7.2 Exercise: In-class exercises are meant to introduce you to a new topic and provide some practice with the new topic. Work in a team of up to 4 people to complete this exercise. You can work simultaneously on the problems, or work separate and then check your answers with each other. You can take the exercise home, score will be based on the in-class quiz the following class period. **Work out problems on your own paper** - this document just has examples and questions.

7.2 Proofs about Graphs and Trees

Although this section is named "Proofs", we are actually going to focus on Trees for this section.

7.2.1 Introduction to Trees

Question 1

[1]

a. How many vertices does each graph have?

G_1 ______ G_2 ______ G_3 ______ G_4 ______

b. How many edges does each graph have?

G_1 ______ G_2 ______ G_3 ______ G_4 ______

c. Which graph is NOT a connected graph?

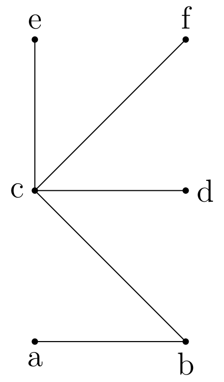
d. Which of the graphs has at least one cycle?

e. Which of the graphs is a tree?
Hint: A simple connected graph with no cycles is a ***tree****.*

[1]From Jim Van Horn's POGIL Activity 16

Question 2

[2]



G_1

a. What is the degree of each of the vertices in G_1?

$deg(a)$ ______ $deg(b)$ ______ $deg(c)$ ______

$deg(d)$ ______ $deg(e)$ ______ $deg(f)$ ______

b. List the leaves for G_1.

Vertices of degree 1 in a tree are called **leaves** of the tree.

[2]From Jim Van Horn's POGIL Activity 16

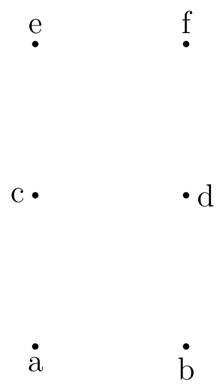
Trees
A **Tree** is a connected simple graph that has no cycles. Vertices of degree 1 in a tree are called **Leaves** of the tree. [a]

[a] Discrete Mathematics, Ensley and Crawley

Question 3

[3]

Given these 6 vertices, draw a tree other than G_1 or G_2.



a. How many edges are in your new tree?

b. How many leaves on your new tree?

c. If you removed one edge, would the graph still be connected?

Property of a tree
A tree with n vertices will have $n - 1$ edges. In other words, it is a connected graph and if you remove an edge then it will become a disconnected graph.

[3] From Jim Van Horn's POGIL Activity 16

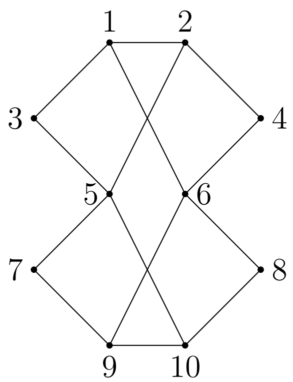
7.2.2 Subgraphs and Trees

Subgraphs

A graph H is a **subgraph** of a graph G if all nodes and edges in H are also nodes and edges in G. [a]

[a] Discrete Mathematics, Ensley and Crawley

Question 4
[4]



G G_1 G_2

a. Draw a graph G_1 above using vertices and edges from G...
Vertices: 1, 2, 5, Edges: {1, 2} and {2, 5}.

Is this a **subgraph**?

Are all the vertices of G_1 also nodes of G?

Are all the edges of G_1 also edges of G?

b. Draw a graph G_2 above using vertices and edges from G...
Vertices: 1, 3, 4, Edges: {1, 3} and {3, 4}

Is this a **subgraph**?

Are all the vertices of G_2 also nodes of G?

Are all the edges of G_2 also edges of G?

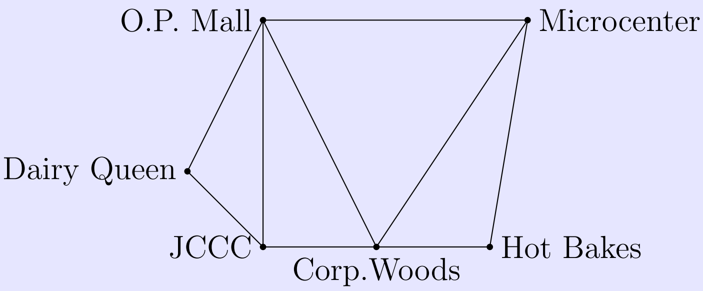
[4] From Jim Van Horn's POGIL Activity 16

7.2.3 Spanning Trees

Spanning Trees

Let G be a simple connected graph. The subgraph T is a **spanning tree** of G if T is a tree and every node in G is a node in T. [a]

Example: [b]

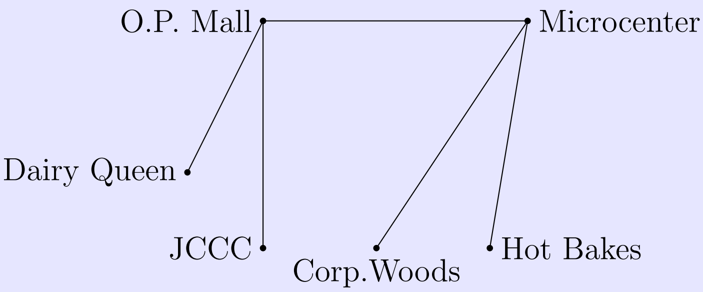


To get from Corporate Woods to JCCC, there are three paths leading into JCCC:

(1) Directly from CW $\rightarrow$ JCCC,
(2) CW $\rightarrow$ OP Mall $\rightarrow$ JCCC, and
(3) CW $\rightarrow$ OP Mall $\rightarrow$ Dairy Queen $\rightarrow$ JCCC

We want to make a graph that connects all locations with the fewest paths. One way to do this is to remove edges of a cycle until no additional edges can be removed without getting a disconnected graph.

One example result is this:



[a] Discrete Mathematics, Ensley and Crawley
[b] From Jim Van Horn's POGIL Activity 16

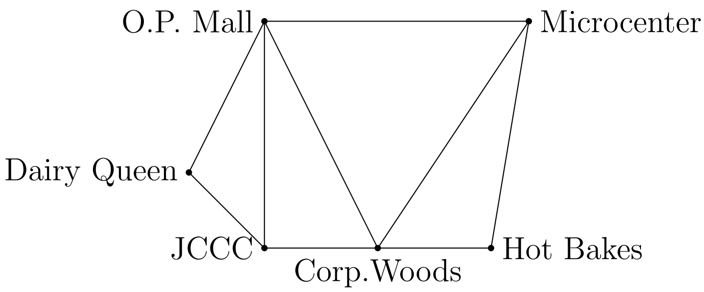
Spanning Tree algorithm [a]

1. Begin with a simple connected graph G_0.
2. For each $i \geq 1$, as long as there is a cycle in G_{i-1}...
   (a) Choose an edge e in any cycle of G_{i-1}, and form the subgraph G_i of G_{i-1} by deleting e from G_{i-1}
3. The final result G_k will be a spanning tree of G_0. This is a spanning tree.

[a] Discrete Mathematics, Ensley and Crawley

Question 5

Follow the algorithm to create a Spanning Tree from this map. "x" out edges that you choose to delete as you go. Draw your spanning tree below.



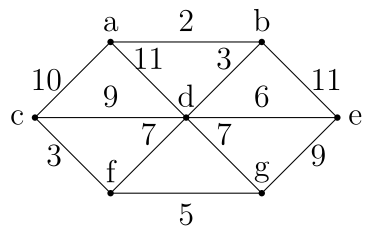
7.2.4 Minimal Spanning Trees

Minimal Spanning Trees
Prim's Minimal Spanning Tree algorithm [a]

1. Given a connected simple graph G with $n+1$ nodes.
2. Let v_0 be any node in G, and let $T_0 = \{v_0\}$ be a tree with one node and no edges.
3. For each k from $\{1, 2, ..., n\}$...
   (a) Let $E_k = \{e$ an edge in G : e has one endpoint in T_{k-1} and the other endpoint not in $T_{k-1}\}$.
   (b) Let e_k be the edge in E_k with the smallest weight. (In case of a tie, choose any edge of the smallest weight.)
   (c) Let T_k be the tree obtained by adding edge e_k (along with its node not already in T_{k-1} to T_{k-1}.
4. T_n is the tree returned by the algorithm.

[a] Discrete Mathematics, Ensley and Crawley

Question 6



Use Prim's algorithm to find a minimal spanning tree for the graph.