Java 2 Minimum Spanning Tree

- A spanning tree of a connected, undirected graph is a subgraph that contains
  - All the vertices in the graph
  - And the fewest number of edges such that the subgraph is connected

- Use DFS (or BFS) and keep track of the edges crossed.
  - The set of crossed edges form a spanning tree.

- Minimum Spanning Trees
  - A Minimum Spanning Trees of a connected, weighted, undirected graph is a spanning tree for the graph such that the sum of the edge wrights is the minimum of the all possible spanning trees for the graph.

- **How to construct a MST?**
  - Use DFS and keep track of the edges crossed.
  - How to choose which neighbor to visit?
    - Choose the one with the smallest weight. (A greedy choice)
    - A “greedy choice” will get you close to an optimal solution
  - Greedy algorithms are those that apply the following problem-solving heuristic when faced with choices
    - Make locally optimal choices in hopes of arriving at a globally optimal solution
    - A greedy heuristic isn’t always applicable and, even when it is, it isn’t guaranteed to find an optimal solution
  - Given any division of the vertices of a graph into two sets, the minimum spanning tree contains the minimum cost edge that connects a vertex in one set a vertex in the other set.

- **Prim’s Algorithm**
  - Use Prim’s Algorithm to find MST

- **Kruskal’s Algorithm**
  - Select the edges in ascending order of weight. Add each edge to the MST unless it would create a cycle. Stop when n-1 have been added.