Key Lemma:

- Given a subset of edges \( F \), suppose \( F \) is a subset of edges of a minimum spanning tree \( T \). Pick any cut \((S, \overline{S})\) that does not intersect with \( F \), let \( e \) be an edge with minimum cost in this cut \((S, \overline{S})\) then \( F \cup \{e\} \) is a subset of edges in a minimum spanning tree \( T' \).

\((T'\) may not be equal to \( T \))

Proof: in case 1, \( e \) is actually an edge in \( T \), this case is easy because \( F \cup \{e\} \subseteq T \), can choose \( T' = T \).

in case 2, \( e \) is not an edge in \( T \).

adding \( e \) to \( T \) will form a cycle, call it \( C \).

we know cycle \( C \) must intersect with cut \((S, \overline{S})\).

in an even number of edges.

so there must be another edge \( e' \in C \), \( e' \) also crosses cut \((S, \overline{S})\).

we will swap \( e \) and \( e' \).

define \( T' = (T \setminus \{e\}) \cup \{e'\} \).

\[
\text{cost}(T') = \text{cost}(T) - w(e') + w(e)
\]

\[
\leq \text{cost}(T), \quad \text{by assumption } w(e') \leq w(e)
\]
this means $T'$ is a MST, also, $F \cup \{e\} \subseteq T'$, this concludes the proof.

- Proof of general algorithm

  . Induction Hypothesis: At iteration $i$, the edges selected by the alg is a subset of some MST.
  
  - Base case: $i = 0$, set of edges selected is empty
  
  - Induction step: Key lemma

- Prim's algorithm

```
3 3
1 4

2 5

3
```

- in implementation maintain $\text{dist}(u)$

  $\text{dist}(u)$: minimum cost of an edge $(u, v)$ where $v$ is a visited vertex
  (only maintained if $u$ has not been visited)

- Kruskal's algorithm

```
3 3
1 4

2 5

3
```

Lectures Page 2
- Proof of Kruskal’s algorithm.

  main idea: Know general MST is correct.
  if show Kruskal is a special case of general MST
  then Kruskal must also be correct

going to show: every time Kruskal’s algorithm adds an edge (u,v), can find a cut (S, \overline{S}) where u \in S, v \in \overline{S}
the cut does not contain any previous edges, and (u,v)
is the min cost edge in (S, \overline{S})

Proof: when Kruskal adds an edge (u,v)
let S be the set of vertices that are connected to u using edges already selected by Kruskal.
by design of Kruskal, u \notin S
by design of Kruskal, \( u \not\in S \)
only need to prove \((u, v)\) is the min cost edge between \((S, \overline{S})\)
assume towards contradiction that there is an edge \(e\)
that crosses the cut \((S, \overline{S})\)
\[ w(e) < w(u, v) \]
by Kruskal, edge \(e\) is going to be considered before \((u, v)\)
when edge \(e\) is considered, \((S, \overline{S})\) were not connected,
so edge \(e\) cannot create a cycle, Kruskal must have
selected edge \(e\). This is a contradiction.

- running time

  Prim: Naïve \( O(n^2) \)

  Fibonacci Heap \( O(m+n\log n) \)

  Kruskal: \( O(m\log n) \)