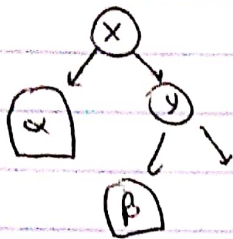


- ③ the runtime of the worst and best case is $O(n \log(n))$ because everytime you insert a node into the tree, it will run $\log(n)$ times and we will do that action n times, so the runtime will be $n \log(n)$
- ④ best case - no rotation - tree remaining balanced cause you choose what to insert

Worst case - every other insert needs a rotation.

- ④ after correction, the tree will look like this:

④



$$S_y = S_p + S_r + 1$$

$$S_x = S_a + S_y + 1$$

- ⑤
- ```

node* rotateLR (node* p)
{
 node* n1 = p
 node* n2 = n1 -> left
 node* n3 = n2 -> right
 n2 -> right = n3 -> left
 if (n2 -> right)
 n2 -> right -> parent = n2
 n1 -> left = n3 -> right
 if (n1 -> left)
 n1 -> left -> parent = n1
 n3 -> right = n1
 n1 -> parent = n3
 n3 -> left = n2
 n2 -> parent = n3
 return n3
}

```

- ⑥ no, because we know that the height of an AVL tree is  $O(\log n)$ , therefore it can't be  $\omega(\log n)$  because  $O$  means less than infinity and  $\omega$  means infinity, so there is a contradiction.