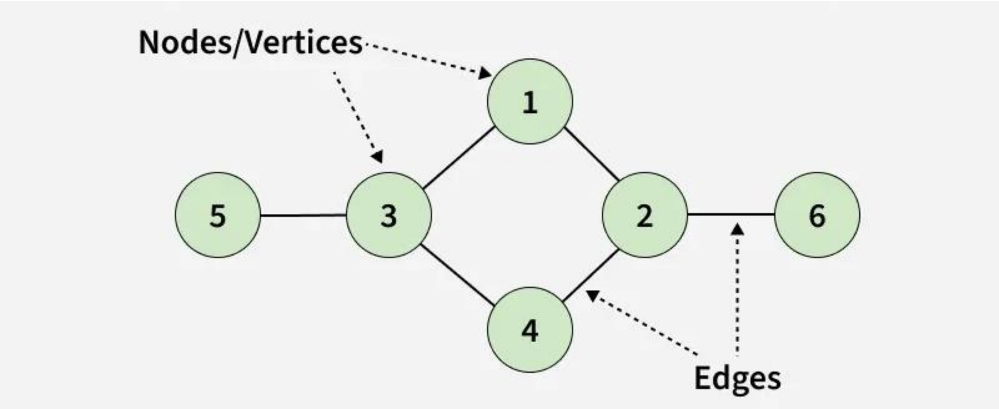
Lecture 6

Graphs I



Announcements

- Homework 3 due Friday night
  - OH today from 4 to 6
- Reflections on Homework 2 due Sunday night
  - **New question**: Did you use AI to assist with this assignment? If so, how?
- Group Meetings start this week
  - Self-scheduled meeting for an hour studying, working on HW, completing practice exercises
- Start thinking about Individual Project 1
  - Due 2/27
  - Project guide and instructions posted

Motivation

Questions:

- What is the shortest driving route between South Hadley and Boston?
- How can we identify fraud in financial transactions?
- What makes someone an influencer on social media?

Motivation

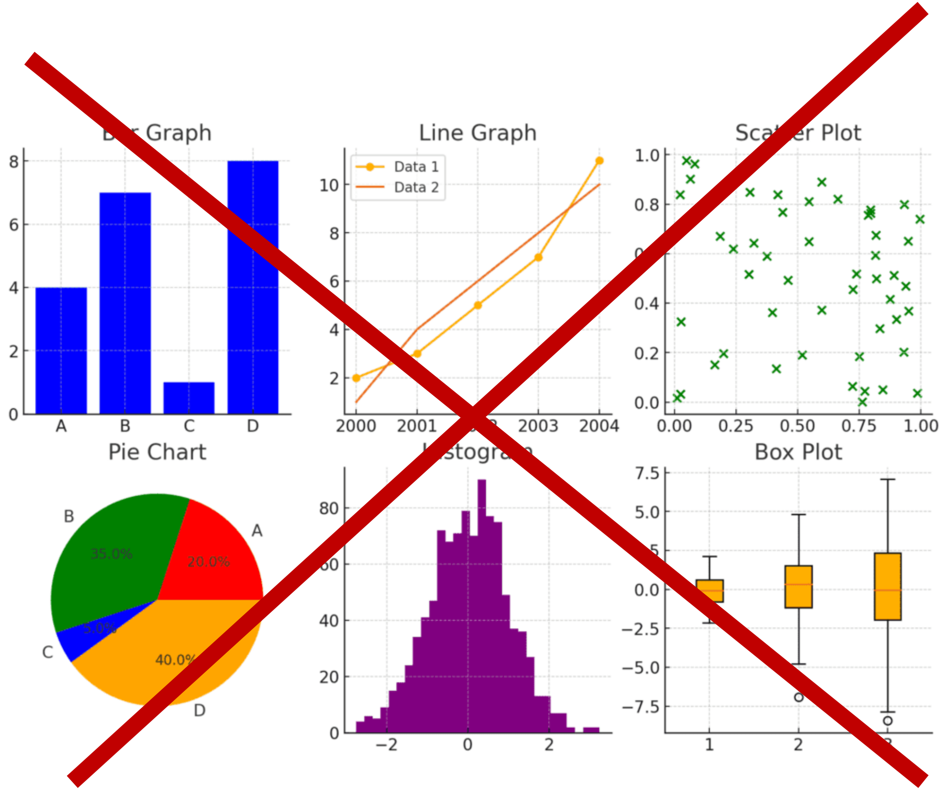
Questions:

- What is the shortest driving route between South Hadley and Boston?
- How can we identify fraud in financial transactions?
- What makes someone an influencer on social media?

Graphs and graph algorithms

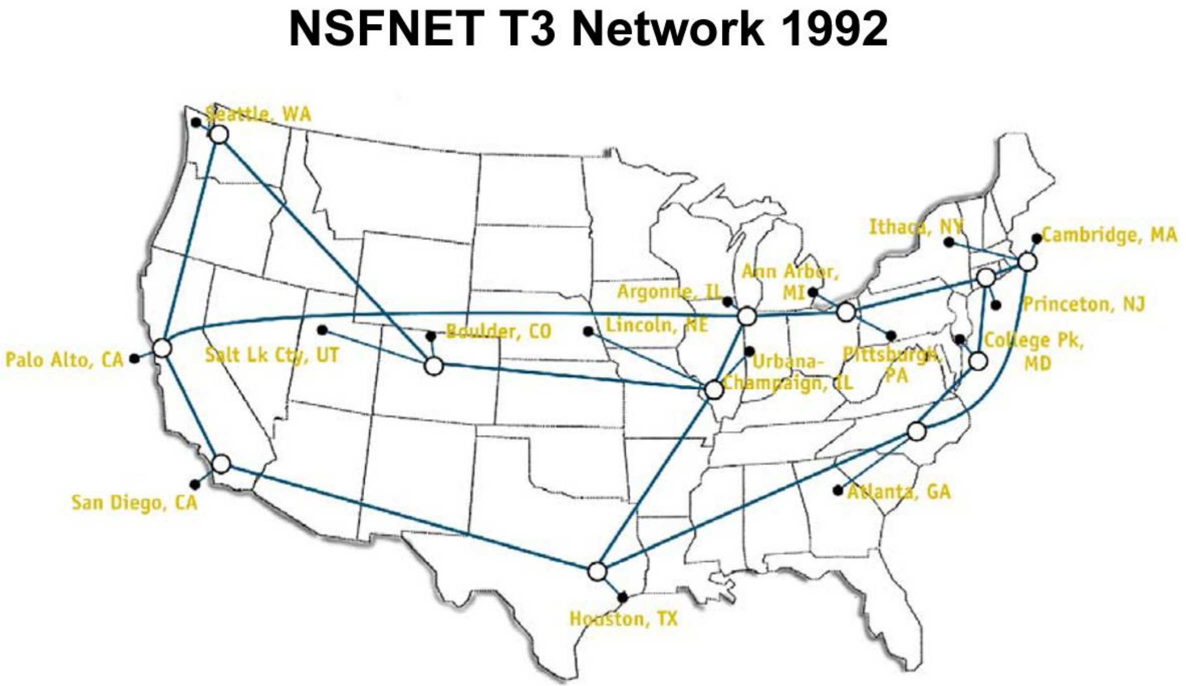
- Represent these problems using the language of graphs (or networks)
- Solve them using graph algorithms

Not These

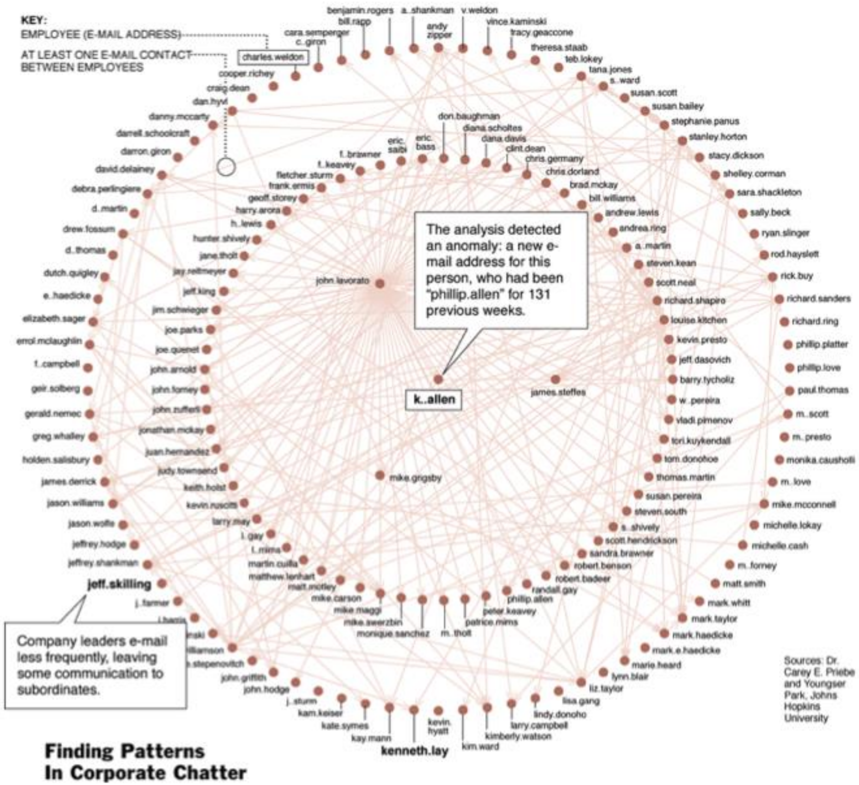


Bar Graph
Line Graph
Data 1
Data 2
Scatter Plot
Pie Chart
35.0%
20.0%
5.0%
40.0%
A
B
C
D
Histogram
Box Plot

The earlish internet



One week of Enron emails



Representations

graph	node	edge
communication	telephone, computer	fiber optic cable
circuit	gate, register, processor	wire
mechanical	joint	rod, beam, spring
financial	stock, currency	transactions
transportation	street intersection, airport	highway, airway route
internet	class C network	connection
game	board position	legal move
social relationship	person, actor	friendship, movie cast
neural network	neuron	synapse
protein network	protein	protein-protein interaction
molecule	atom	bond

Graphs

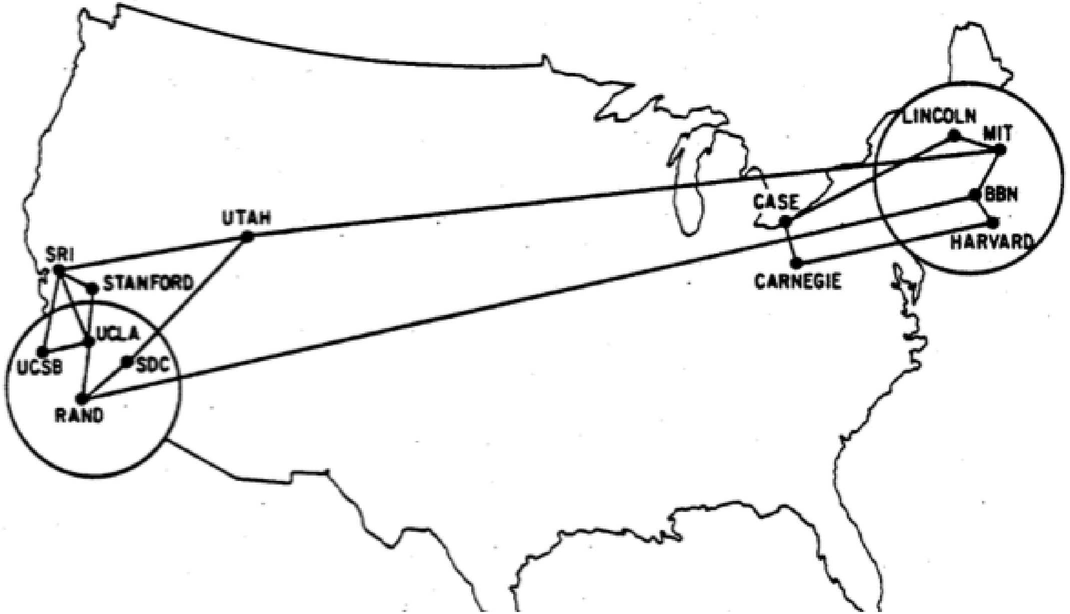
A graph is a mathematical representation of a network

- Set of nodes (vertices) V
- Set of edges (pairs of vertices) E

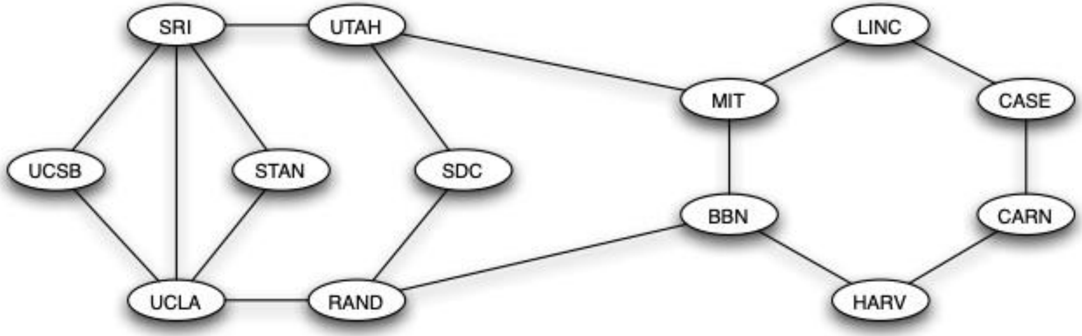
Graph $G = (V, E)$

- G has $n = |V|$ vertices and $m = |E|$ edges

Example: Internet in 1970



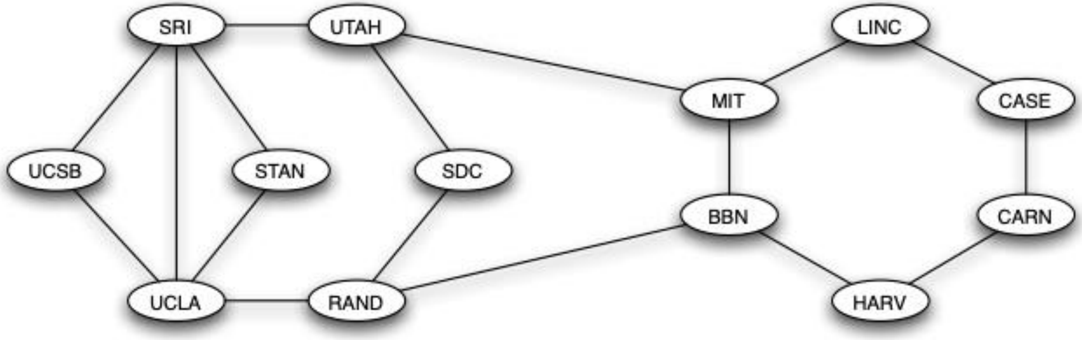
Example: Internet in 1970



Definitions:

- An **edge** $e = \{u, v\}$ is a set of vertices
- Usually written $e = (u, v)$
- We say that u, v are neighbors, are adjacent, are the endpoints of e
- …and edge e is incident to u and v

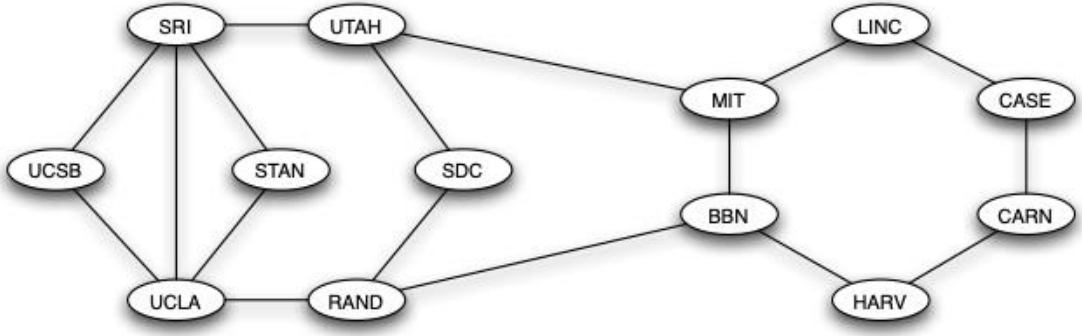
Example: Internet in 1970



Definitions:

- A **path** is a sequence $P = v_1, v_2, \ldots, v_k$ such that each consecutive pair v_i, v_{i+1} are joined by an edge in G
- We call P a path from v_1 to v_k or a $v_1 - v_k$ path

Exercise I



Q: Which of the following is not a path?

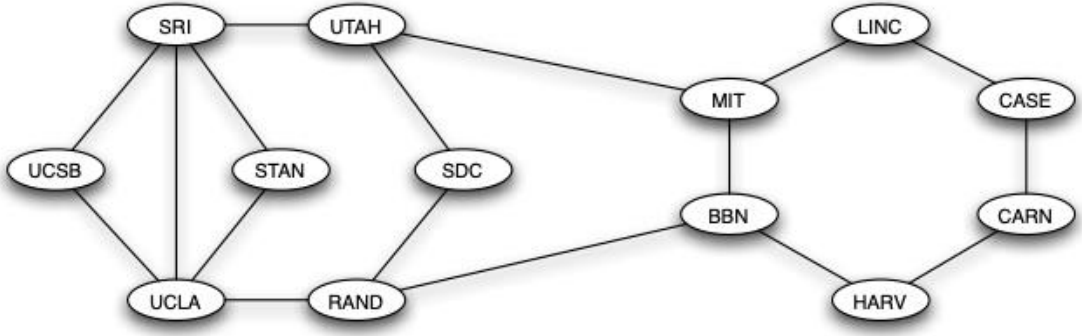
a) UCSB – SRI – UTAH

b) LINC – MIT – LINC – CASE

c) UCSB – SRI – STAN – UCLA – UCSB

d) All are paths

Exercise I



Q: Which of the following is not a path?

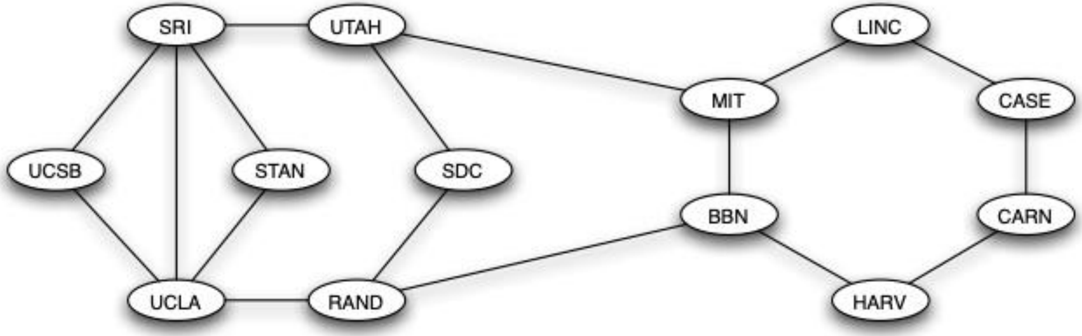
a) UCSB – SRI – UTAH

b) LINC – MIT – LINC – CASE

c) UCSB – SRI – STAN – UCLA – UCSB

d) All are paths

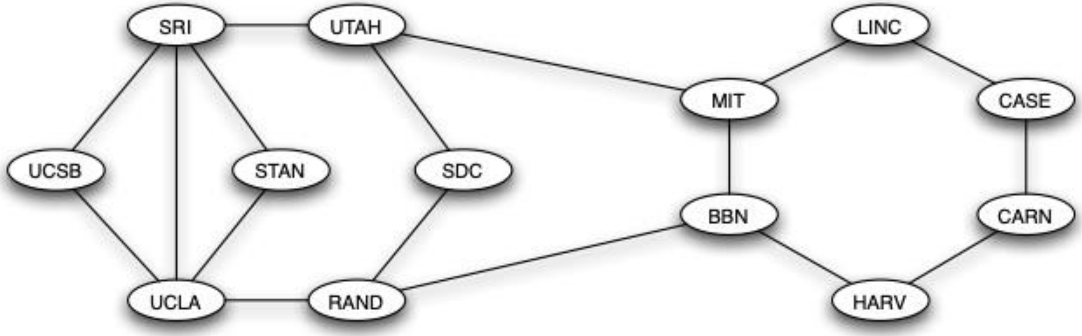
Example: Internet in 1970



Definitions:

❖ A **simple path** is a path where all vertices are distinct

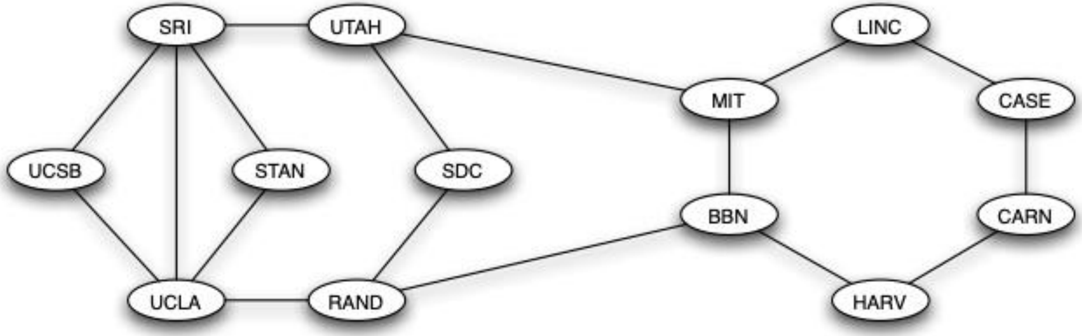
Example: Internet in 1970



Definitions:

- A **simple path** is a path where all vertices are distinct
- A **cycle** is a path $\mathrm{C} = v_1, v_2, \ldots, v_{k-1}, v_k$ where
  - $v_1 = v_k$
  - First $k - 1$ vertices are distinct
  - All edges are distinct

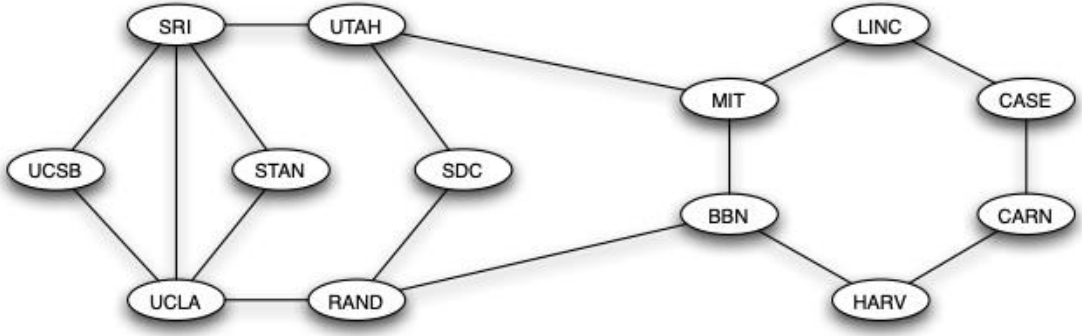
Example: Internet in 1970



Definitions:

- A **simple path** is a path where all vertices are distinct
- A **cycle** is a path $C = v_1, v_2, \dots, v_{k-1}, v_k$ where
  - $v_1 = v_k$
  - First $k-1$ vertices are distinct
  - All edges are distinct
- The **distance** from u to v is the minimum number of edges in a $u-v$ path

Exercise II



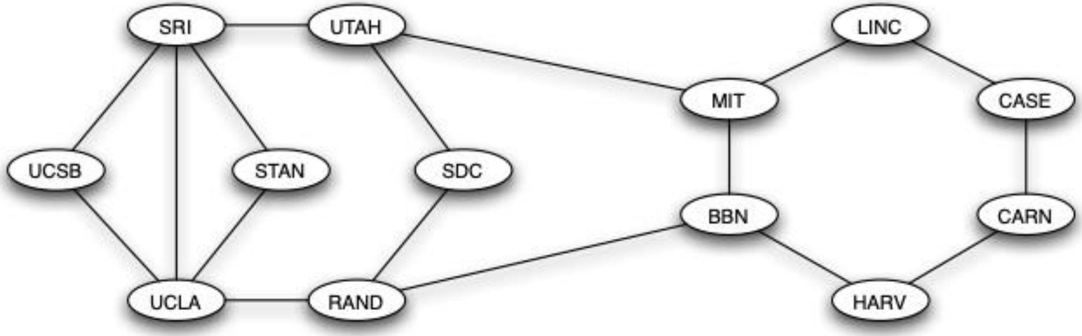
Q: What can we say about each of the following paths?

a) UCSB – SRI – UTAH

b) LINC – MIT – LINC – CASE

c) UCSB – SRI – STAN – UCLA – UCSB

Exercise II



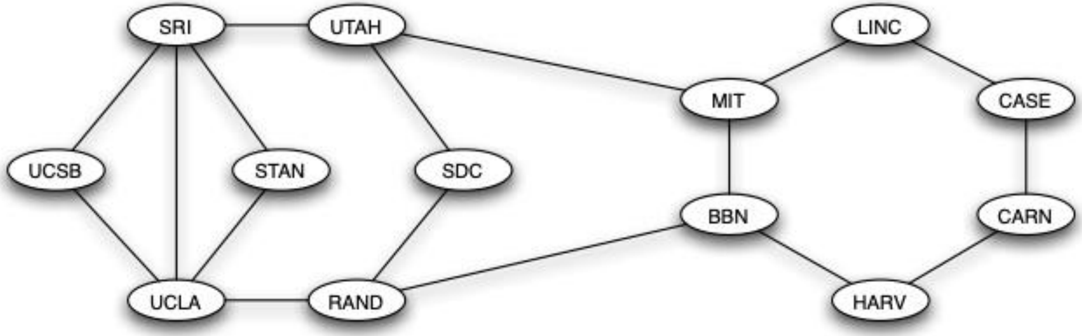
Q: What can we say about each of the following paths?

a) UCSB – SRI – UTAH   Simple Path

b) LINC – MIT – LINC – CASE   Path but not a Simple Path

c) UCSB – SRI – STAN – UCLA – UCSB   Cycle

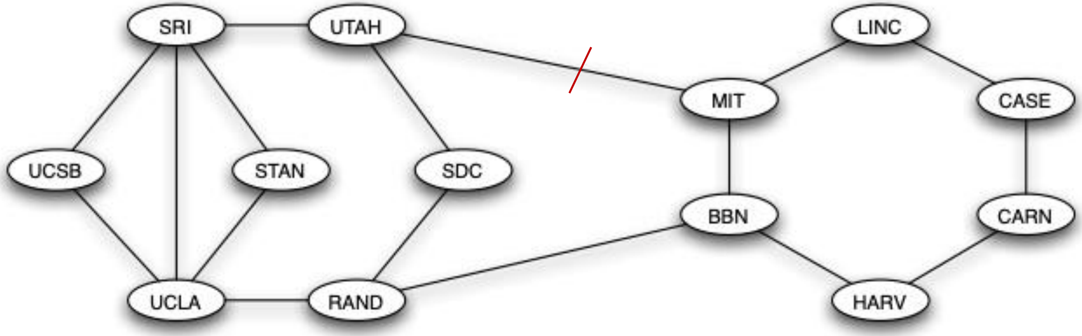
Example: Internet in 1970



Definitions:

- A **connected graph** is a graph with paths between every pair of vertices

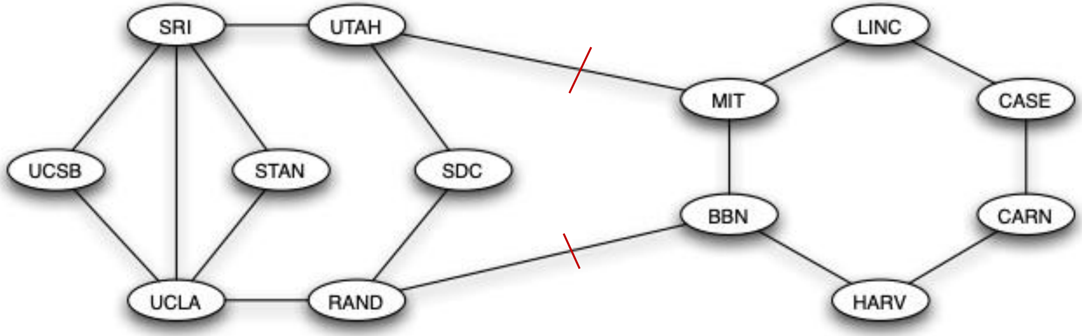
Example: Internet in 1970



Definitions:

- A **connected graph** is a graph with paths between every pair of vertices
- Q: Is still graph still connected?

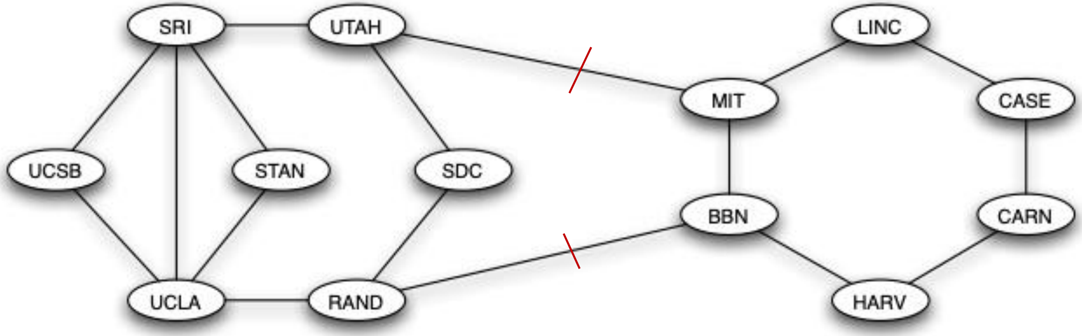
Example: Internet in 1970



Definitions:

- A **connected graph** is a graph with paths between every pair of vertices
- Q: Is still graph still connected? Yes, but how about now?

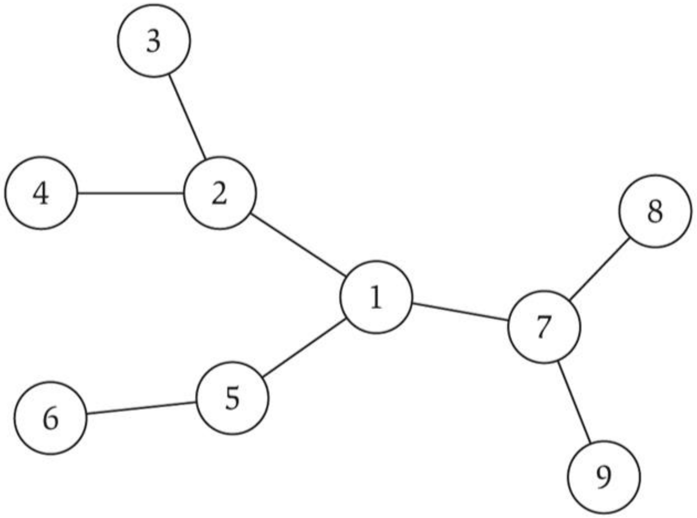
Example: Internet in 1970



Definitions:

- A **connected graph** is a graph with paths between every pair of vertices
- Q: Is still graph still connected? Yes, but how about now?
- A **connected component** is a maximal subset of vertices such that a path exists between every pair in the subset
- **Maximal** means that if a new vertex is added then there will no longer be a path between every pair

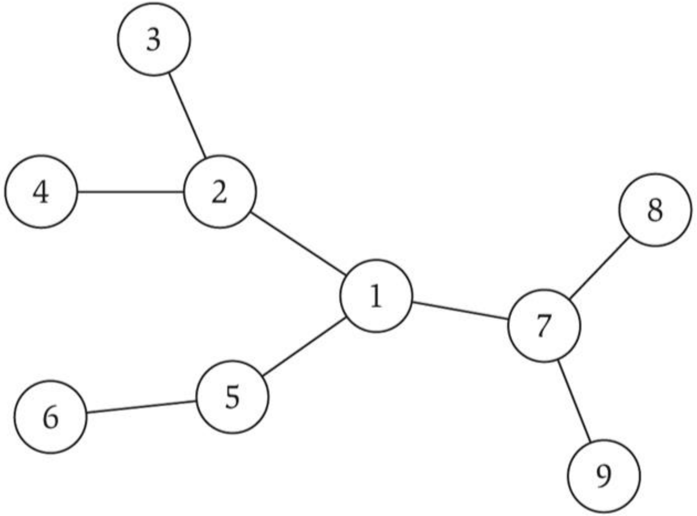
Trees



Definitions:

❖ A **tree** is a connected graph with no cycles

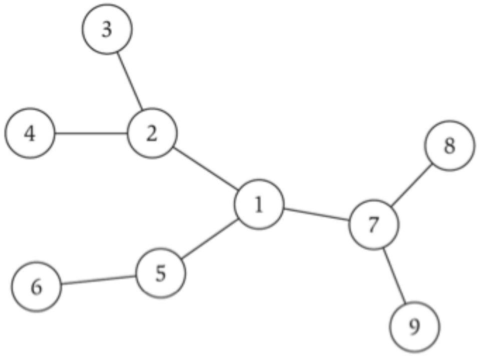
Trees



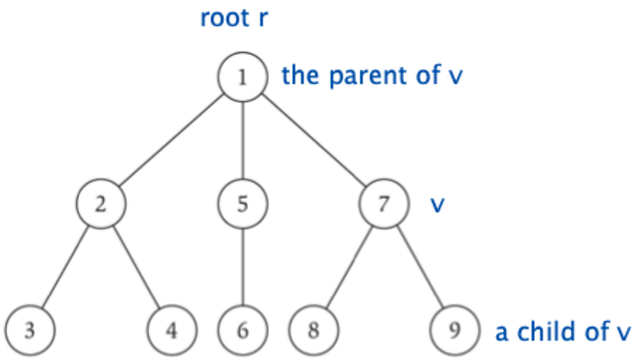
Definitions:

- A **tree** is a connected graph with no cycles
- A **rooted tree** is a tree with a parent-child relationship
  - Pick a root r and "orient" all edges away from the root
  - Parent of v means predecessor on path from r to v

Trees



a tree

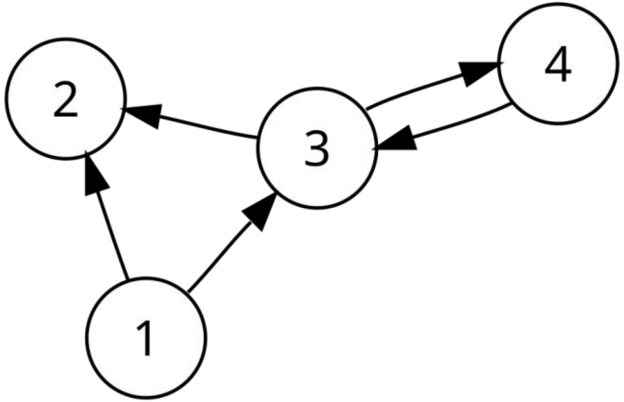


the same tree, rooted at 1

Definitions:

- A **tree** is a connected graph with no cycles
- A **rooted tree** is a tree with a parent-child relationship
  - Pick a root r and "orient" all edges away from the root
  - Parent of v means predecessor on path from r to v

Directed Graphs



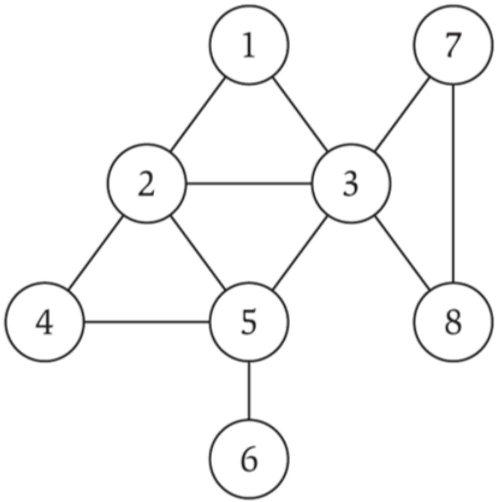
Graphs can be **directed**

- Edges point from one vertex to another
- Encode an asymmetric relationship
- Graphs are undirected unless noted so

Graph Traversal

Imagine we're plopped down onto a vertex in a graph

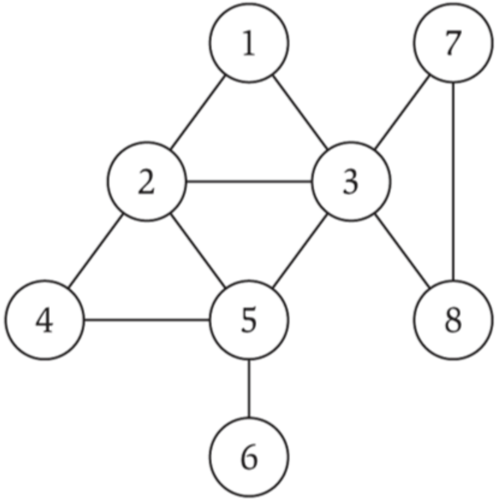
- How/what can we learn about the graph?
- Is it connected?
- If not, how big is the largest connected component?
- Is there a path between 2 and 8?



Graph Traversal

Imagine we're plopped down onto a vertex in a graph

- How/what can we learn about the graph?
- Is it connected?
- If not, how big is the largest connected component?
- Is there a path between 2 and 8?



How can you answer these algorithmically?

- Graph traversal
- Bread-first search (BFS): explore locally
- Depth-first search (DFS): deep dive and backtrack

Next Time

- Dive into BSF and DSF
- Analyze implementations using stacks and queues