# Spanning and weighted spanning trees

## A different kind of optimization

(graph theory is cool.)

## Spanning and weighted spanning trees

A different kind of optimization

(graph theory is cool.)

Definitions and examples

# Graphs

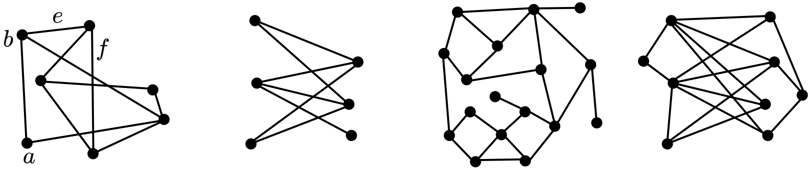
A *graph* is a collection of vertices (that look like dots $\bullet$)

and edges (that look like curves ),
where each edge joins two vertices.
(Formally, a *graph* is a pair $G = (V, E)$, where $V$ is a set of dots and $E$ is a set of pairs of vertices.)

## Graphs

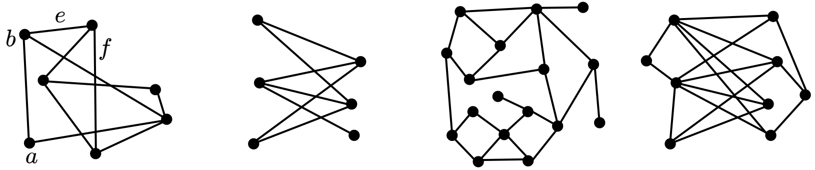
A *graph* is a collection of vertices (that look like dots $\bullet$)

and edges (that look like curves  ),
where each edge joins two vertices.
(Formally, a *graph* is a pair $G = (V, E)$, where $V$ is a set of dots and $E$ is a set of pairs of vertices.)
Here are a few examples of graphs:



## Graphs

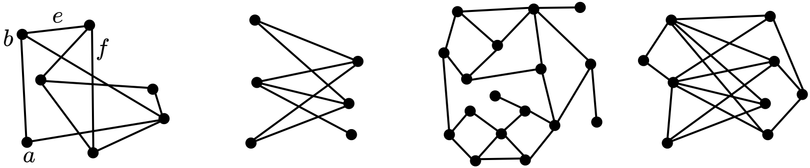
A *graph* is a collection of vertices (that look like dots $\bullet$)
and edges (that look like curves ),
where each edge joins two vertices.
(Formally, a *graph* is a pair $G = (V, E)$, where $V$ is a set of dots and $E$ is a set of pairs of vertices.)
Here are a few examples of graphs:




Two vertices joined by an edge are called *adjacent* (see $a$ and $b$).
Two edges that meet at a vertex are called *incident* (see $e$ and $f$).

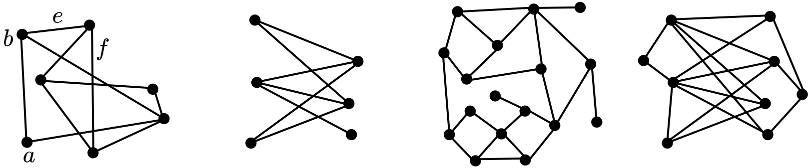
## Subgraphs

A *subgraph* is a graph that is contained within another graph. For example, here the second graph is a subgraph of the fourth graph.

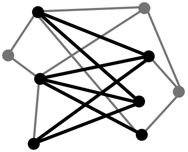


## Subgraphs

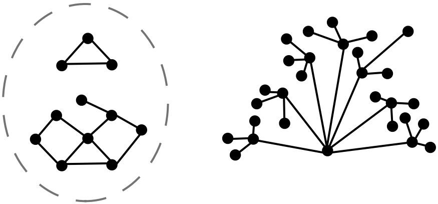
A *subgraph* is a graph that is contained within another graph. For example, here the second graph is a subgraph of the fourth graph.




Here is the second graph, shown as a subgraph of the fourth graph.

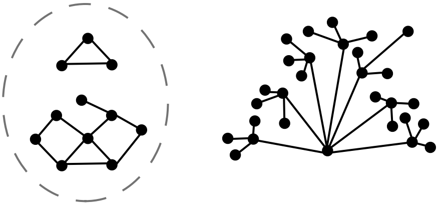


## Trees



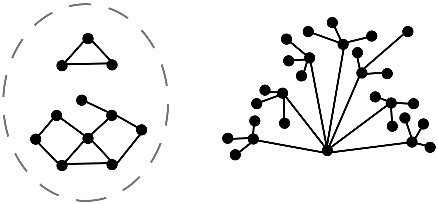
In a *connected* graph, there is a way to get from any vertex to any other vertex without leaving the graph.

## Trees



In a *connected* graph, there is a way to get from any vertex to any other vertex without leaving the graph. The left graph above is *not* connected.

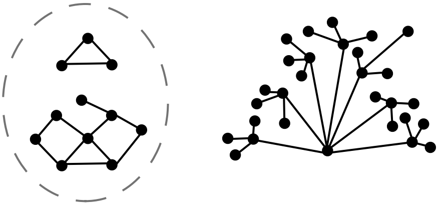
# Trees



In a *connected* graph, there is a way to get from any vertex to any other vertex without leaving the graph. The left graph above is *not* connected.

A *cycle* is a sequence that alternates between vertices and edges, and whose only repetition is the first/last vertex.

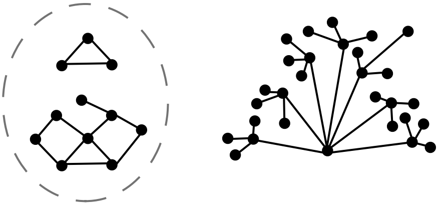
# Trees



In a *connected* graph, there is a way to get from any vertex to any other vertex without leaving the graph. The left graph above is *not* connected.

A *cycle* is a sequence that alternates between vertices and edges, and whose only repetition is the first/last vertex. A cycle is shown by itself as the top part of the left graph above.

## Trees



In a *connected* graph, there is a way to get from any vertex to any other vertex without leaving the graph. The left graph above is *not* connected.

A *cycle* is a sequence that alternates between vertices and edges, and whose only repetition is the first/last vertex. A cycle is shown by itself as the top part of the left graph above.

A *tree* is a graph that is connected and has no cycles. One is shown to the right above.

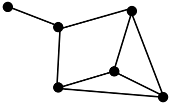
A *forest* is a bunch of trees.

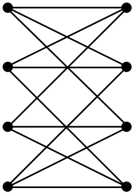
## Spanning Trees

A *spanning tree* is a tree that contains all the vertices of a given graph. Basically, it is the largest tree contained in a graph.

## Spanning Trees

A *spanning tree* is a tree that contains all the vertices of a given graph. Basically, it is the largest tree contained in a graph.

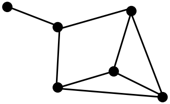


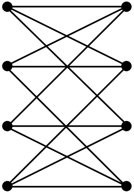


Here are spanning trees of the above-pictured graphs:

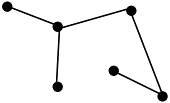
# Spanning Trees

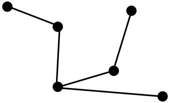
A *spanning tree* is a tree that contains all the vertices of a given graph. Basically, it is the largest tree contained in a graph.

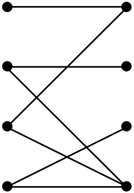


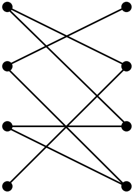


Here are spanning trees of the above-pictured graphs:



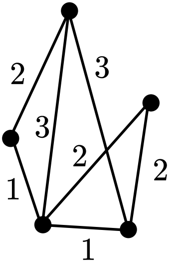






## Weighted Graphs

*Weights* are labels on the edges and/or vertices of a graph that often denote costs or distances or energies. Here's a weighted graph:



## Weighted Spanning Trees

The *total weight* of a spanning tree is the sum of the weights on its edges.

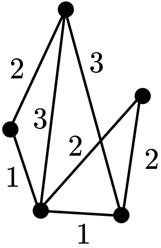
A *minimum-weight* spanning tree is one that has the lowest possible total weight.

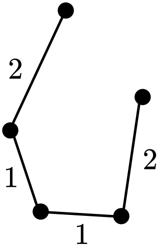
# Weighted Spanning Trees

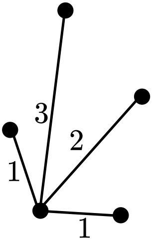
The *total weight* of a spanning tree is the sum of the weights on its edges.

A *minimum-weight* spanning tree is one that has the lowest possible total weight.

Here are a weighted graph, a spanning tree of total weight 6, and a spanning tree of total weight 7; are either of these minimum-weight spanning trees?



# Time for Worksheets!

No, really. It's time to work on worksheets now.

## Final notes: MathILy

- intensive summer program for super-smart, super-cool students
- extremely interactive and silly and inventive classes
- discrete and applicable college-level mathematics
- Root class, then Week of Chaos, then Branch classes

`http://www.mathily.org`