Math 140: Tuesday, Sep 10, Day 7 Notes

Agenda for Tuesday, Sep 10,2024

Reminders

- **Online Homework 3** due Thursday by 11:59pm
- Office Hour **today 1-2:15pm and Wed 3:45-5pm** in C162
- **Second Group Quiz** Thursday in class (you can use your hand-written notes)
- **Travel Abroad Fair today by Cafe Willow**

Recall

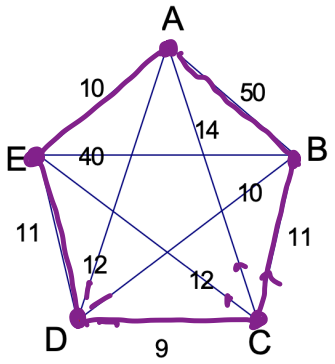
- Brute Force
- Nearest Neighbor

Quiz 2 Topics:

- Eulerization / Semi-Eulerization
- Hamilton circuits / Paths
- Complete graphs
- Algorithms for finding optimal circuit

Group Activity:

Example 1. A delivery service must deliver packages at Buckman (B), Chatfield (C), Dayton (D), and Evansville (E), and then return to Arlington (A), the home base. The following graph shows the estimated travel times (in minutes) between the cities.



Check your answers!

1. Find the nearest-neighbor circuit for starting vertex A.

   A E D C B A

   travel time: 10 + 11 + 9 + 11 + 50 = 91 minutes

2. Find the nearest-neighbor circuit for starting vertex B.

   B D C E A B

   10 + 9 + 12 + 10 + 50 = 91 minutes

3. Find the nearest-neighbor circuit for starting vertex C.

   C D B E A C

   9 + 10 + 40 + 10 + 14 = 83 minutes

4. Find the nearest-neighbor circuit for starting vertex D.

   D C B E A D

   9 + 11 + 40 + 10 + 12 = 82 minutes

5. Find the nearest-neighbor circuit for starting vertex E.

   E A D C B E

   10 + 12 + 9 + 11 + 40 = 82 minutes

6. Which circuit is the best of all of these? Rewrite it with starting vertex A.

   either starting at D or E: start at D rewrite:

   A D C B E A

Repeated Nearest Neighbour Algorithm

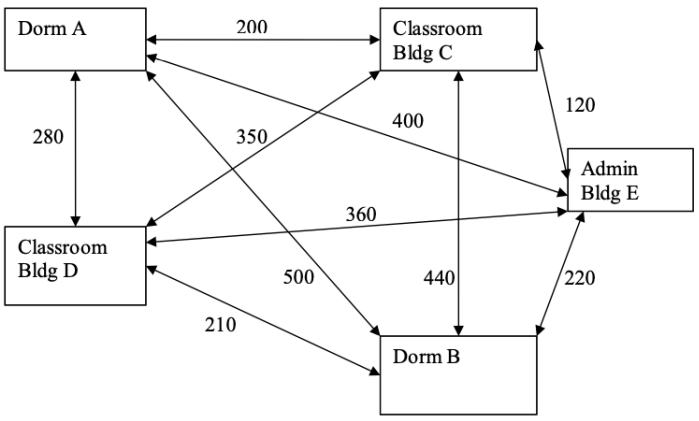
What you just did for the previous example is called the Repeated Neighbour Algorithm.

Steps for RNNA:

1. Do the Nearest Neighbour Algorithm starting at each vertex.
2. Choose the circuit produced with minimal weight.

Networks and Spanning Trees

During the first year on its new campus, Cold Region State College had several unfortunate cases of frostbite among students walking to class. To avoid this happening in the future, the administration intends to convert several of the existing walkways to protected walkways so that students can **go from any building on campus to another building** without being exposed to the inclement weather. To minimize expenses and the time of construction, they wish to draw up plans for the **minimum total length of walkways that need to be converted**. How can this be done? The campus map is shown below with each existing walkway shown. The distances are given in yards.

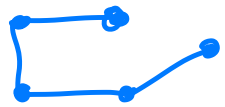


We want a subset of the existing edges so that the graph is still connected includes all vertices, and there exactly one path between any two buildings.

What are we asked to find? A circuit? A path? Or something else?

minimum spanning tree (get to it Thursday)

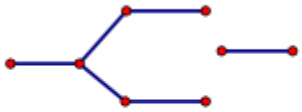
Network - is a connected graph

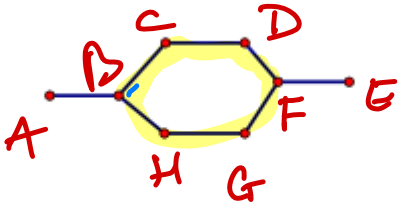
ex: 
yes

ex: 
not a network.

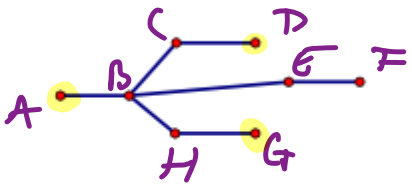
Tree - is a network that has no circuits.

Example 2. Are the following [redacted] a tree? If not, why not?


no
because
not connected



BCDFGHB
is a circuit
so not a
tree



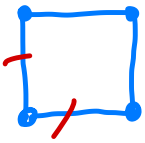
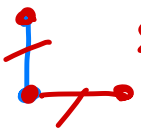
yes a
tree
it has 8
vertices
and 7 edges.

Properties of Trees

1. The Single-Path Property: There is exactly one path between any two vertices in a tree.
2. The All-Bridges Property: Every edge is a bridge in a tree.
3. The N-1 Edges Property: A tree with N vertices has n-1 edges.

ex: A tree with 10 vertices has 9 edges.

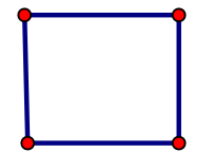
Subtree is a tree that contains a subset of edges from the original graph.

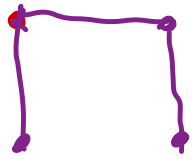
Ex  tree   subtree

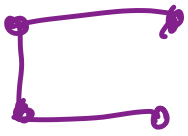
some of them

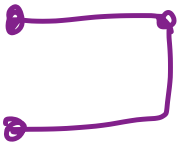
Spanning Tree - is a subtree that contains all of the vertices of the original graph.

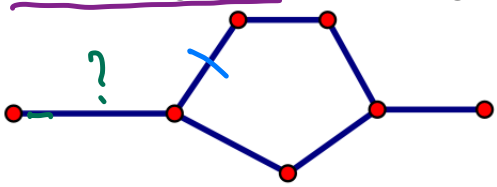
Example 3. Find all possible spanning trees of each graph.

1. 

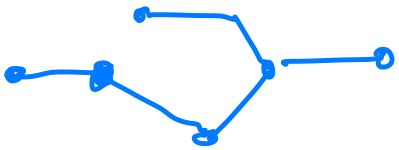


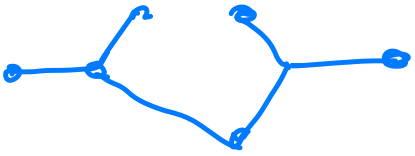


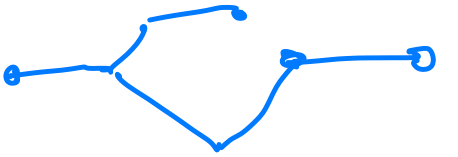


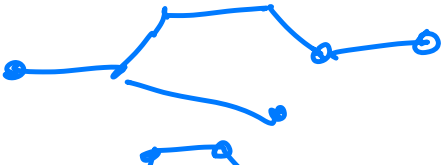
2. 

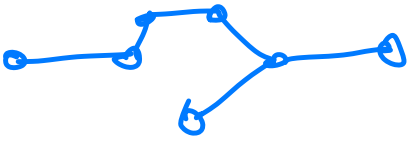
4 vertices
3 edges









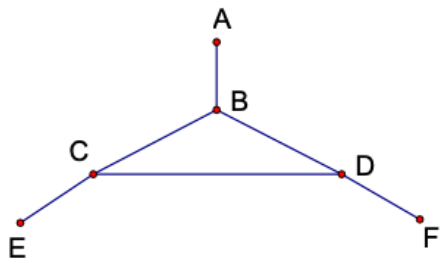


7 vertices
6 edges

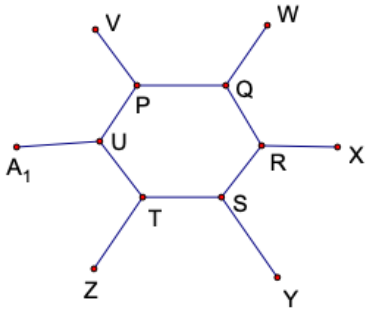
5 spanning trees

Example 4. Find how many spanning trees each graphs has (No need to draw them.)

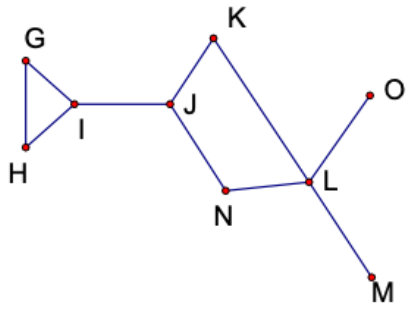
1.



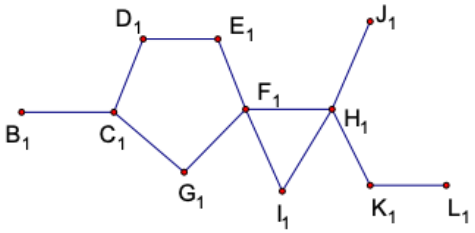
2.



3.



4.



5.