NOTE: I use set notation "$\{u,v\}$" to represent undirected edges.

(a) There is **no** solution for the input $(G,w,L)$ with $G=(V,E)$ where $V=\{a,b,c\}$, $E=\{\{a,b\},\{b,c\}\}$, $w(a,b)=w(b,c)=1$ and $L=\{b\}$.

(b) *Main idea:* For each required leaf, use a minimum-weight edge to connect it to the rest of the spanning tree, then complete the rest of the spanning tree using any algorithm. (Alternatively, we could begin by finding a MST on the graph without all of the leaves, then connect all of the leaves back using a minimum-weight edge for each one.) **One "trap" to watch out for: each leaf must be connected to some non-leaf.**

MST($G=(V,E),w,L$):
  $T \leftarrow \varnothing$
  # Use a minimum-weight edge for each leaf.
  **for** $v \in L$:
    let $\{u,v\}$ be a minimum-weight edge containing $v$ **with $u \notin L$**
    $T \leftarrow T \cup \{\{u,v\}\}$
    remove all edges that contain $v$ (including $\{u,v\}$) from $E$
  Run Kruskal's algorithm on the remaining graph, starting with the current value of $T$, to finish the spanning tree.

(c) We know Kruskal's algorithm finds a MST on the portion of the graph without the leaves. Moreover, for each leaf, using any edge other than a minimum-weight edge yields a spanning tree whose weight is not minimum.