1 AVL tree: insert

Consider the following AVL tree.

```
           4
        /     \
      1         10
       \       /   \
        3     7      20
```
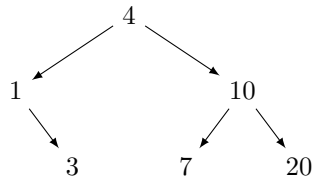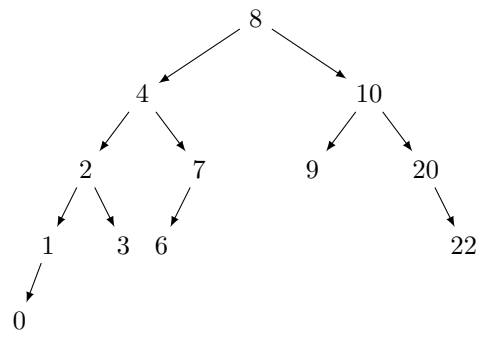
1. Give an example sequence of `insert` operations that, starting with an empty AVL tree, could have resulted in the above tree.

2. Insert 8. Show the process and the resulting tree.

3. Insert 9 (into the tree resulting from inserting 8 above). Show the process and the resulting tree.

4. Insert 6 (into the tree resulting from inserting 9 above). Show the process and the resulting tree.

5. Insert 2 (into the tree resulting from inserting 6 above). Show the process and the resulting tree.

6. Finally, insert 22.

2 AVL tree: delete

We begin with the tree we constructed in the previous question.

```
              8
           /     \
          4        10
         /  \     /   \
        2    7   9     20
       / \   /            \
      1   3 6              22
```
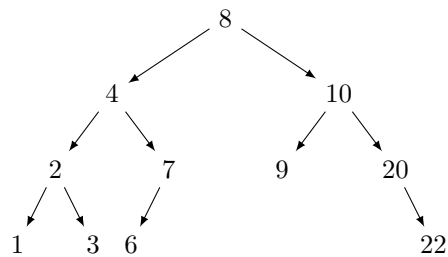
1. Delete 9.
2. Delete 8.
3. Delete 20.
4. Now let's go back to the original tree, but with an added 0:

```
                    8
                 ↙     ↘
               4         10
             ↙   ↘      ↙   ↘
            2     7    9     20
          ↙  ↘   ↙            ↘
         1    3 6              22
        ↙
       0
```

Let's delete 8.