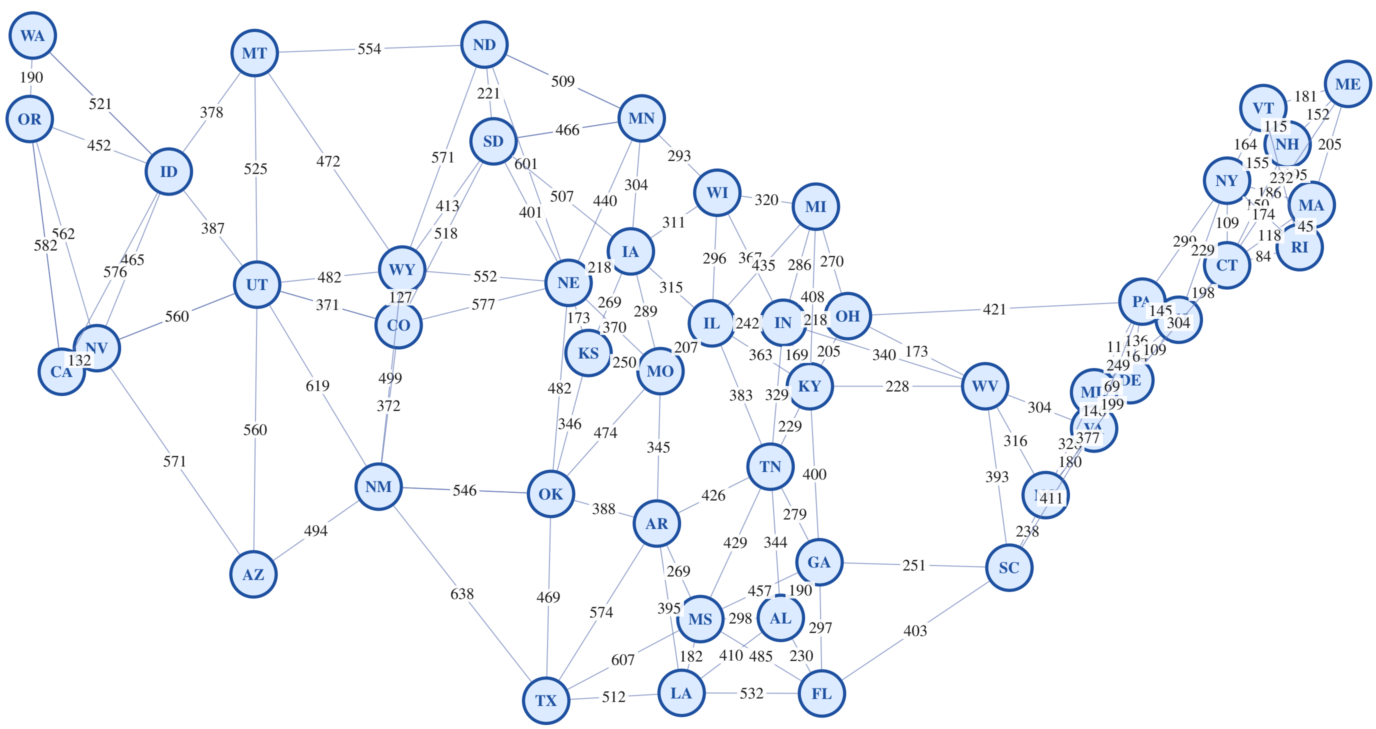
MATH F113X: Kruskal's Algorithm

Goals:

- Understand the terms: tree, spanning tree, minimum cost spanning tree
- Understand how to use Kruskal's Algorithm to find a minimum cost spanning tree
- Know of applications of minimum cost spanning trees

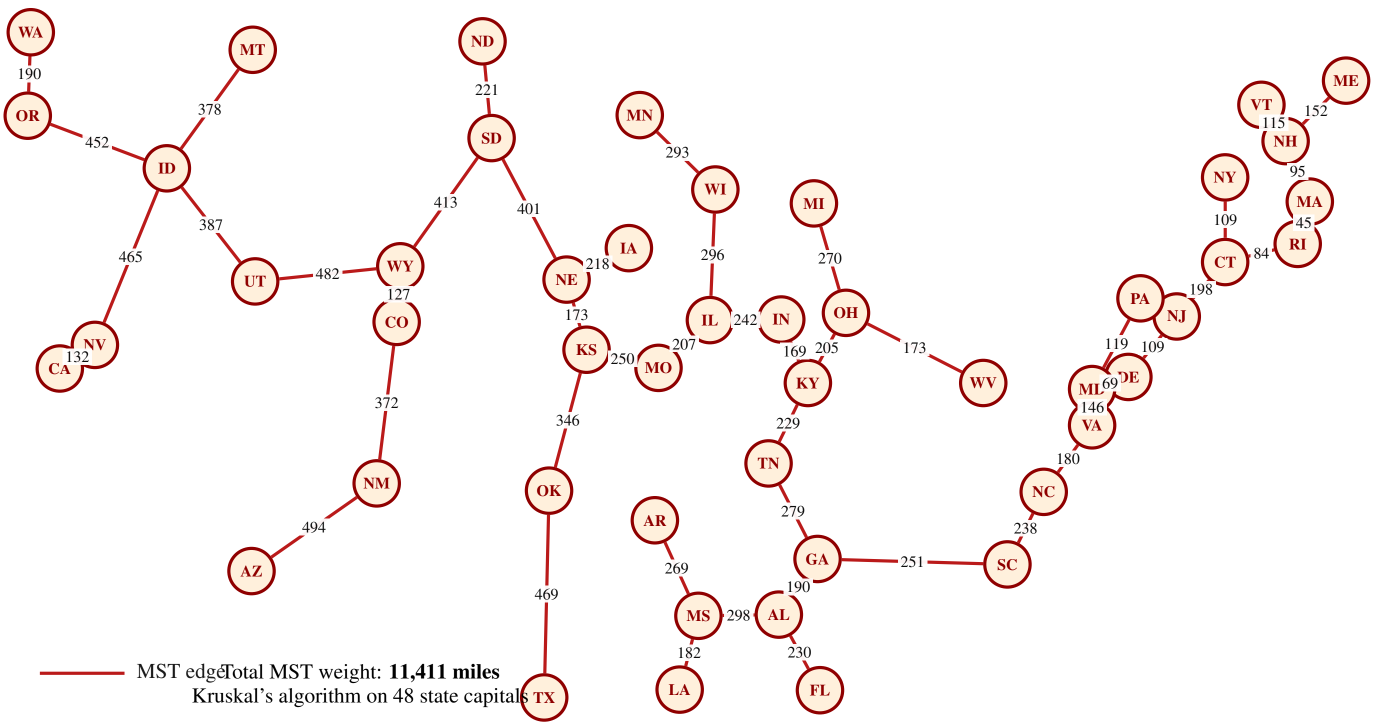
Weighted Road Graph: 48 Contiguous U.S. State Capitals

Edge weights = approximate road distance in miles. Each capital connected to its 4–5 nearest neighbors.



Minimum Weight Spanning Tree (Kruskal's Algorithm)

Total MST weight ≈ **11,411** miles. Red edges form the unique spanning tree of minimum total road distance.



MATH F113X: Kruskal's Algorithm

1. Definitions

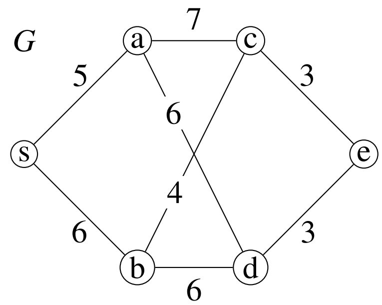
   (a) **weighted graph**

   (b) **tree**

   (c) **spanning tree**

   (d) **minimum cost spanning tree**

2. Example:



3. $\boxed{\text{Kruskal's Algorithm}}$

   input: a graph, G, with costs (or weights) on the edges
   output: a spanning tree, T, of minimum cost
   Steps:

   (a) (Initialization Step:) T is a graph on the vertex set of G but with no edges.

   (b) (Iterative Step:)

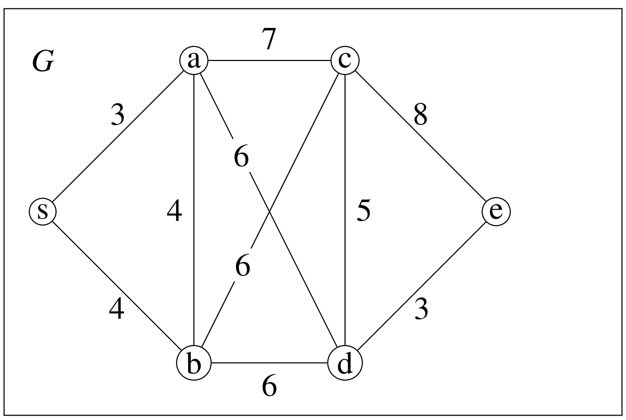
      i. Select the cheapest unused edge in the graph. (Ties are broken alphabetically.)

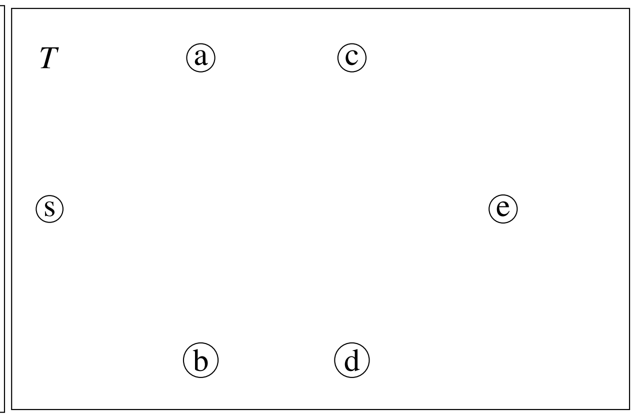
      ii. If the edge does **not** create a cycle, add the edge to T. Otherwise, reject the edge.

      iii. Mark the edge as used.

      iv. If T is a spanning tree, STOP. Otherwise return to the beginning of the iterative step.

4. Use Kruskal's Algorithm to find the minimum cost spanning tree for the graph G below.





Used?	edges	weights

5. Think of an application of Kruskal's Algorithm.