* Cycle property:
  + Let C be any cycle in a graph, and let f be the max-cost edge belonging to C. Then the MST does not contain f.
* Cut property:
  + Let S be any subset of vertices, and e be the n=min-cost edge with exactly one endpoint in S. Then the MST surely contains e.
* Kruskal’s Algorithm to compute MST:
  + Steps:
    1. Sort all edges in ascending order of cost (weight).
    2. Add the next edge of the sorted list into a set T, unless doing so would create a cycle in T.
    3. T will be the MST once T contains all vertices of (|V| - 1) edges.
* Proving that T is the minimum spanning tree: Probably on next exam
  + Kruskal’s correctness proof.
  + Proposition: Kruskal’s algorithm computes the MST.
  + Proof [case 1]: Suppose that adding e to T creates a cycle C:
    - (e is the edge that you are currently on, in the sorted list of edges)
    - e is the max-cost edge in cycle C.
    - e is not in the MST (due to cycle property)
  + Proof [case 2]: suppose adding e=(v, w) to T does not create a cycle in T.
    - Let S be the vertices in v’s connected component.
    - w is not in S. (true)
    - e is the min-cost edge with exactly one endpoint in S.
    - e is in MST, due to the Cut property.