


```
    } else if (isEmpty(tree.left)) {
      tree = tree.right;
    } else if (isEmpty(tree.right)) {
      tree = tree.left;
    } else {
      const [side, other] =
        tree.left.priority < tree.right.priority
          ? ["right", "left"]
          : ["left", "right"];
      tree = _rotate(tree, side);
      tree = remove(tree, keyToRemove);
    }

    return tree;
  };
```

14.18 Trees as Heaps

What would happen if you used binary search trees to represent heaps? What would the performance be of the three basic operations: `add()`, `remove()`, and `top()`? Can you think of ways to make `top()` faster?