Trees:

1. Inorder and preorder traversal
   1. How bifurcation works?

We actually take root from preorder and indexes from inorder, so for the next recursion consider indexes from the inorder only.

1. Validate BST:

Problem is how to validate, you need to know min and max value.

Since it is just True or False you can start from top to bottom, the root should be between -inf to +inf and then traverse and define range for each, if any of the left or right is False we can say whether it is valid or not

1. Problem comes when we have to identify a subtree:

In that case we cannot move from bottom to top instead we have to move from bottom to root, but How?

In this case you return the output from recursion and use it.

1. Inorder of BST is sorted array.
2. Serialize and Deserialize, use preorder its easy
3. Pass by value and pass by reference:
   1. Pass by value

Let’s say you are doing traversal and passing an index, as direct value, if you want it as output, it will not update, instead you have to pass it by reference.

Local variable : will be available only inside function

and Global variable ; will be available for whole script

in case you want to pass a variable as value use global variable like self.root\_idx

What’s Pass by Value?

* Pass by value is like giving someone a copy of your toy. They can play with the copy, but your original toy stays safe.
* In programming, when you pass a variable to a function, the function gets a copy of the value. Changes to the copy don’t affect the original.
* Python uses pass by value for immutable types like integers, floats, strings, and tuples. You can’t change the original value, so the function works on a copy.

What’s Pass by Reference?

* Pass by reference is like giving someone your actual toy box. They can change the toy inside, and since it’s the same box, your toy changes too.
* In programming, the function gets the memory address of the variable, not a copy. Changes in the function affect the original.
* List, Dictionaries