Fundamental Algorithms, Section 003
Homework 12, Additional Problems, Fall 22.

1. Let $G = (V, E)$ be an undirected weighted graph and let T be a minimum spanning tree of G. Suppose edge e in T is removed from T and G. Give a linear time algorithm to compute a spanning tree for the resulting graph $G^e = (V, E \setminus \{e\})$. You may assume that G^e is connected. Explain why your algorithm works and justify your runtime.

2. Let $G = (V, E)$ be an undirected graph with non-negative edges weights. Suppose you are given an MST T for G, and an array Pred$[1 : n]$ of the predecessors of each vertex $v \in V$ on a shortest path from a designated vertex $s \in V$ to v. Now suppose every edge length in increased by 1. Is T always an MST for the resulting graph? Does the array Pred always remain change? Justify your answers. i.e. if the answer is "yes" explain why; if the answer is "no", provide an example demonstrating this.