CSC148 - Tree Deletion Algorithms

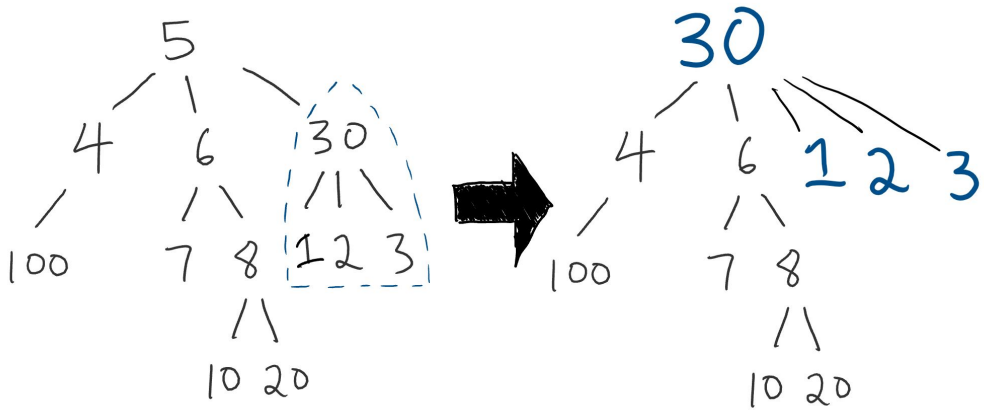
Now that we've explored several non-mutating tree methods, we'll look at one of the two core mutating operations on trees: deleting an element from the tree. The main work for deletion is usually done in deleting the *root* of a tree:

```
def delete_root(self) -> None:
    """Remove the root of this tree. Precondition: this tree is not empty."""
```

On this worksheet, we'll look at two different ways we can delete the root of a tree. For each strategy, we'll only describe the overall idea and an example; it'll be up to you and your group to plan and then implement the required behaviours.

Strategy 1: "Promoting" a subtree

Idea: to delete the root, take the rightmost subtree t_1, and make the root of t_1 the new root of the full tree, and make the subtrees of t_1 become subtrees of the full tree.



Method 2: Replace the root with the leftmost leaf

Idea: to delete the root, find the leftmost *leaf* of the tree, and move that leaf so that it becomes the new root value. (No other values in the tree should move.)