Graphs

Algorithmic Thinking

Luay Nakhleh

Department of Computer Science

Rice University

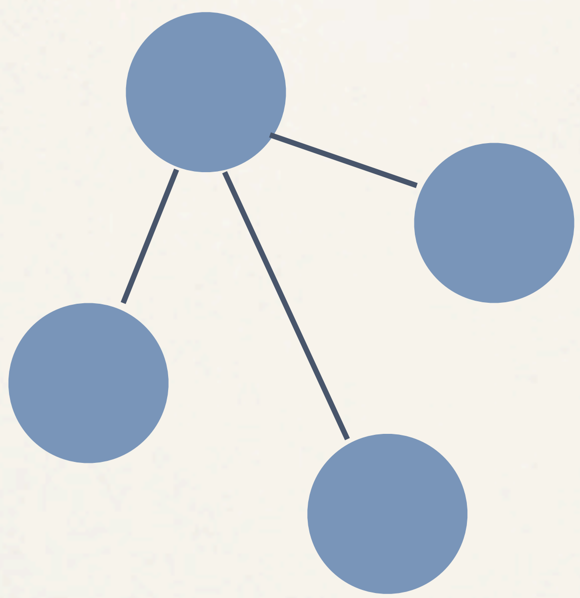
Why Graphs?

- Biological networks
- Maps
- Social networks
- ...

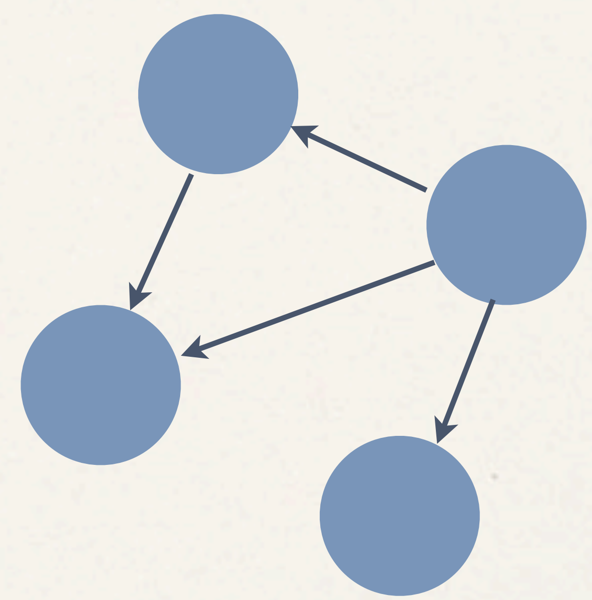
Graphs:
Basic Definitions

The Connectivity of Networks

- Common to all networks we saw is that the <u>connectivity</u> of each of them can be represented via some form of a graph

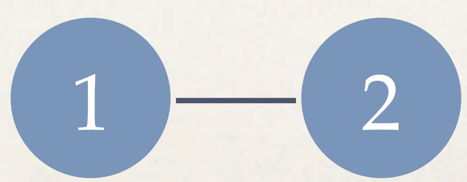


Undirected graph



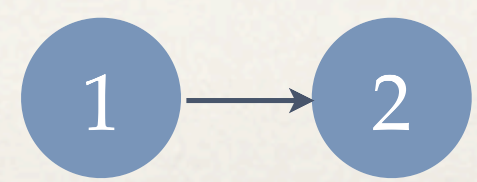
Directed graph

: node
(or, vertex)



: edge

{1,2}



: directed edge

(1,2)

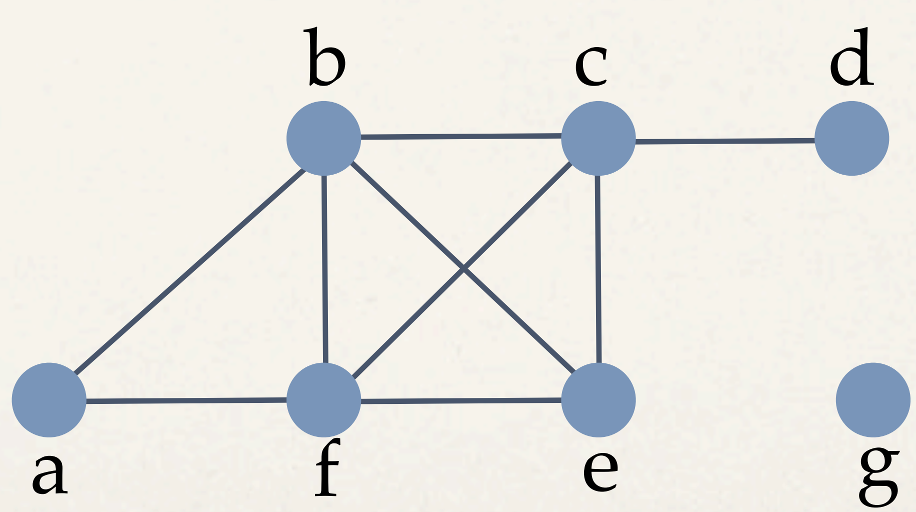
Graphs

- An **undirected graph**, or **graph** for short, G, is a pair (V,E), where
  - V={0,1,...,n-1} is a nonempty set of nodes, and
  - $E \subseteq \{\{i,j\}: i,j \in V\}$ is a set of unordered pairs, each of which corresponds to an undirected edge in the graph G.
- A **directed graph**, or **digraph** for short, G, is a pair (V,E), where
  - V={0,1,...,n-1} is a nonempty set of nodes, and
  - $E \subseteq (V \times V)$ is a set of ordered pairs, each of which corresponds to a directed edge in the graph G.

IMPORTANT: In this course, graphs have no self-loops or parallel edges, unless explicitly stated otherwise.

Basic Terminology

- Two nodes i and j in a graph G=(V,E) are called adjacent (or neighbors) in G if there is an edge between i and j; that is, if $\{i,j\} \in E$.
- The degree of a node in an undirected graph is the number of edges incident with it. The degree of node i is denoted by deg(i).

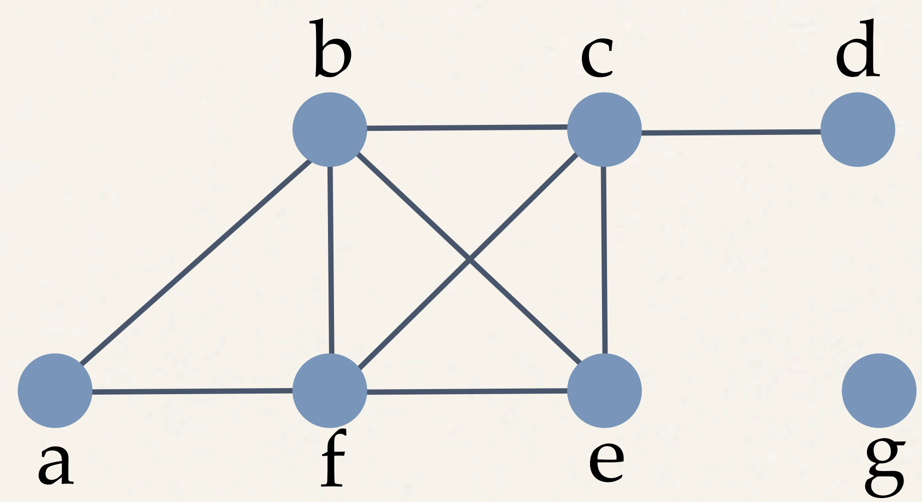


What are the degrees of the nodes?

Degree Distribution

- Define p_k to be the fraction of nodes in the graph that have degree k.
- The degree distribution of a graph can be visualized by making a histogram of the p_k values.

Degree Distribution



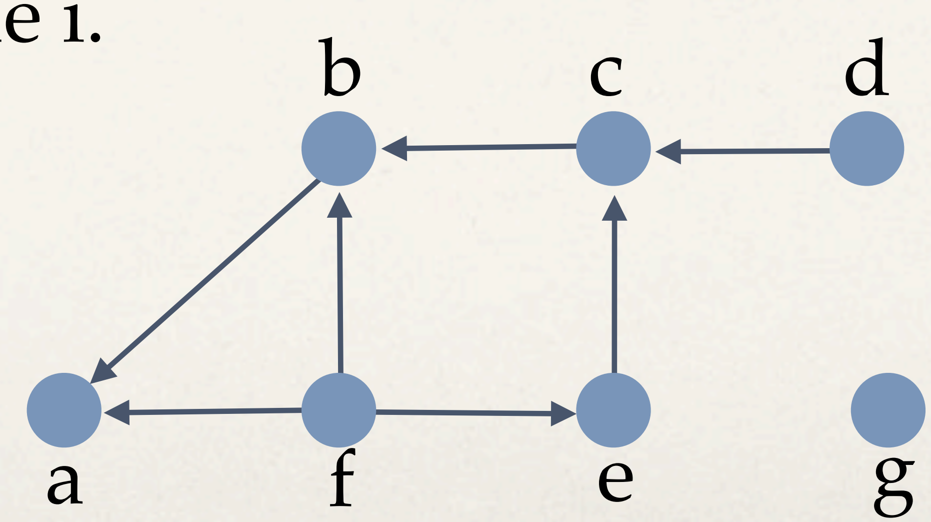
$p_0=1/7$ $p_1=1/7$ $p_2=1/7$
$p_3=1/7$ $p_4=3/7$

Notice that if m is the highest node degree, then:

$$\sum_{k=0}^{m} p_k = 1$$

Basic Terminology

- If e=(i,j) is a directed edge from node i to node j, we say that i is the <u>tail</u> (or, initial node) of e and j is the <u>head</u> (or, terminal node) of e.
- The <u>in-degree</u> of a node i in a directed graph is the number of edges whose head is the node i. The in-degree of node i is denoted by indeg(i).
- The <u>out-degree</u> of i, denoted by outdeg(i), is the number of edges whose tail is the node i.



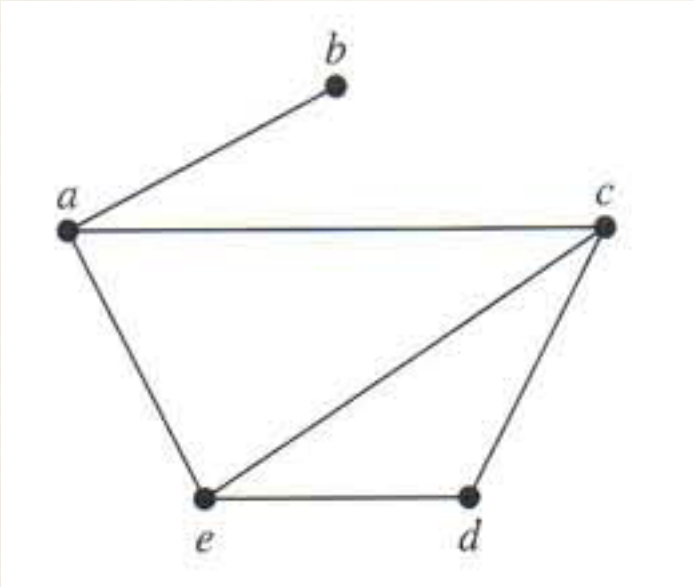
What are the in- and out-degrees of the nodes?

In- and Out-Degree Distributions

- For in-degree distribution: Define p_k to be the fraction of nodes in the graph that have in-degree k.
- For out-degree distribution: Define q_k to be the fraction of nodes in the graph that have out-degree k.
- The in- and out-degree distributions of a graph can be visualized by making a histogram of the p_k and q_k values, respectively.

Graph Representation: Adjacency Lists

- Adjacency lists specify the nodes that are adjacent to each node of the graph.

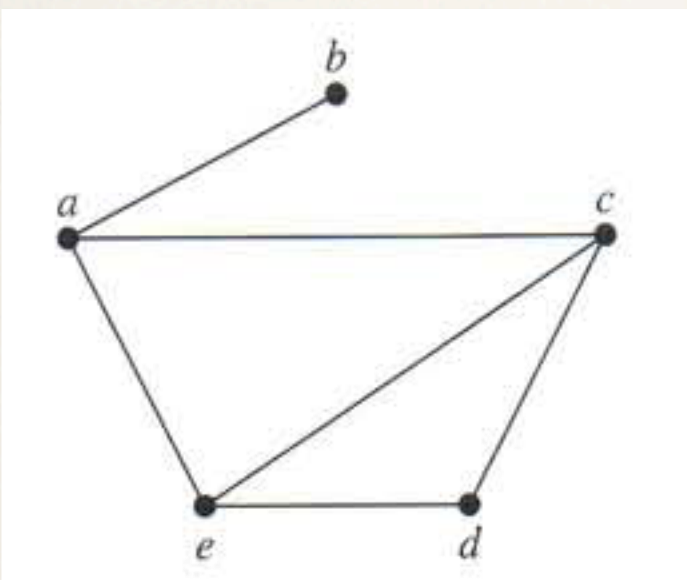


Node	Adjacent nodes
a	b, c, e
b	a
c	a, d, e
d	c, e
e	a, c, d

While "adjacency list" is a historical name, the adjacent nodes of a given node form a set; so, you can think of this representation as an "adjacency set."

Graph Representation: Adjacency Lists

- Adjacency lists specify the nodes that are adjacent to each node of the graph.



Node	Adjacent nodes
a	b, c, e
b	a
c	a, d, e
d	c, e
e	a, c, d

Notice the redundancy!
(ensures symmetry)
In the case of digraphs,
there is no redundancy.

While "adjacency list" is a historical name, the adjacent nodes of a given node form a set; so, you can think of this representation as an "adjacency set."

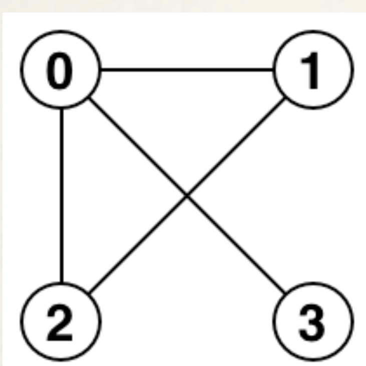
Adjacency Lists and Their Dictionary Representation

Node	Adjacent nodes
a	b, c, e
b	a
c	a, d, e
d	c, e
e	a, c, d

{a: set([b,c,e]),
 b: set([a]),
 c: set([a,d,e]),
 d: set([c,e]),
 e: set([a,c,d])}

Graph Representation: Adjacency Matrices

- Let G=(V,E) be a graph with V={0,1,...,n-1}
- The <u>adjacency matrix</u> of G, denoted by A_G, is the n×n 0-1 matrix with 1 as its $(i,j)^{th}$ entry when i and j are adjacent, and 0 as its $(i,j)^{th}$ entry when i and j are not adjacent.



G

$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

A_G

Notice the redundancy!
In the case of digraphs, A_G is not necessarily symmetric.

Draw a graph whose adjacency matrix is

$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Trade-offs Between Adjacency Lists and Adjacency Matrices

- When the graph is <u>sparse</u>, i.e., contains relatively few edges (relative to what?), it is usually preferable to use adjacency lists (Why?)
- When the graph is <u>dense</u>, i.e., contains relatively many edges (again, relative to what?), it is usually preferable to use adjacency matrices (Why?)

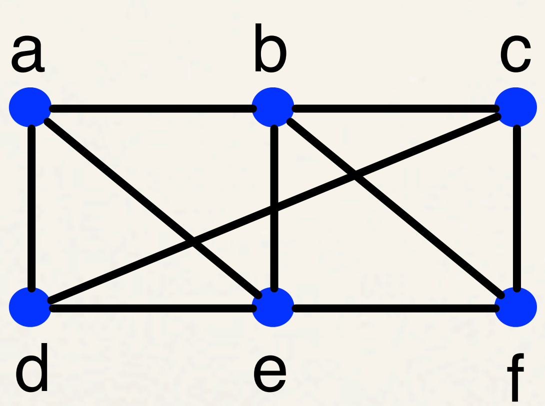
Graph Connectivity: Paths

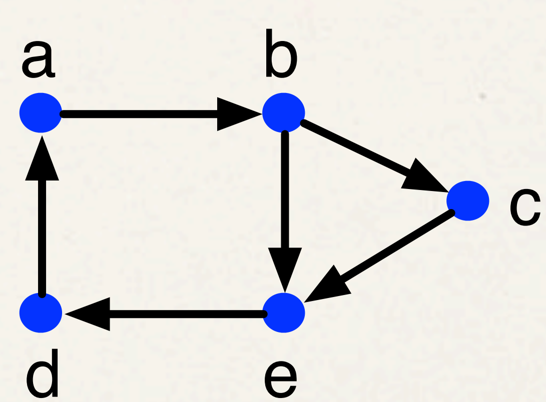
- Let k be a nonnegative integer and G a graph.
    - A <u>path of length k</u> from node v_0 to node v_k in G is a sequence of k edges $e_1,e_2,\ldots,e_k$ of G such that $e_1=\{v_0,v_1\}$, $e_2=\{v_1,v_2\}$, …, $e_k=\{v_{k-1},v_k\}$, where $v_0,\ldots,v_k$ are all nodes in V, and $e_1,\ldots,e_k$ are all edges in E.
- We usually denote such a path by its node sequence $(v_0,v_1,\ldots,v_k)$.
- A path is <u>simple</u> if it does not contain the same node more than once.
- A <u>cycle</u> is a simple path that begins and ends at the same node.
- A path (not necessarily simple) that begins and ends at the same node is called a <u>circuit</u>.

Graph Connectivity: Paths

- If G is a directed graph, a path must traverse edges in their respective directions.

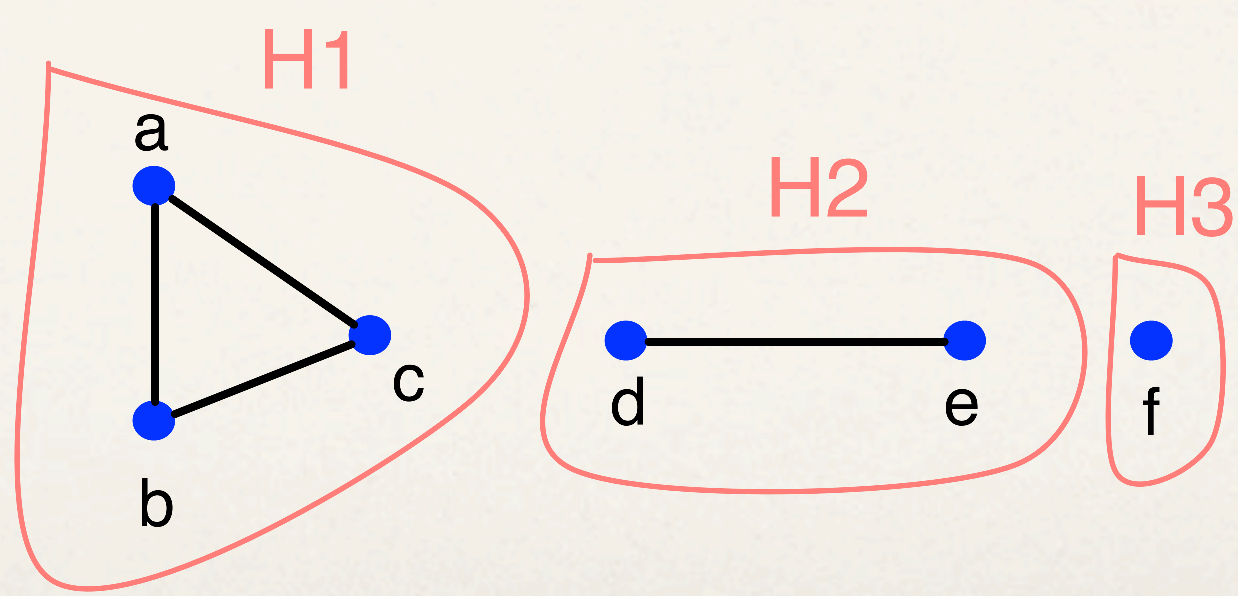
Show a simple path from node e to node d in each of the following two graphs:





Graph Connectivity

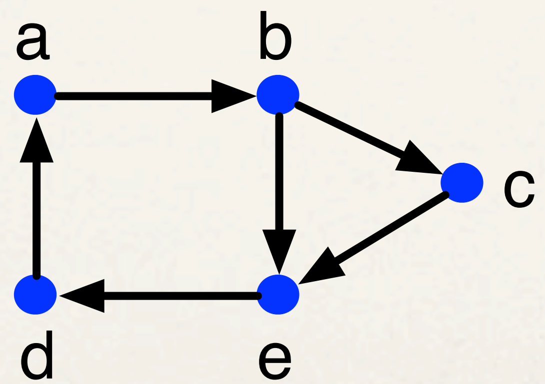
- An undirected graph is called connected if there is a path between every pair of distinct nodes of the graph.
- A connected component (CC) of a graph G is a connected subgraph of G that is not a proper subgraph of another connected subgraph of G.

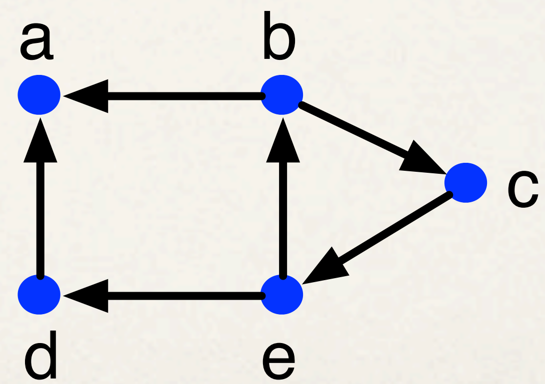


A graph with 3 CCs

Graph Connectivity

- A digraph is strongly connected if there is a path from i to j for every pair of nodes i and j of the digraph (note that there must be a path from i to j and another from j to i).
- A digraph is weakly connected if there is a path between every two nodes in the underlying undirected graph.
- The subgraphs of a directed graph G that are strongly connected but not contained in larger strongly connected subgraphs are called the strongly connected components (SCC) of G.





Strongly connected? Weakly connected? What are the SCCs?