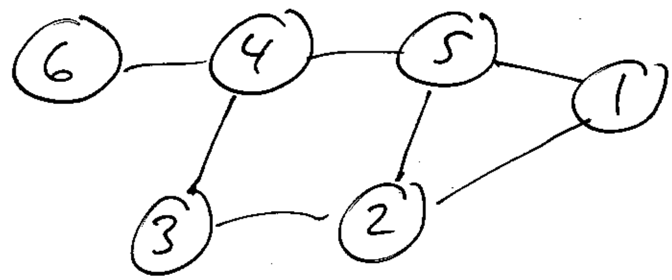
Graphs



dfn:

A graph G is defined by a pair of sets:

- a finite set V of items, called vertices
- a set of pairs (maybe empty) of items, called edges.

$V = \{1, 2, 3, 4, 5, 6\}$

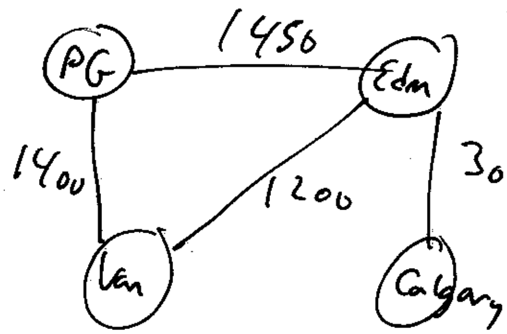
$E = \{\{1,2\}, \{2,3\}, \{3,4\}, \{4,5\}, \{4,6\}, \{5,1\}, \{5,2\}\}$

Adjacency List or Adjacency Matrix } how to represent a graph in java.

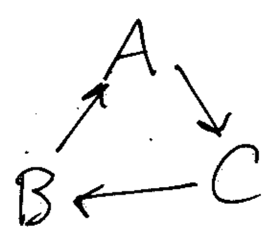
Weighted graph: "network"

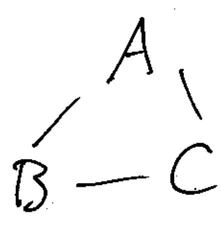
Associates a label/weight/cost with every edge in the graph.

- Minimum cost spanning tree
- shortest paths
- maximal flow



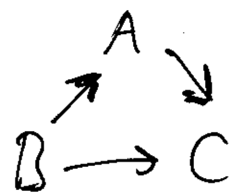
Cyclic graph:
a graph is "cyclic" if it contains at least one cycle/circuit/loop:





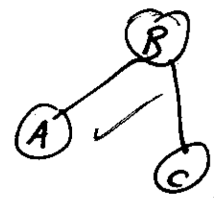
digraph:
directed graph:

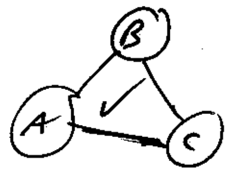
Acyclic Graph



Connected graph

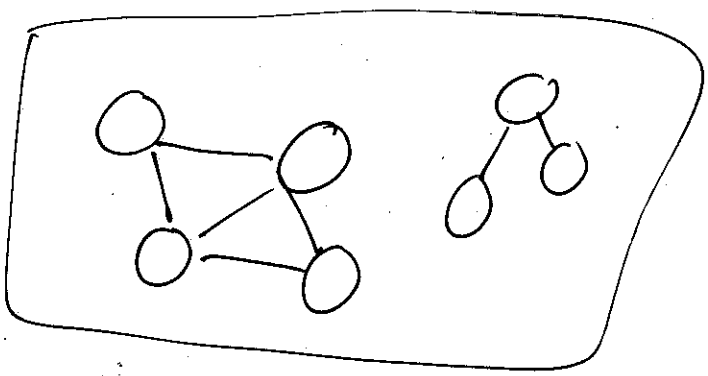
A graph is "connected" iff for every pair of its vertices u and v, there is a path from u to v.





connected

not connected



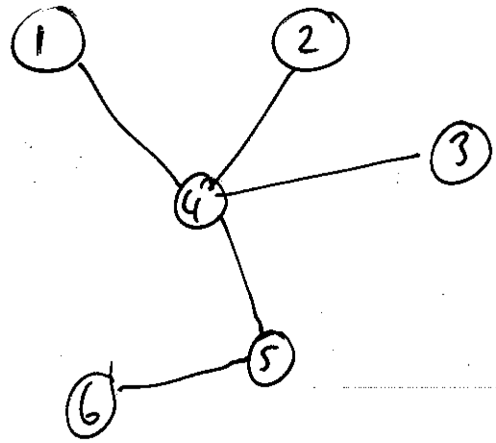
unconnected

Trees

A tree is an undirected, connected, acyclic graph.

n vertices

n-1 edges

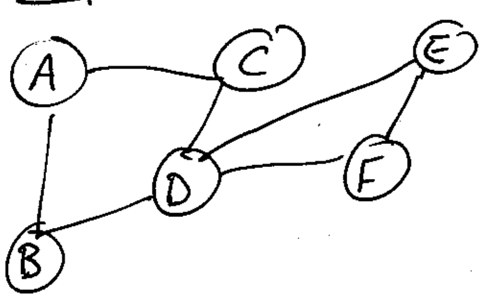


The unique simple path connecting vertex 2 to vertex 6 is 2-4-5-6.

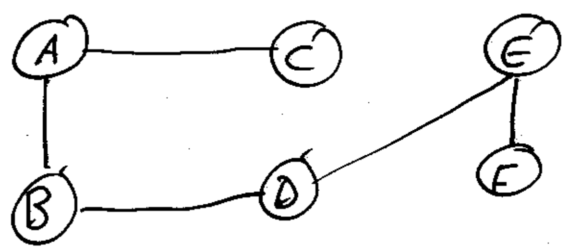
Spanning Tree:

dfn: A spanning tree of a connected graph is its tree that contains all the vertices of the graph, and some (or all) of its edges.

Graph:

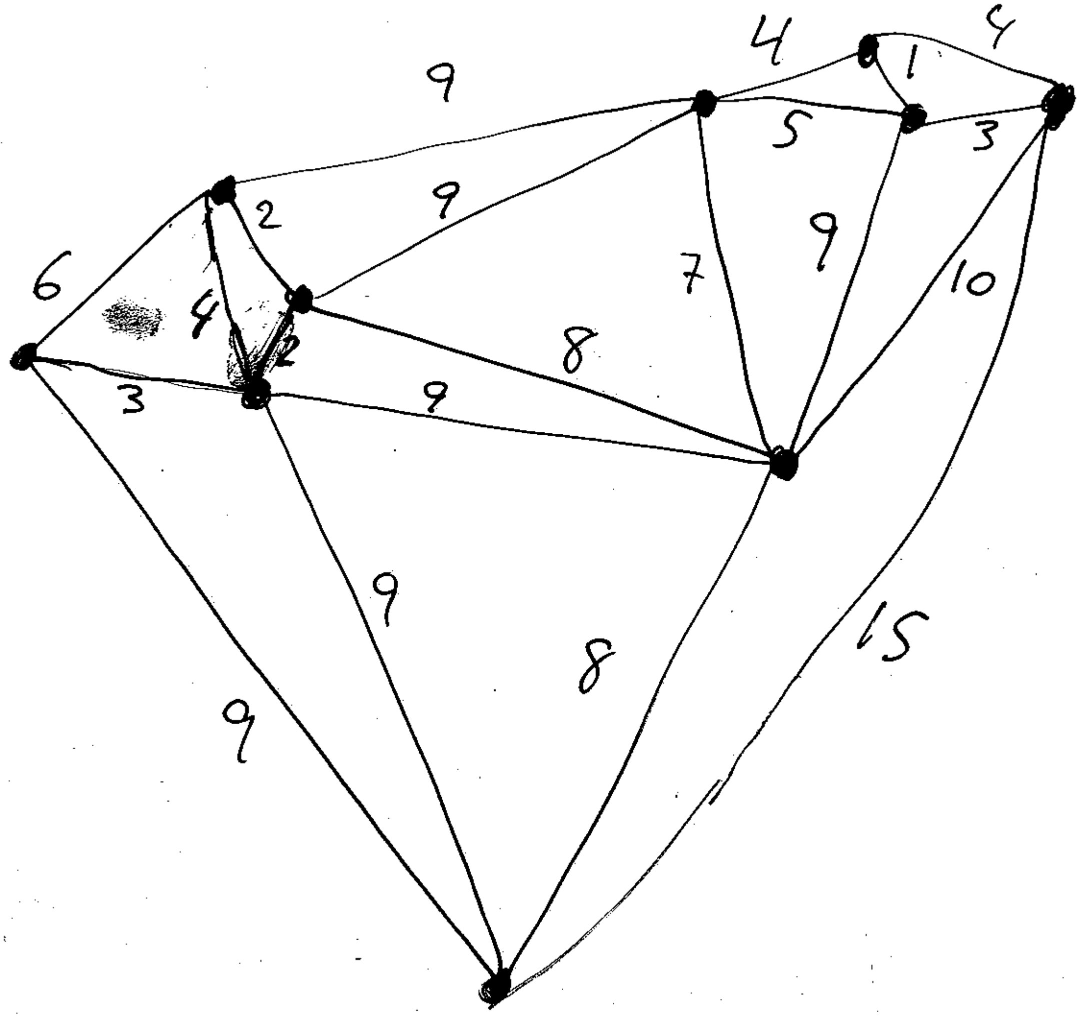


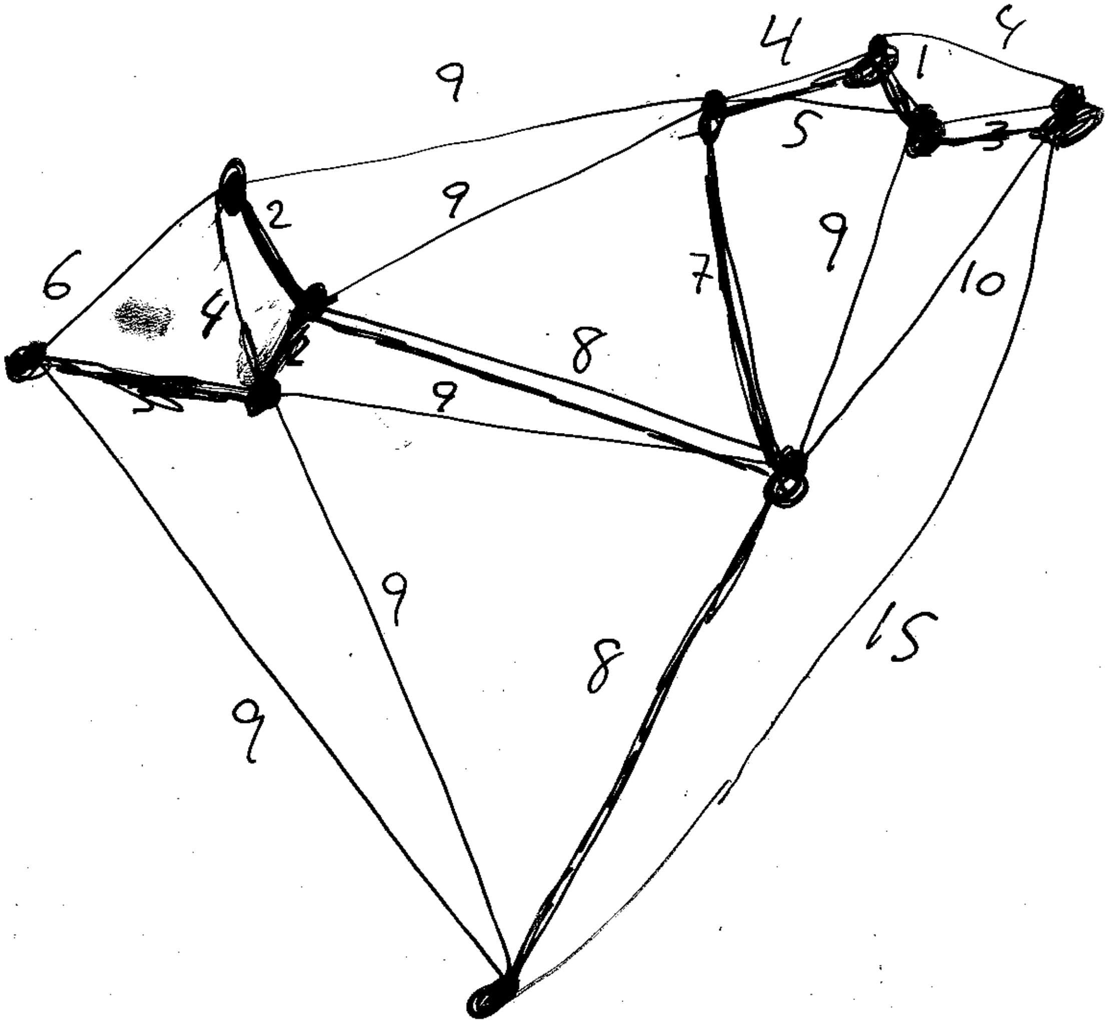
One Spanning Tree (out of many)



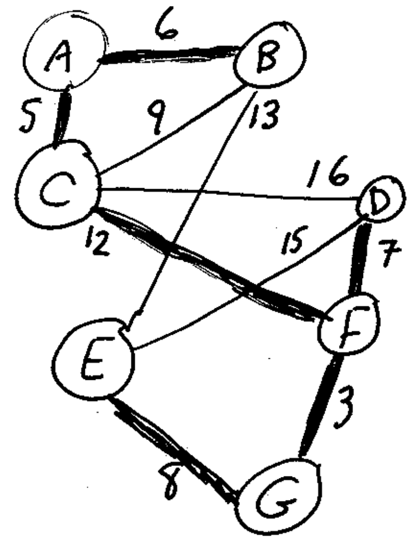
Minimum Spanning Tree MST

- Given a connected, undirected graph, a spanning tree of that graph is a subgraph which is a tree and connects all the vertices together.
- A single graph can have many different spanning tree.
- We assign a weight/cost to each edge, a number describing how unfavorable it is. A spanning tree's weight is the sum of all these edges' weights.
- A minimum (weight) spanning tree is the spanning tree with weight less than/equal to the weight of every other spanning tree.


4
4
9
1
5
3
2
9
9
6
7
10
4
2
8
3
9
9
8
15
9


9
4
4
1
5
3
9
2
6
4
9
7
10
8
9
9
8
15
9

To save money, this airline will shut down all unnecessary connections. It wants to leave open only the minimum connections required that still connect each city to all other cities (although some connections may be indirect).

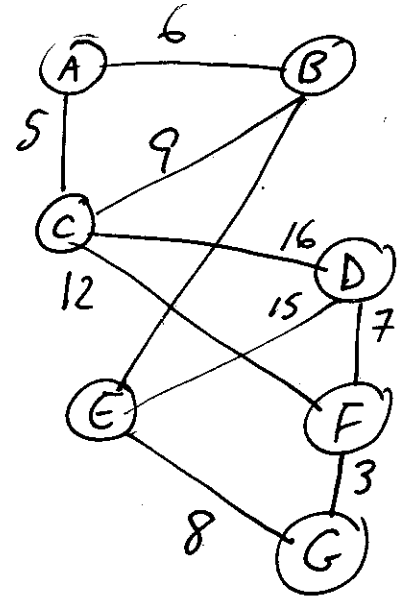


Finding the MST

- There are many ways to find the MST
- Two "greedy" algorithms:
- Kruskal's
- Prim's

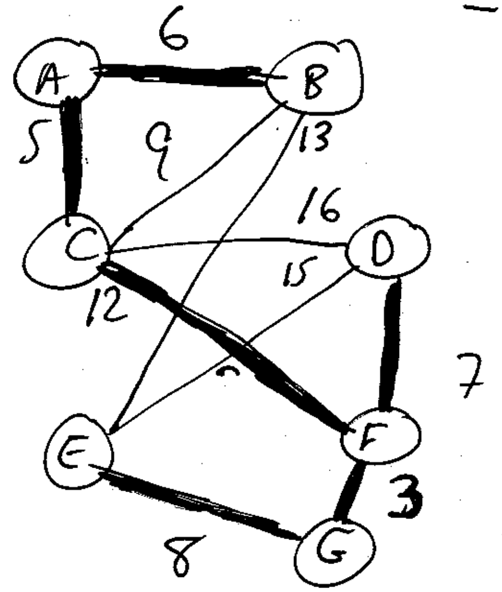
Kruskal's Algorithm:

Consider a connected, weighted graph with n vertices.



1. Look at the edges of the graph one a time, in order of increasing weight.
2. Add the edge-in-question to the MST being built, iff a cycle is not created.
3. After $n-1$ edges have been added, these edges - along with the graph's vertices - form a MST.

Kruskal's



Step	Edge	Weight	Add?
1	F-G	3	Yes
2	A-C	5	Yes
3	A-B	6	Yes
4	D-F	7	Yes
5	E-G	8	Yes
6	B-C	9	No
7	C-F	12	Yes

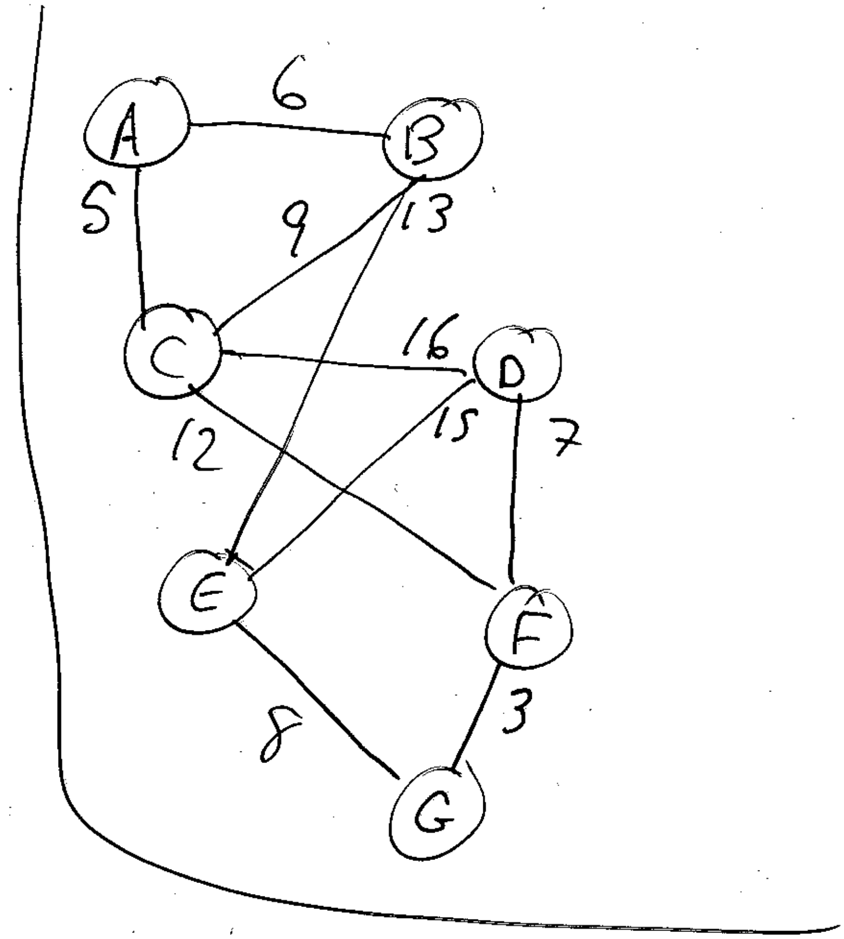
- Easy to code
- Only applicable to small # of problems
- intermediate MST is not a tree/connected but always acyclic.

Prim's Algorithm

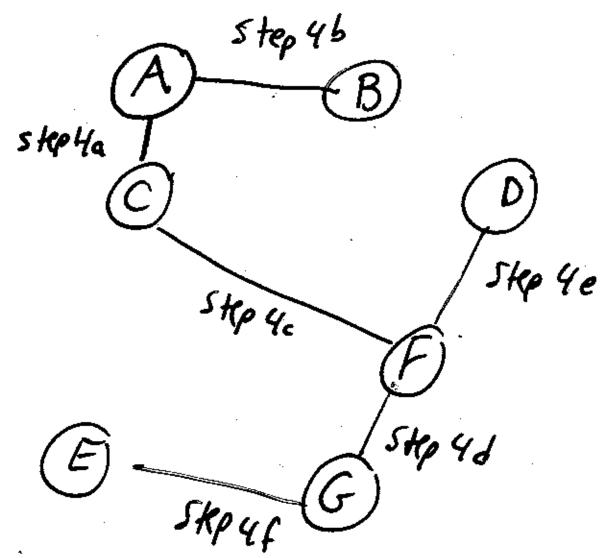
1. Pick a vertex from the graph to start with
2. This vertex is added to the MST-in-progress (with no edges)
3. Let V be the set of all vertices in the graph, not including the vertex you just picked.
4. Find an edge in the graph that connects your MST-in-progress to one of the vertices in V. Make sure this edge has the least weight of all edges that connect your MST-in-progress to a vertex in V. Add this edge and/its endpoint to your MST-in-progress, and delete it from V.
5. Repeat step 4 until every vertex is conected (n-1 edges are in your MST-in-progress).

Prim's:

1. Start with A



MST:



V.

~~B~~ ~~C~~ ~~D~~ ~~E~~ ~~F~~ ~~G~~

Done

always have a connected tree with Prim's.

Q] How many of the $\binom{52}{5} = 2\,598\,960$ 5-card poker hands contain 2 pairs?

YES Do THIS!	NO DON'T DO THIS!

YES — Do THIS!

A 2 3 4 5 6 7 8 9 10 J Q K

Step 1: pick 2 denominations: $\binom{13}{2}$ eg 5's & J's

Step 2: pick 2 of the 4 cards for denomination #1: $\binom{4}{2}$

Step 3: pick 2 of the 4 cards for denomination #2: $\binom{4}{2}$

Step 4: $52-4-4=44$

$\binom{44}{1}$ ways to pick the 5th card.

$\binom{13}{2}\binom{4}{2}\binom{4}{2}\cdot 44 = \boxed{123552}$ ✓

NO — DON'T DO THIS!

Step 1: pick denomination #1: $\binom{13}{1}$

Step 2: pick 2 of those 4 cards: $\binom{4}{2}$

Step 3: pick a 2nd denomination: $\binom{12}{1}$

Step 4: pick 2 of those 4 cards: $\binom{4}{2}$

Step 5: pick the 5th card: $\binom{44}{1}$

247104 ✗

This permits this:

	AND
Step 1: 5's	Step 1: J's
Step 2: 5♡ 5◇	Step 2: J♡ J◇
Step 3: J's	Step 3: 5's
Step 4: J♡ J◇	Step 4: 5♡ 5◇
Step 5: K◇	Step 5: K◇

COUNTS ALL HANDS TWICE

Exam!

Midterm: Avg mark $68\frac{1}{2}/70$

- Perm. / combinations ☆ probability
- sets
- graphs / trees
- simple FSM
- binomial thm.
- complexity of algorithms.

exam based on my notes.

aim for 2 hours (3 if we need it)