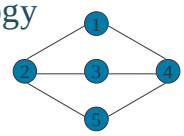
CS122 Algorithms and Data Structures

MW 11:00 am - 12:15 pm, MSEC 101

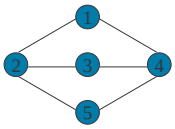
Instructor: Xiao Qin

Lecture 14: Graphs (1)

Graph Terminology

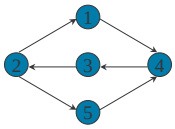


- vertex, node, point
- edge, line, arc 
- G = (V, E)
  - V is set of vertices
  - E is set of edges
- Each edge joins two different vertices



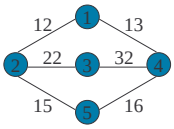
Undirected Graph

- edges do not have a direction
- The edge from 1 to 2 is also an edge from 2 to 1
- Edge (1, 2) implies that there is also an edge (2, 1) [The same edge]



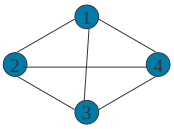
Directed Graph

- edges have a direction
- Edge (2, 1) means only that there is an edge from 2 to 1
- In this example,there is no edge from 1 to 2



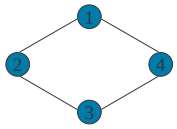
Weighted Graph

- weights (values) are associated with the edges in the graph
- may be directed for undirected
- Weighted graphs are also referred to as networks



Complete Graph

- For each pair of vertices, there is one edge
- If G = (V, E) is a complete graph, and |V| = n, then can you calculate |E|?



Subgraph

- A subgraph G' of graph G = (V, E) is a graph (V', E') that V' $\subseteq$ V and E' $\subseteq$ E.

7

Path

- the sequence of edges (i_1, i_2), (i_2, i_3), …,(i_{k-1}, i_k).
- Denoted as path i_1, i_2, ..., i_k
- Simple path – all vertices (expect possibly first and last) are different
- Length of path is sum of the lengths of the edges

8

Representation of Graphs

- Adjacency matrix
- Incidence matrix
- Adjacency lists: Table, Linked List
- Space/time trade-offs depending on which operation are most frequent as well as properties of the graph

9

Can we use tree traversal algorithms to traverse graphs?

Why?

10

Depth first search

- Starting from vertex v
- Mark v as marked
- Select u as an unmarked node adjacent to v
- If no u, quit
- If u, begin depth first search from u
- When search from u quits, select another node from v
- Similar to preorder tree traversal

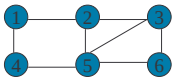
11

Breadth first search

- Starting from node v
- Identify all nodes adjacent to v
- Add these to the set
- Determine set of unvisited nodes which are adjacent to this set
- Add these to the set
- Continue until no new nodes are encountered

12

An Example



What would the visit orders for DFS(1), DFS(5), BFS(1), BFS(5) look like?

13