Some Useful Ideas From Graph Theory

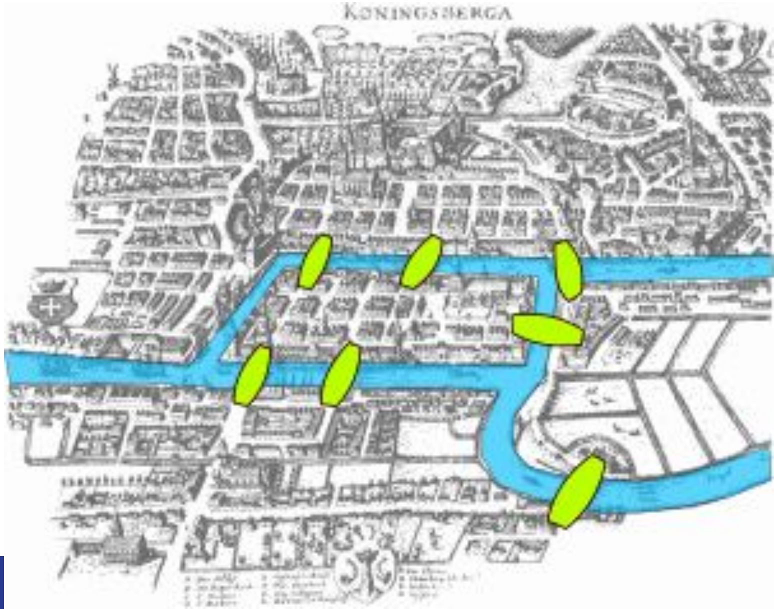
By Grayson York

Graph Theory Origins

- Started by Leonhard Euler in 1736
- First major result was the resolution of the Konigsberg bridge problem
- For a long time, considered completely pure math
- With the arrival of computers, now considered very applied
- Applications include (not nearly exhaustive)
  - Social networks
  - Circuit design
  - Neuroscience
  - Epidemiology
  - Chemical Bonds
  - Pretty much everything in computer science

The Bridges Of Konigsberg

- The city of Konigsberg (modern day Kaliningrad) has 4 islands and 7 bridges
- The people of the city want to go on a walk crossing each bridge exactly once
- They do not care where they start and where they end, but cannot swim and cannot cross the same bridge twice.
- Is this possible?
- We would call such a path an “Euler Trail”



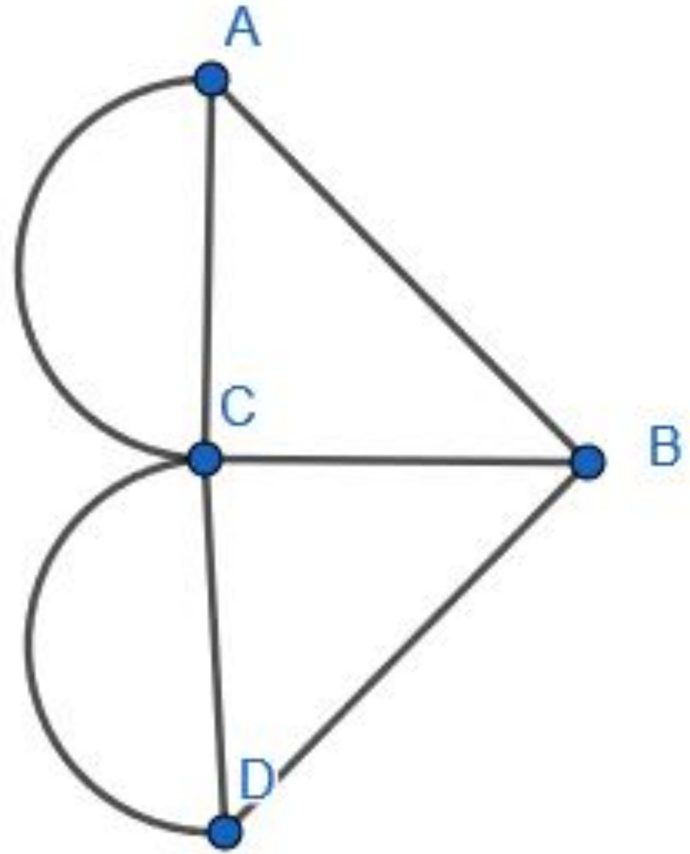
Solution to Konigsberg Bridge Problem

- Represent each island as a node
- Represent each island as a vertex
- 4 vertices have an odd number of edges
- One can be the start, one can be the end
- The other nodes must be left every time

  they are entered

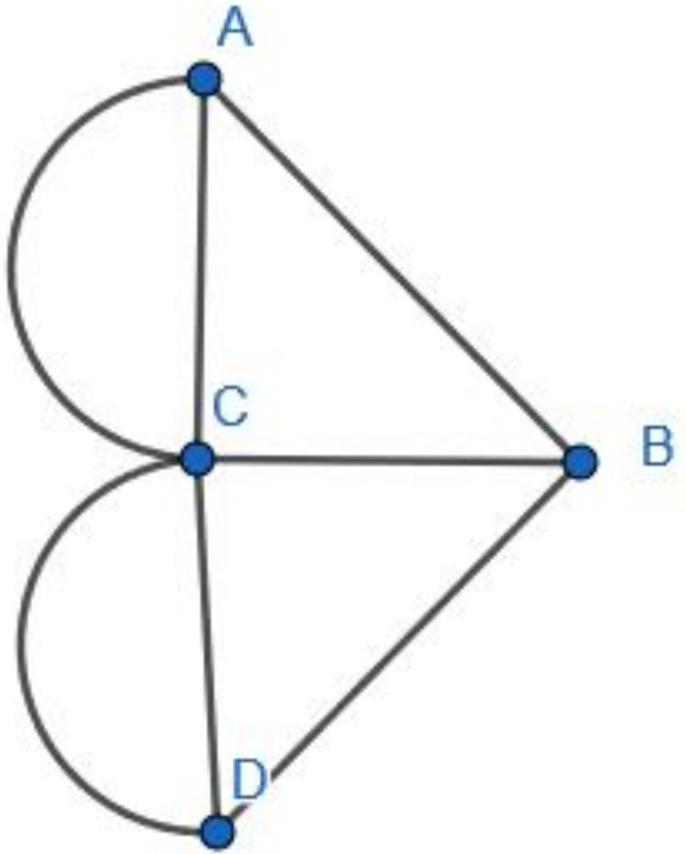
- This is impossible!
- More generally, any graph with more than 2

  Vertices of odd degree has no Euler Trail



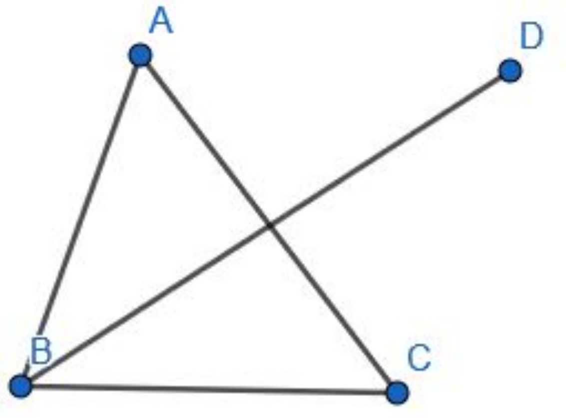
What is a Graph?

- A set of vertices (A,B,C,D)
- A set of edges, which connect the vertices

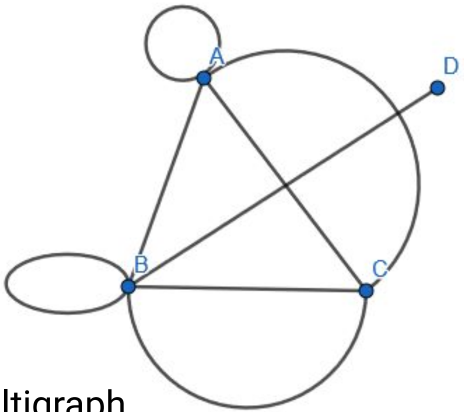


Simple Graphs

- A simple graph has at most one edge connecting a pair of vertices and no self loops
- A multigraph can have parallel edges and self loops
- As a convention, you can assume graphs are simple if unspecified



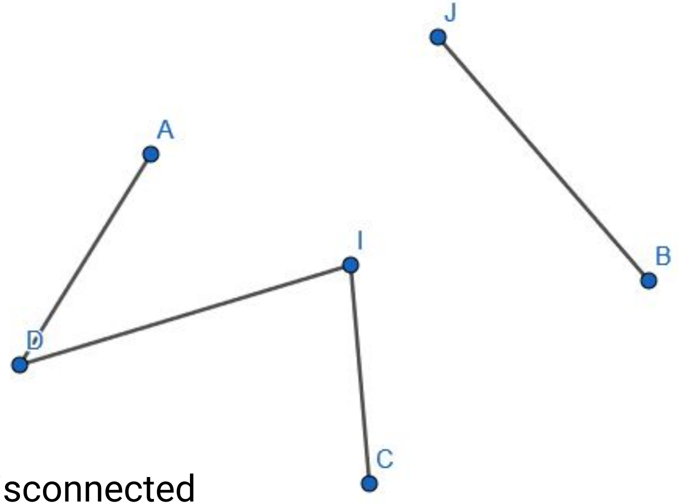
Simple Graph



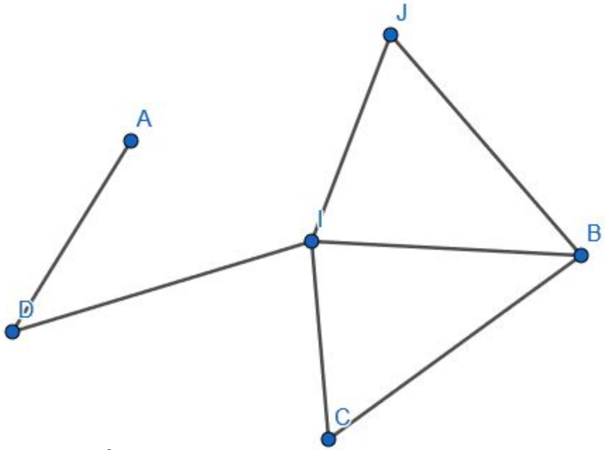
Multigraph

Connected Graphs

- In a connected graph, every vertex can be reached from every other vertex by traversing edges
- A connected subgraph of a disconnected graph is called a component



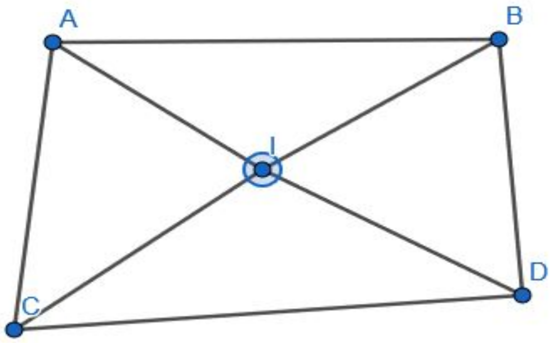
Disconnected



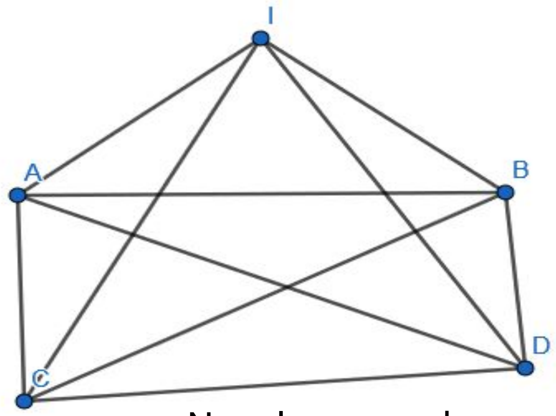
Connected

Planar Graphs

- Planar graphs can be drawn in such a way that no edges cross
- Note: This does not mean that the way they are currently drawn must have no edges crossing
- In a planar graph, the polygons bounded by edges are called “faces”
- The infinite area outside the graph is a face as well



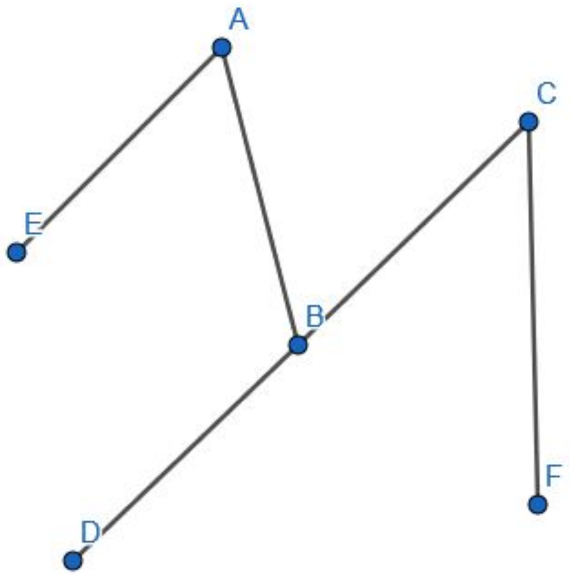
Planar graph



Nonplanar graph

Trees

- A tree is any connected graph with no cycles
- A tree with v vertices has v-1 edges and 1 face
- A tree is simple and planar

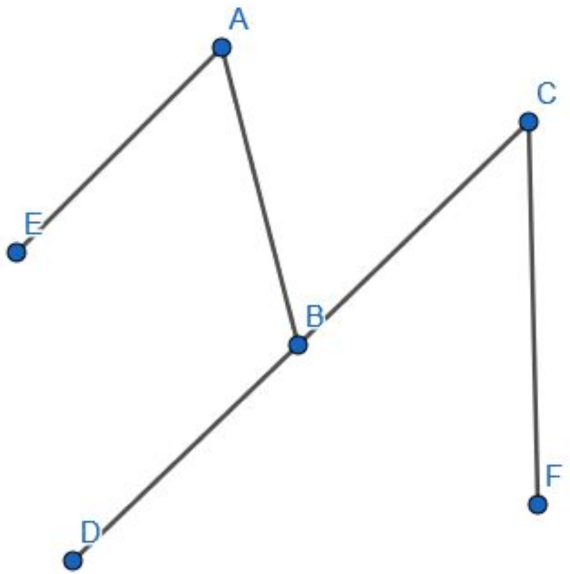


The Euler Characteristic

- For any connected planar graph with V vertices, E edges, and F faces, V-E+F=2
- This also works for vertices, faces, and edges of polyhedra
- A useful property, if we have an algorithm that runs in time O(E), then on planar graphs, that algorithm runs in time O(V)

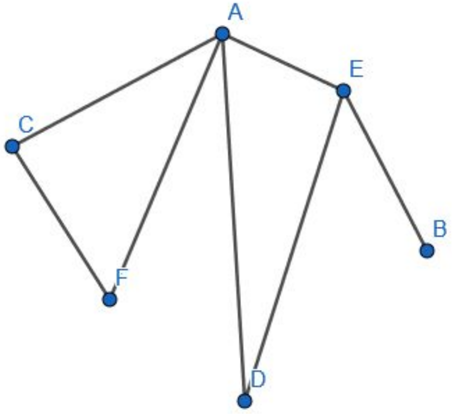
A Quick Proof of the Euler Characteristic:

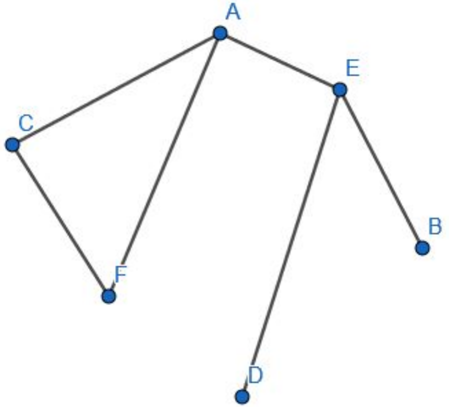
- Induction on F
- Base case: F = 1 (a tree)
- V-E+F = V - (V-1) + 1 = 2, holds!



A Quick Proof of the Euler Characteristic:

- Inductive Step: Assume EC holds for F-1 or fewer faces
- Take some graph with E edges, V vertices, F faces
- Delete one edge
- The new graph has E-1 edges, V vertices, F-1 faces
- By inductive hypothesis, V - (E-1) + (F-1) = 2
- Rearranging, V-E+F=2
- The Theorem Holds!



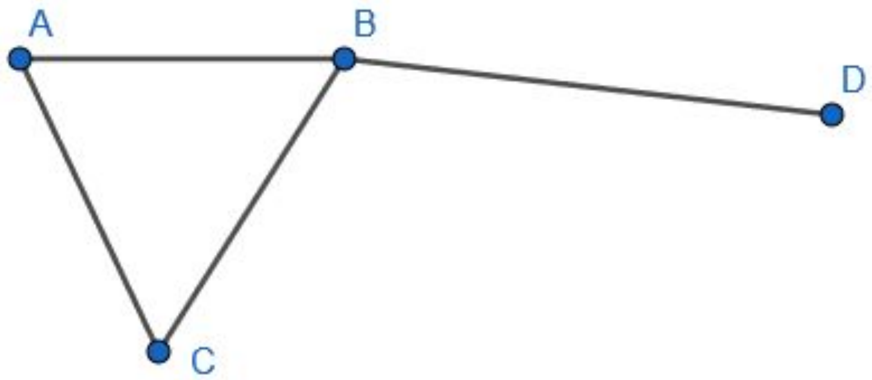


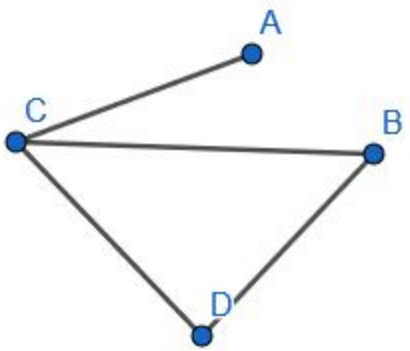
The Handshake Theorem

- The degree of a vertex, deg(v), refers to the number of edges adjacent to v
- **The sum of the degrees of the vertices is twice the number of edges**
- Notice that each edge has 2 vertices adjacent
- Therefore, if we count the edges adjacent to each vertex, each edge will be counted twice
- Intuitively, when one handshake occurs, 2 people have shook a hand

Graph Isomorphism

- Two graphs are isomorphic if they are equal up to relabeling and rearrangement
- Determining whether two large graphs are isomorphic is in general a hard problem. (Believed to be NP-Intermediate, between NP hard and P)



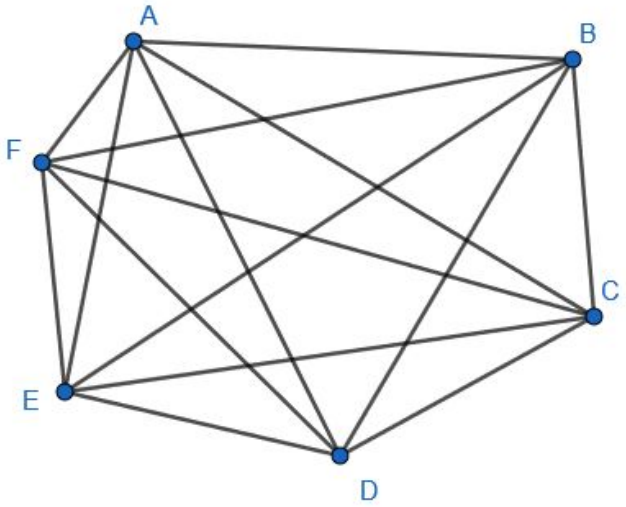


Graph Coloring

- In a graph coloring, each node is painted, and no nodes which share an edge can be painted the same color
- A 1977 has shown that every planar graph can be 4 colored
- This was one of the first major examples of a major problem solved entirely by computer
- The result was controversial as no human could follow the proof, but they could verify that the computer program was correct
- Whether a simple proof exists is still an open question

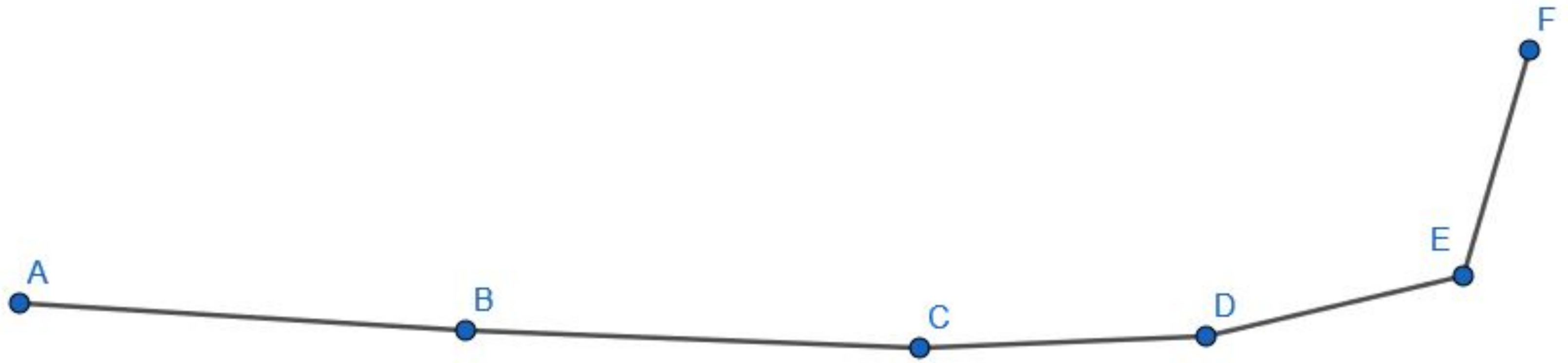
Complete Graphs

- Every vertex adjacent to every other vertex
- Has n vertices and $n(n-1)/2=O(n^2)$ edges
- Sometimes called K_n
- Simple, Connected, Planar if $n < 5$, nonplanar if $n >= 5$
- Every vertex has degree n-1
- Sometimes called a “clique”



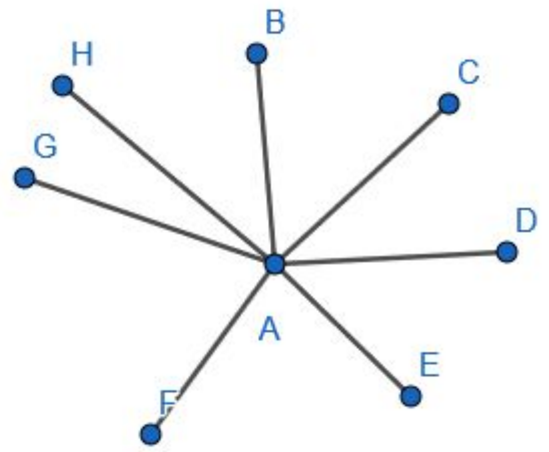
Path Graphs

- All of the vertices form a “path”
- 2 vertices have degree 1, others have degree 2
- Simple, Connected, Planar
- A special case of a tree



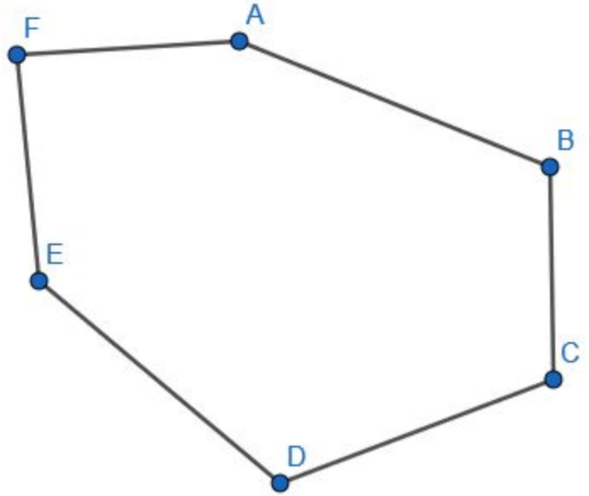
Star Graphs

- One central “hub” with many leaves attached
- Simple, connected, planar
- One vertex has degree n-1, all others have degree 1
- Also a special case of a tree



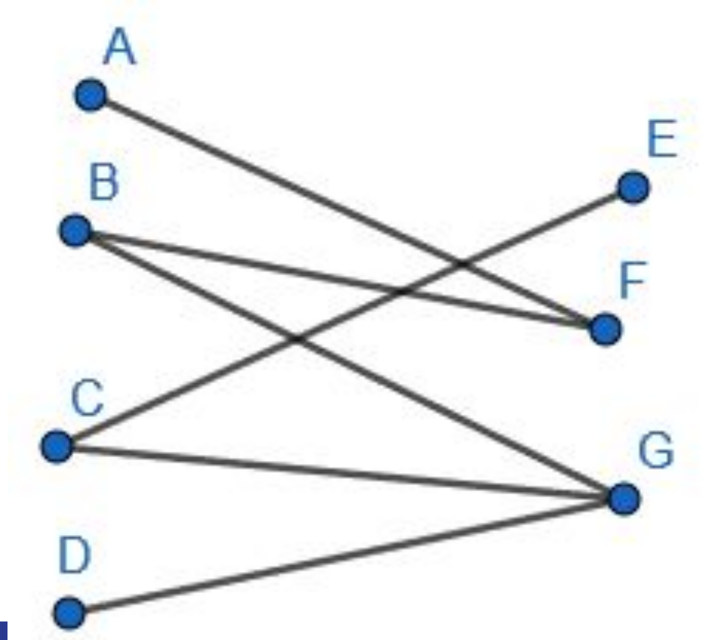
Cycle Graphs

- Vertices all form a large "loop"
- Simple, connected, planar
- All vertices have degree 2



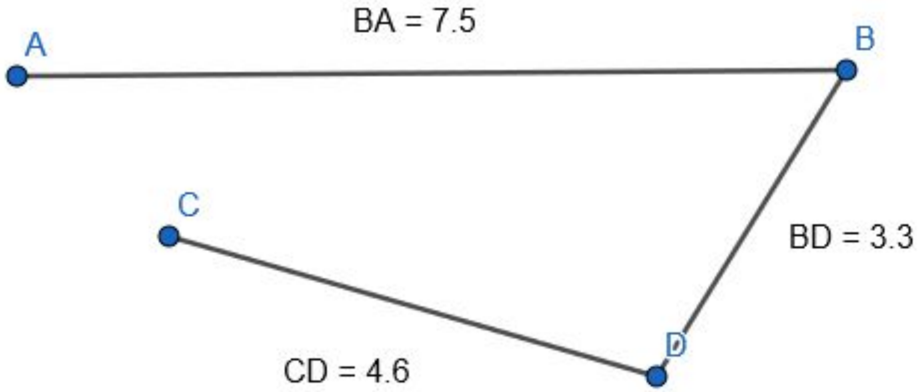
Bipartite Graphs

- 2 sets of vertices
- All edges cross from one set to the other
- No edges within a set
- Simple, Connected, sometimes planar



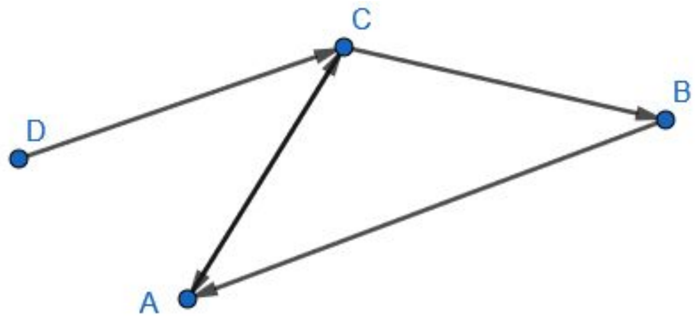
Weighted Graphs

- Sometimes it is useful to not think of all the edges as equal.
- In this case, we may attach numerical “weights” to the edges
- These could represent the length of a road, the capacity of a pipe, the strength of a bond, etc.
- Weighted graphs can facilitate a much wider variety of algorithms than unweighted graphs, as we will see.



Directed Graphs (Digraphs)

- Sometimes it is useful to give edges a direction
- Think about one way roads, water flowing in one direction through pipes, payments among peers, etc.
- Some of the ideas from undirected graphs do not translate directly to digraphs
- Components can be strongly connected (every node can be reached from every other) or weakly connected (if the edge directions are ignored, the component is connected)



Graphs In The Real World

- Facebook has 2.9 billion users, who can be represented as nodes, and friend relationships can be modeled as edges
- The world has 4 million cities, which can be represented as nodes, and roads can be modeled as edges
- The brain contains 86 billion neurons, which can be represented as nodes, and their connections can be modeled as edges

Properties We Often Observe

- Connections appear somewhat Random
- Some nodes will have far more connections than others
- Nodes are often only a few steps away from each other
- Nodes can have a combination of local and global connections

Questions We Often Ask

- Will a “giant component” (GC) form which contains a large fraction of the nodes of the network?
- If such a GC forms, how large will it be?
- If some process (like an epidemic) runs on the graph, how will the graph structure affect that process?
- If an adversary attacks the network, how easy is the GC to destroy?

Erdos Renyi Random Graphs (1959)

- An erdos Renyi random graph has 2 parameters, a number of nodes n, and a probability p
- Each pair of nodes u,v has an edge connecting them with probability p
- Easier to do math with, but usually a poor model of the real world

Barabasi Albert (Scale Free) Networks (1999)

- Start with a small set of nodes, and randomly attach new nodes to existing nodes in the graph. Sometimes nodes with more attachments are weighted more heavily during the random choices.
- Nodes which are placed earlier tend to have more connections than newer nodes.
- Very good at modeling the heterogeneous degree distributions we see in the real world
- Does not model local spatial relationships

Watts-Strogatz (Small World) Networks (1998)

- Start with a graph with a lot of local structure (such as a cycle, or a slightly more general cycle)
- Randomly rewire some of the connections in the graph
- Models the combination of local and global connections from the real world well
- Tends to produce unrealistic degree distributions