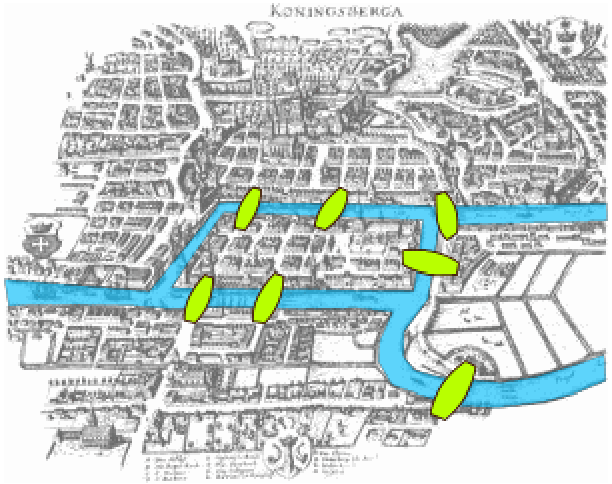
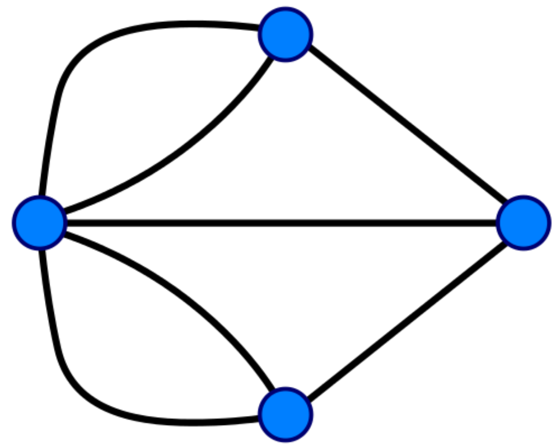
Lecture 14: Introduction to Graphs

The Seven Bridges of Königsberg

Q: Can you find a path to cross all seven bridges, each exactly once?





Q: (Reformulated as a graph problem) Can you find a path in the graph that includes every edge exactly once?

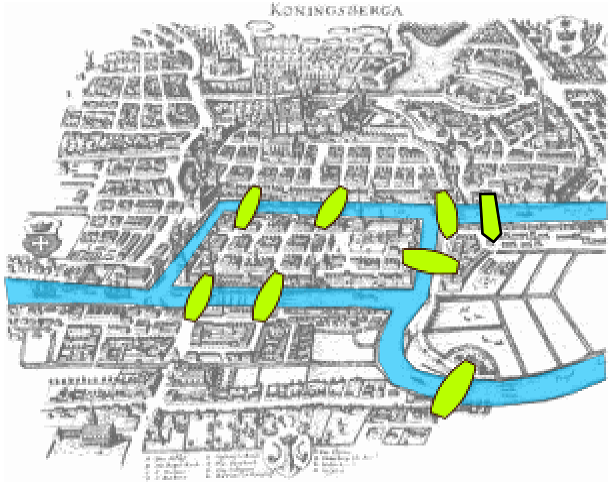
A: Not possible.

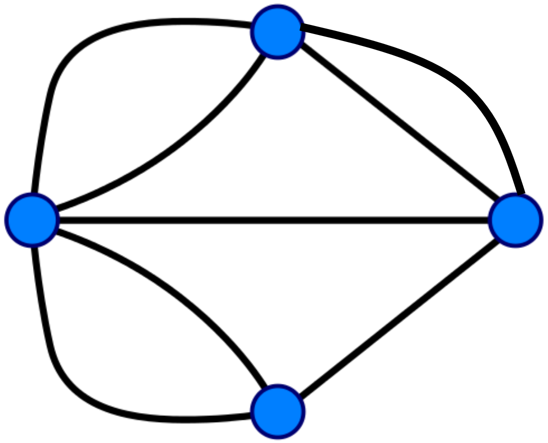
Theorem: A graph has such a path (known as an Euler path) iff there are 0 or 2 vertices with an odd degree.

Q: Can a graph have exactly one vertex with an odd degree?

The Seven Bridges of Königsberg

Solution: Build one more bridge to remove 2 odd-degree vertices.





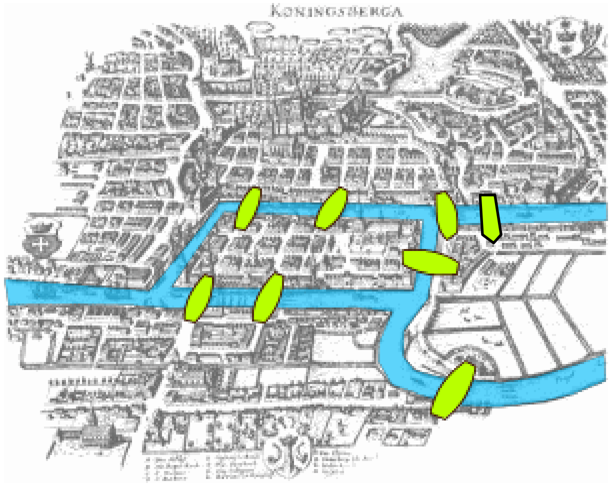
Algorithm:

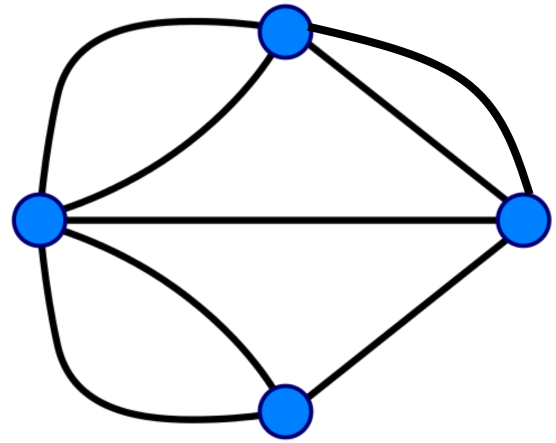
```
u ← any odd-degree vertex
if no such vertex exists
     u ← any vertex
while u has an edge not taken yet
     take that edge (u,v)
     u ← v
```

But, this algorithm may get stuck…

The Seven Bridges of Königsberg

Solution: Build one more bridge to remove 2 odd-degree vertices.





Algorithm:

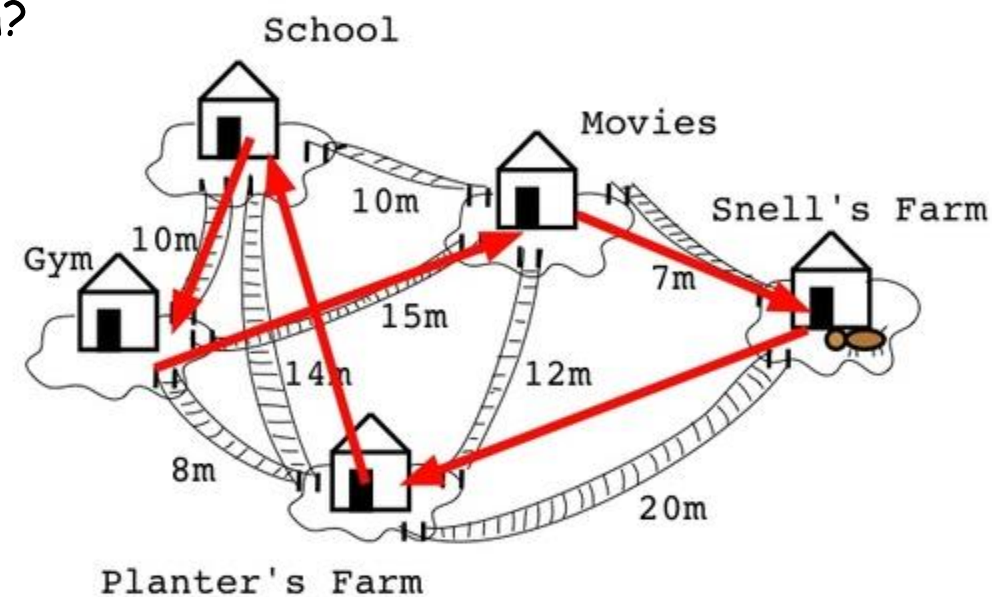
```
while there are still edges not taken yet
     u ← any odd-degree vertex
     if no such vertex exists
          u ← any vertex
     p ← Find-Path(u)
     insert p into existing path at u

Find-Path(u):
while u has an edge not taken yet
     take that edge (u,v)
     u ← v
```

Graph Applications

Graph	Nodes	Edges
transportation	street intersections	highways
communication	computers	fiber optic cables
World Wide Web	web pages	hyperlinks
social	people	relationships
food web	species	predator-prey
software systems	functions	function calls
scheduling	tasks	precedence constraints
circuits	gates	wires

The Traveling Salesman Problem

Q: How to visit all places with the shortest total distance, and come back to origin?



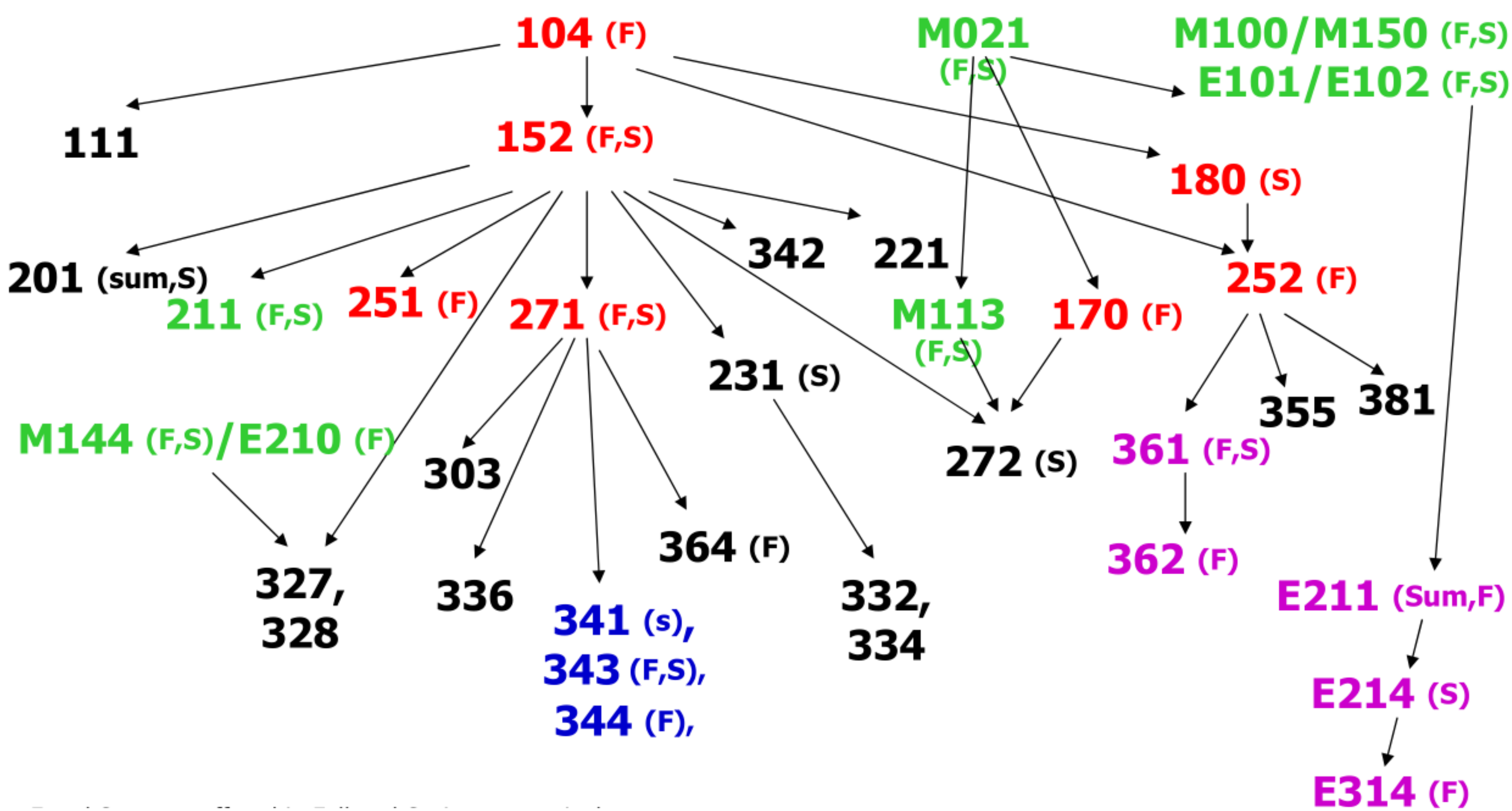
Q: (Reformulated as a graph problem) Given a graph where edges have weights (lengths), how to find a cycle with minimum total weight that includes all vertices?

A: Don't know.

- Don't have an algorithm that runs in polynomial time. (Conjecture is that such an algorithm doesn't exist.)
- This is actually equivalent to the P = NP problem (still open).

Course Dependency Chart

Q: Find an order to take all courses, while respecting all dependencies?

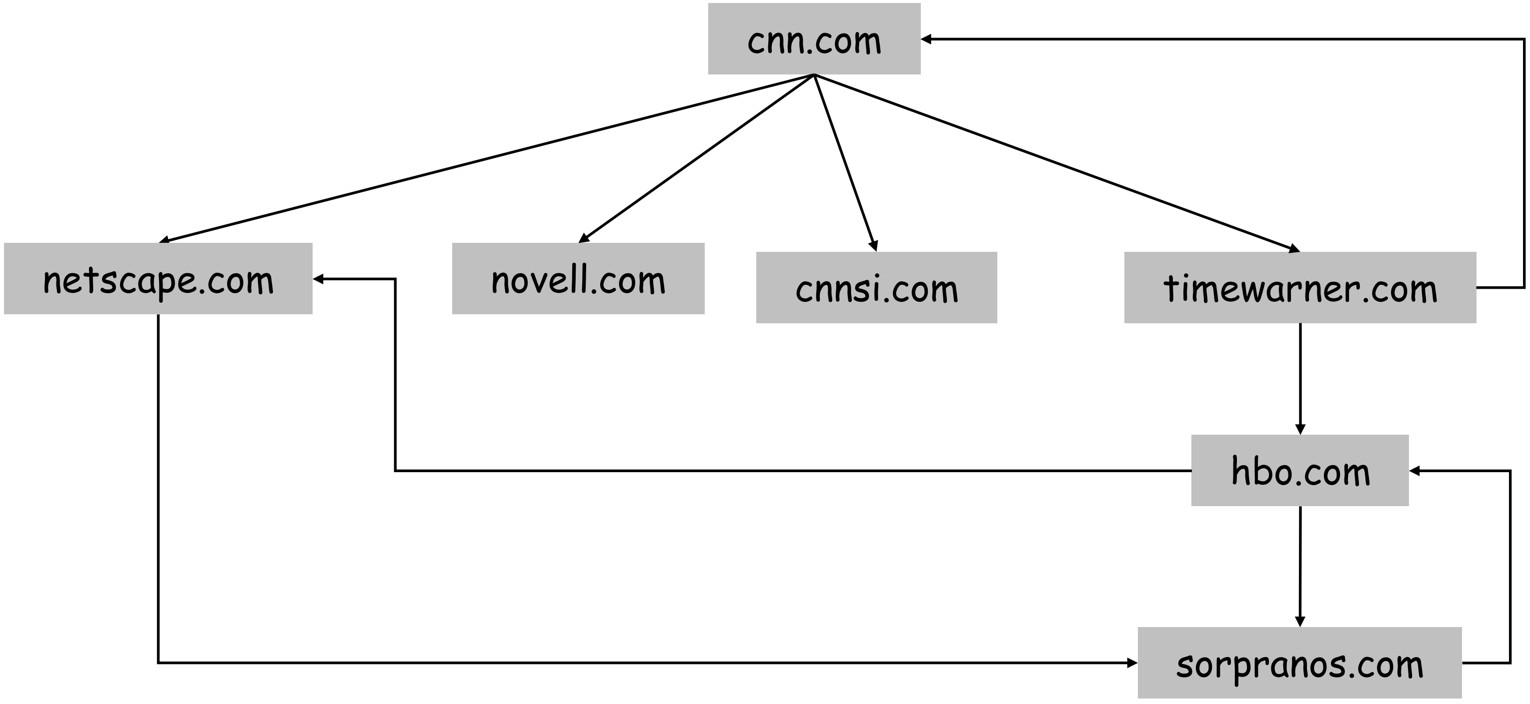


Q: (Reformulated as a graph problem) Given a directed graph, find an ordering of the vertices such that for any edge $u \rightarrow v$, u is ordered before v, or declare that there is a cycle in the graph.

World Wide Web

Web graph.

- Node: web page.
- Edge: hyperlink from one page to another (directed).



Social Networks

Social network graph.

- Nodes: people.
- Edges: relationship between two people
  - Can be directed or undirected

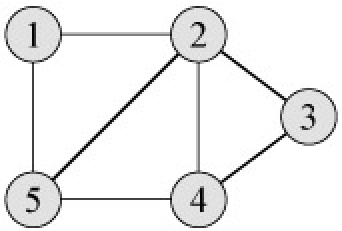
Reference: Valdis Krebs, http://www.firstmonday.org/issues/issue7_4/krebs

Undirected and Directed Graphs

Graph. $G = (V, E)$

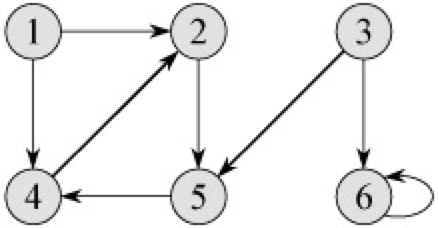
- V: set of nodes (vertices).
- E: set of edges between pairs of nodes.
- Abusing notation, we also use V and E to denote the number of nodes and edges. We sometimes also use $n = |V|$, $m = |E|$.

Undirected graph.



- Edges have no direction (or both directions)
- $\deg(v) =$ # edges at v
- $\sum_{v \in V} \deg(v) = 2E$

Directed graph.



- Edges have directions
- If an edge has both directions, we will use two edges with opposite directions
- $\deg^{out}(v) =$ # edge leaving v; $\deg^{in}(v) =$ # edge entering v.
- $\sum_{v \in V} \deg^{out}(v) = \sum_{v \in V} \deg^{in}(v) = E$

Exercises

Q: Can an undirected graph have exactly one vertex with an odd degree?

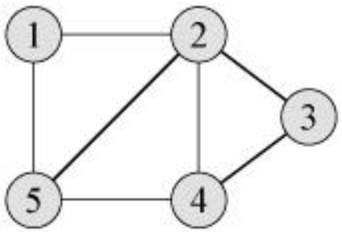
A: No, since $\sum_{v \in V} \deg(v) = 2E$, which is an even number.

Q: [Handshaking lemma] Suppose that the guests in a party shake hands with each other arbitrarily. Show that, no matter how they shake hands, the number of guests who shake hands an odd number of times must be even.

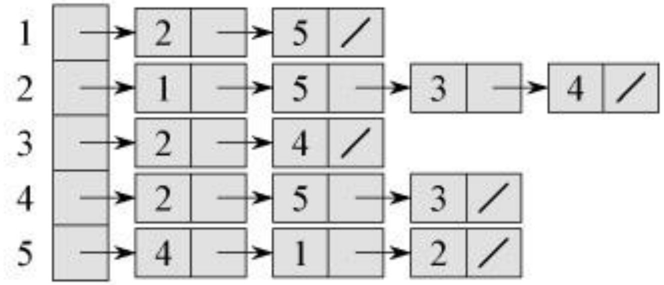
A: Model each guest as a node, a handshake as an edge. This is an undirected graph, so $\sum_{v \in V} \deg(v)$ is even. If an odd number of people shake an odd number of times, then the total degree would be

$$\underbrace{\text{odd} + \text{odd} + \ldots + \text{odd}}_{\text{odd}} + \underbrace{\text{even} + \text{even} + \ldots + \text{even}}_{\text{no matter how many}} = \text{odd}$$

Graph Representation: Adjacency List and Adjacency Matrix



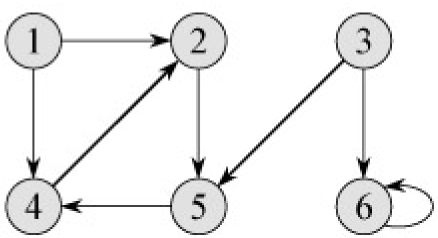
(a)



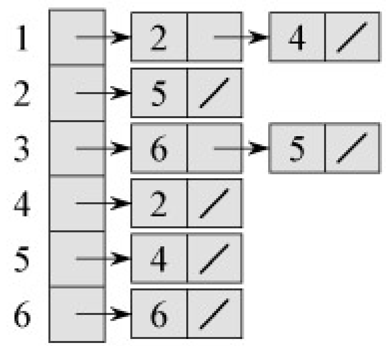
(b)

	1	2	3	4	5
1	0	1	0	0	1
2	1	0	1	1	1
3	0	1	0	1	0
4	0	1	1	0	1
5	1	1	0	1	0

(c)



(a)



(b)

	1	2	3	4	5	6
1	0	1	0	1	0	0
2	0	0	0	0	1	0
3	0	0	0	0	1	1
4	0	1	0	0	0	0
5	0	0	0	1	0	0
6	0	0	0	0	0	1

(c)

Graph Representation: Adjacency List and Adjacency Matrix

Adjacency list.

- A node-indexed array of lists.
- Given node u, retrieving all neighbors in $\Theta(\deg(u))$ time
- Given u, v, checking if (u, v) is an edge takes $\Theta(\deg(u))$ time.
- Space: $\Theta(V + E)$.

Adjacency matrix.

- A $V \times V$ matrix.
- Given node u, retrieving all neighbors in $\Theta(V)$ time
- Given u, v, checking if (u, v) is an edge takes $O(1)$ time.
- Space: $\Theta(V^2)$.

Note:

- Adjacency list is used more often, since most graphs are sparse.
- Usually, assume no self-loops and duplicated edges.
  - Thus, for undirected graphs, $0 \leq E \leq V(V-1)/2$
  - For directed graphs, $0 \leq E \leq V(V-1)$
- Can convert from one to the other in $\Theta(V^2)$ time.

Q: How to represent weights?

Paths and Connectivity

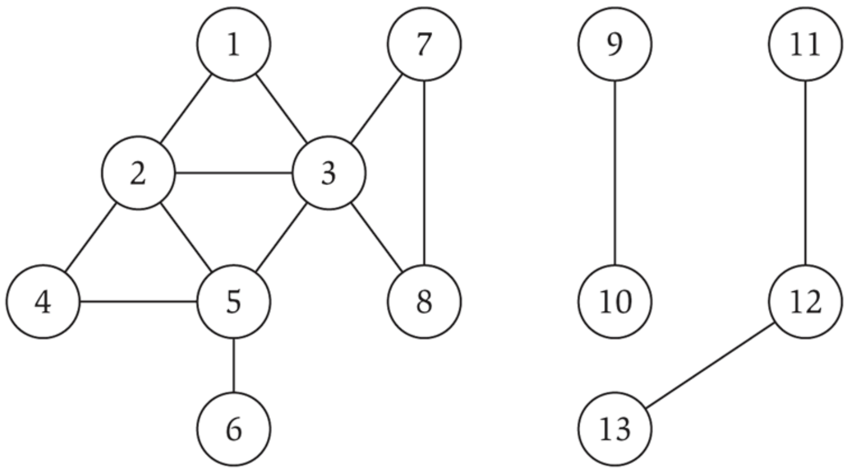
Def. A path in a (directed or undirected) graph $G = (V, E)$ is a sequence P of nodes $v_1, v_2, \ldots, v_{k-1}, v_k$ such that (v_i, v_{i+1}) is an edge. The length of the path is $k - 1$ (i.e., # edges in the path).

Def. A path is simple if all nodes are distinct.

Def. An undirected graph is connected if for every pair of nodes u and v, there is a path between u and v.

Theorem: For a connected graph, $E \geq V - 1$.

Def. A cycle is a path $v_1, v_2, \ldots, v_{k-1}, v_k$ in which $v_1 = v_k$, $k > 2$, and the first $k - 1$ nodes are all distinct.



Exercise

Q: Suppose in a wireless network of n mobile devices, each device is within communication range with at least $n/2$ other devices (assuming n is an even number). Show that all devices are connected.

Reformulated as a graph problem: Let G be an undirected graph where each node has degree $\geq n/2$. Show that G is connected.

Pf: Consider any two nodes u and v in G. There are two cases:

- If there is an edge (u, v), then u and v are connected.
- If there is no direct edge between u and v, then they must have a common neighbor, say w, because
  - There are $n - 2$ nodes other than u and v.
  - u and v each have $\geq n/2$ neighbors among these nodes.
- Thus there is a path between u and v.
- The above argument holds for any two nodes u, v, so the graph G is connected.

Q: If the threshold $n/2$ is changed to $n/2 - 1$, does the claim still hold?

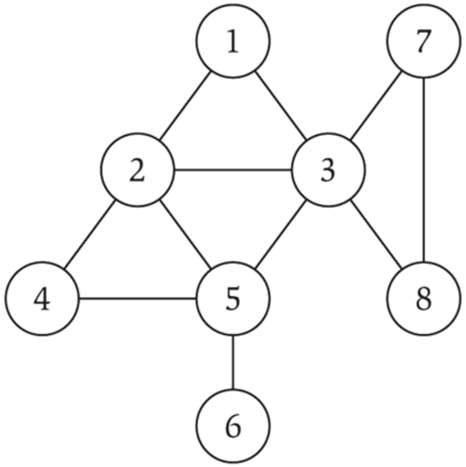
Connectivity and Shortest Path

s-t connectivity problem. Given two nodes s and t, is there a path from s to t?

s-t shortest path problem. Given two node s and t, what is the shortest path from s to t?

Def: The length of the path (in terms of number of edges) is the distance from s to t.

The problem can be defined on either an undirected or directed graph.



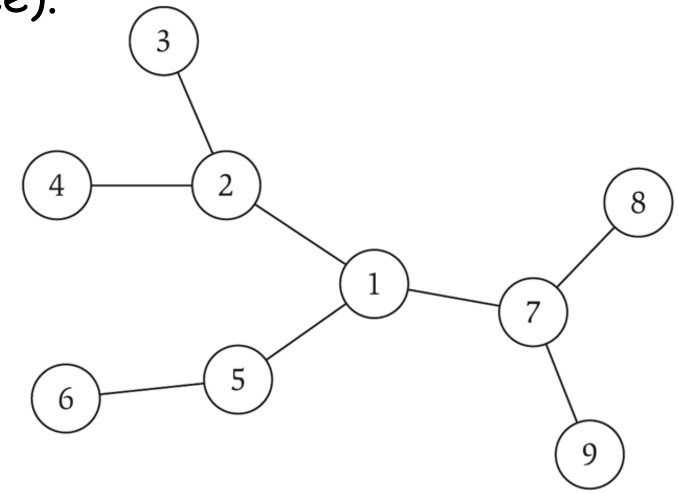
Trees

Def. An undirected graph is a tree if it is connected and does not contain a cycle.

Def. An undirected graph is a forest if it does not contain a cycle (i.e., a collection of trees).

Theorem (simpler version of Theorem B.4 in textbook):
Let G be an undirected graph. Any two of the following statements imply the third (hence G is a tree).

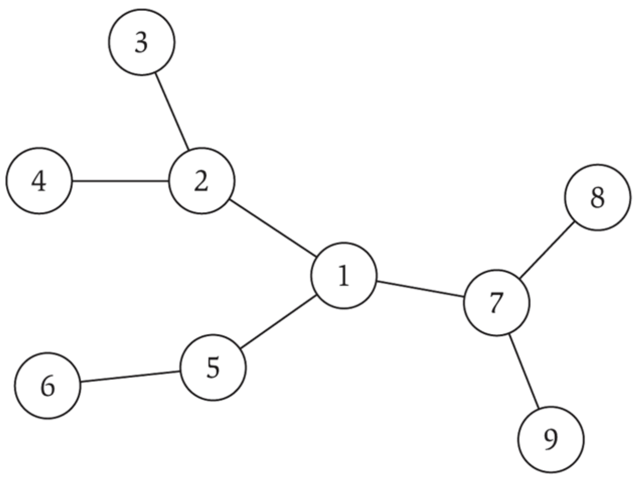
(1) G is connected.
(2) G does not contain a cycle.
(3) $E = V - 1$.



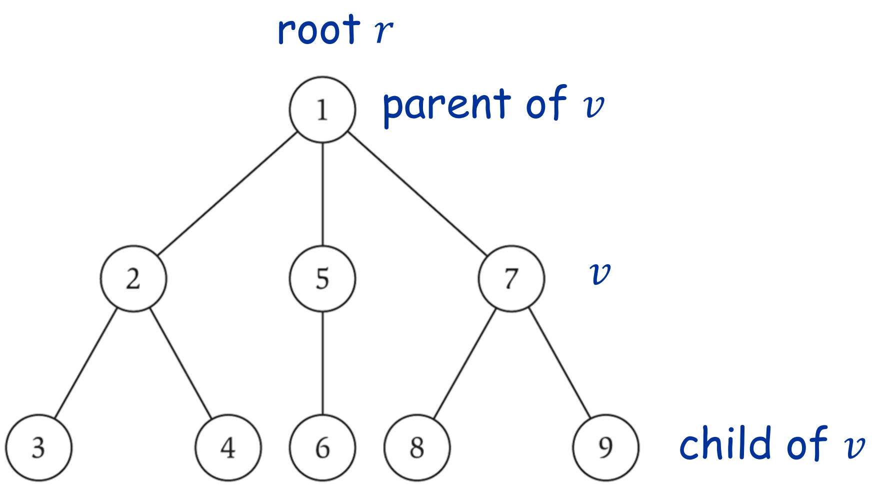
Proof: (Omitted)

Rooted Trees

Rooted tree. Given a tree T, choose a root node r and orient each edge away from r.



a tree



the same tree, rooted at 1