connected
* contains no cycles

can be viewed as a tree by making one of the vertices the root

A picture containing timeline

Description automatically generated

Trees are generalization of Linked Lists, Graphs are generalization of Trees.

**Graph Applications**

Graphs can be used to:

* determine if one node in a network is connected to all the others
* map out multiple course prerequisites (a solution exists if the graph is a directed graph with no cycles)
* find the shortest route from one city to another (least cost or shortest path in a weighted graph)