We will cover two algorithms to construct minimum spanning tree for a weighted graph. Try to recall the definition of a tree: A tree is a connected acyclic graph.

The first algorithm: Prim’s algorithm is copied from our textbook. It starts from one vertex, trying to expand the tree (make the constructed part connected) from the source vertex without creating a cycle.

The second algorithm: Kruskal’s algorithm, works in another direction. It examines all the edges from the least weight, trying to add the least weight edges without creating a cycle.

The algorithm can be presented as follows:

Input: $G, w$
Output: Minimum Spanning Tree for $G$
$A = \phi$
for each vertex $v$ of $G$
    Make-Set(v)
Sort the edges of $E$ based on the weight
for each $e = (u, v)$ in decreasing order
    if $\text{find}(u) \neq \text{find}(v)$ then
        $A = A \cup \{(u, v)\}$
return $A$

Algorithm 1 Kruskal Algorithm

We will walk through the algorithm on the same graph used in Prim’s algorithm.