Binary Search Tree

2 min

Constraints are placed on the data or node arrangement of a tree to solve difficult problems like efficient search.

A *binary tree* is a type of tree where each parent can have **no more than two children**, known as the *left child* and *right child*.

Further constraints make a *binary search tree*:

- Left child values must be lesser than their parent.
- Right child values must be greater than their parent.

The constraints of a binary search tree allow us to search the tree efficiently. At each node, we can discard **half** of the remaining possible values!

Let's walk through locating the value 31.

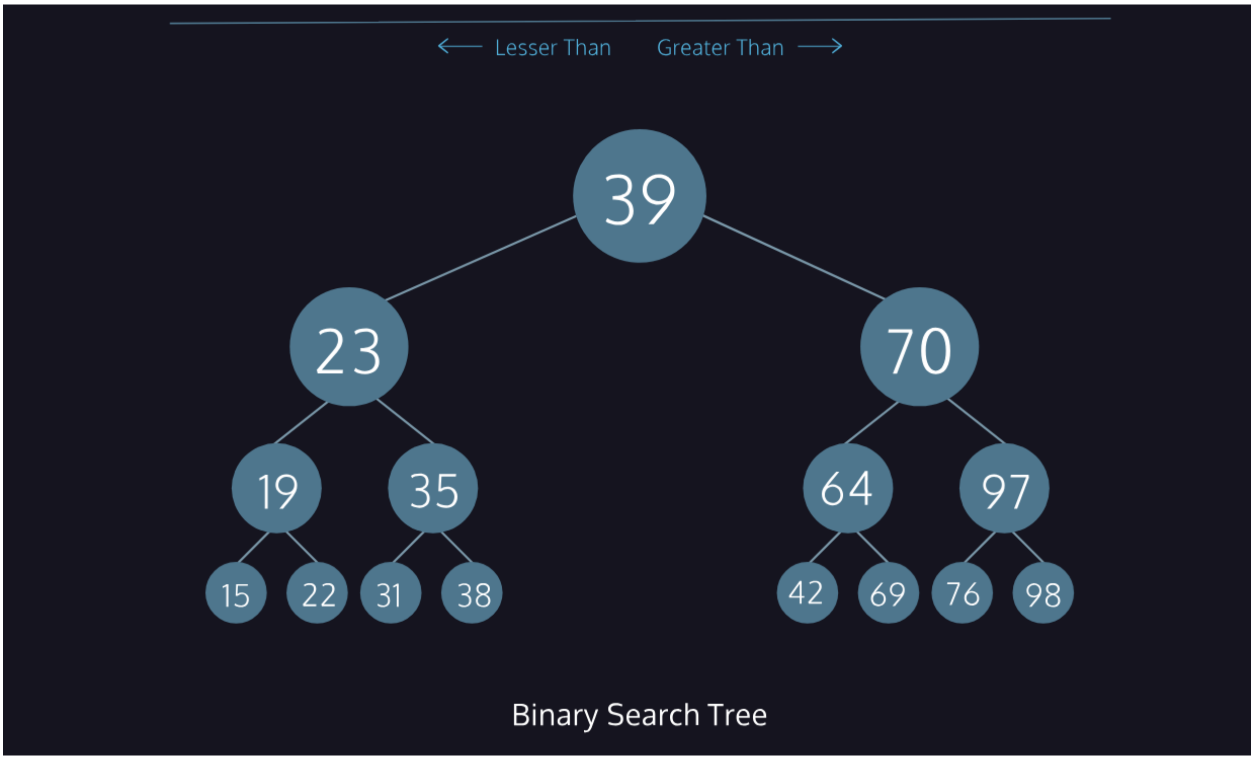
1. Start at the root: 39
2. 31 < 39, we move to the left child: 23
3. 23 < 31, we move to the right child: 35
4. 31 < 35, we move to the left child: 31
5. We found the value 31!

In a dataset of **fifteen** elements, we only made **three** comparisons. What a deal!

Instructions

From the root, follow a node's left or right child to find the following numbers: 22, 42, 97.

How many steps did each number take?



Binary Search Tree