

* we looked at the post order and see that 8 is the last number, which indicates that 8 is the root.
* In the inOrder, after 8 is 4, 3, 11. Which means 4, 3, 11 are on the right subtree and the rest are in the left sub tree.
* After 8, 4 is next, in the inorder, and post order is 4, then 8 which means 4 is the child of 8.
* Next is 3 for inorder but 3 is before 11 in the post order.
  + you would assume that 8 goes to 4 then 3 then 11, which is correct in in order but incorrect in post order
  + post order it would be 11, 3, 4, 8 but its supposed to be 3, 11, 4, 8
* this would mean that 3 would be the left node child of 11 to satisfy this case.
* Next 9 begins both of the inorder and post order, which means it’s a leaf node on the left most subtree.
* 5 is the parent of 9 and the child of 8.
  + We know this because after 9 in the inorder, 5 is next which represents the parent of 5
  + Also, 5 is the last number of the left half of the post order that will be on the left subtree (because next one is 3, 11, 4, 8) so we know that for post order, 5 will be the child of 8, the root.
* Next we work with 7 and 1.
* In the in order, it goes 1 then 7. So we imply that 1 is the child of 5 and 7 is the child of 1. But that’s not the case.
* In the post order, 1 is before 7. Which means just like 3 and 11, 1 has to be left child of 7 in order to satisfy the inorder and post order.
* 2 will be the child node of 12.
* 2 is the left child of 12 so it satisfy both inorder and post order.
* Lastly we connect the 12 as the right side child of 7 because in the post order after 12, itll go to 7. And in the inorder, after 7 itll go to 2 which is the 12’s left child.
* After connecting 12 to 7 as the child node. You have successfully reconstructed the binary tree using inorder and post order sequence of numbers.