Exercise 1. Let us consider the following binary search tree over the nodes {A, E, M, N, S, T} using the alphabetical order. In case it is not a binary 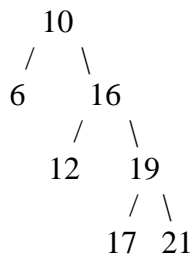search tree, transform it into an "equivalent" (with the same nodes, but not necessarily edges) well-balanced binary search tree using the AVL algorithm.

```
        S
       / \
      E   T
     / \
    A   N
         /
        M
```
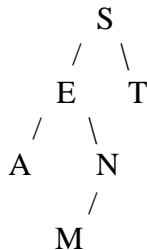
Exercise 2. Let us consider the following binary search tree with nodes over the set of positive integers using the increasing order. In case it is not a binary search tree, transform it into an "equivalent" (with the same nodes, but not necessarily edges) well-balanced binary search tree using the AVL algorithm.

```
        10
       /  \
      6    14
     / \
    3   8
   / \
  1   5
```
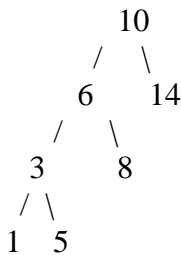
Exercise 3. Let us consider the following binary search tree with nodes over the set of positive integers using the increasing order. In case it is not a binary search tree, transform it into an "equivalent" (with the same nodes, but not necessarily edges) well-balanced binary search tree using the AVL algorithm.

```
        10
       /   \
      6     16
           /  \
         12    19
               / \
             17   21
```

Exercise 4. Let us consider the following binary search tree with nodes over the set of positive integers using the increasing order. In case it is not a binary search tree, transform it into an "equivalent" (with the same nodes, but not necessarily edges) well-balanced binary search tree using the AVL algorithm.

```
        10
       /    \
     6      18
           /    \
         14      21
        /  \
      12   15
```