Generic MST Algorithm

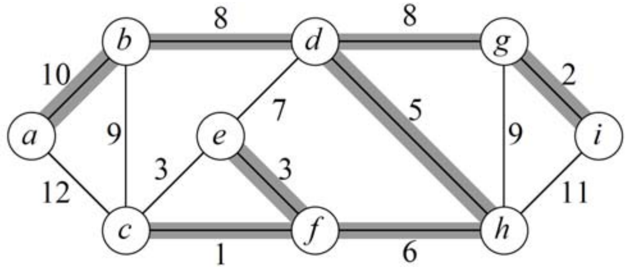
```
GENERIC-MST(G, w)
A = Ø
while A does not form a spanning tree
    find an edge (u, v) that is safe for A
    A = A ∪ {(u, v)}
return A
```

Use loop invariant to show that the generic algorithm works.

- Initialization: The empty set trivially satisfies the loop invariant.
- Maintenance: Since we add only safe edges, *A* remains a subset of some MST.
- Termination: All edges added to *A* are in an MST, so when we stop, *A* is a spanning tree that is also an MST.

Finding a Safe Edge (1/4)

- How do we find safe edges?
- Let's look at the example. Edge (c, f) has the lowest weight of any edge in the graph. Is it safe for $A = \emptyset$?



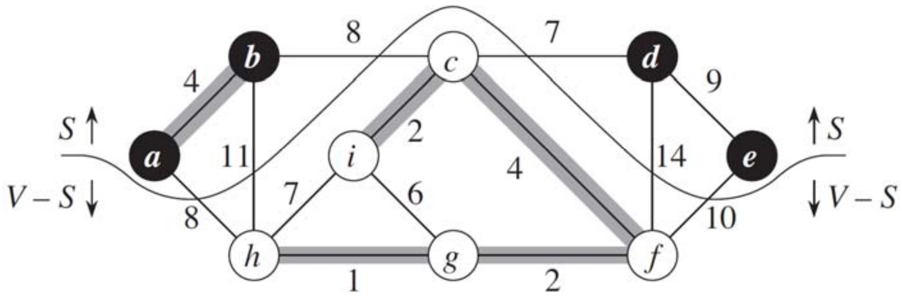
Finding a Safe Edge (2/4)

- Intuitively: Let $S \subset V$ be any set of vertices that includes c but not f (so that f is in $V - S$). In any MST, there has to be one edge (at least) that connects S with $V - S$. Why not choose the edge with minimum weight? (Which would be (c, f) in this case.)

Finding a Safe Edge (3/4)

- Some definitions: Let $S \subset V$ and $A \subseteq E$.
  - A cut $(S, V - S)$ is a partition of vertices into disjoint sets S and $V - S$.
  - Edge $(u, v) \in E$ crosses cut $(S, V - S)$ if one endpoint is in S and the other is in $V - S$.
  - A cut respects *A* if and only if no edge in *A* crosses the cut.
  - An edge is a light edge crossing a cut if and only if its weight is minimum over all edges crossing the cut. For a given cut, there can be > 1 light edge crossing it.

Finding a Safe Edge (4/4)



Rule for Recognizing Safe Edges (1/7)

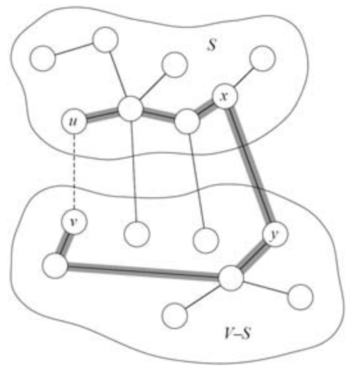
- **Theorem 23.1** Let $G = (V, E)$ be a connected, undirected graph with a real-valued weight function w defined on E. Let A be a subset of E that is included in some minimum spanning tree for G, let $(S, V - S)$ be any cut of G that respects A, and let (u, v) be a light edge crossing $(S, V - S)$. Then, edge (u, v) is safe for A.
- **Proof:**
  Let T be an MST that includes A.
  If T contains (u, v), done.

Rule for Recognizing Safe Edges (2/7)

So now assume that T does not contain (u, v). We'll construct a different MST T' that includes $A \cup \{(u, v)\}$.

Recall: a tree has unique path between each pair of vertices. Since T is an MST, it contains a unique path p between u and v. Path p must cross the cut $(S, V - S)$ at least once. Let (x, y) be an edge of p that crosses the cut. From how we chose (u, v), must have $w(u, v) \leq w(x, y)$.

Rule for Recognizing Safe Edges (3/7)



Except for the dashed edge (u, v), all edges shown are in T. A is some subset of the edges of T, but A cannot contain any edges that cross the cut $(S, V - S)$, since this cut respects A. Shaded edges are the path p.