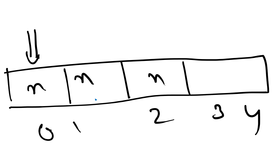
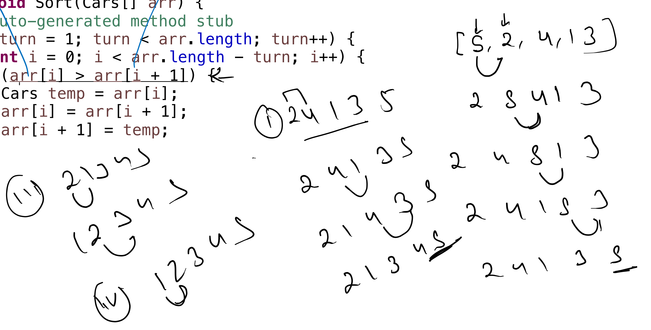
Lec-35

27/07/24 3:43 PM

```
Cars[] ar = new Cars[5];
```



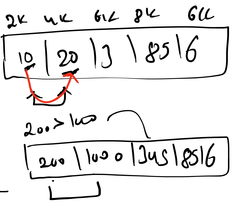
```
public static void Sort(Cars[] arr) {
    // TODO Auto-generated method stub
    for (int turn = 1; turn < arr.length; turn++) {
        for (int i = 0; i < arr.length - turn; i++) {
            if (arr[i] > arr[i + 1]) {
                Cars temp = arr[i];
                arr[i] = arr[i + 1];
                arr[i + 1] = temp;
            }
        }
    }
}
```

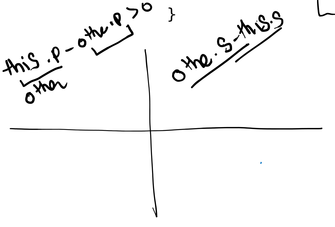


```
ar[0] = new Cars(200, 10, "White");
ar[1] = new Cars(1000, 20, "Black");
ar[2] = new Cars(345, 3, "Yellow");
ar[3] = new Cars(34, 89, "Grey");
ar[4] = new Cars(8907, 6, "Red");
```

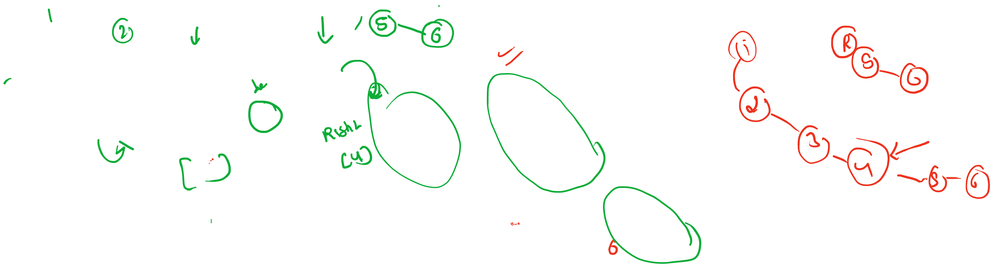
```
@Override
public int compareTo(Cars o) {
    // TODO Auto-generated method stub
    return 0;
}
```

```
public static <T extends Comparable<T>> void Sort(T[] arr) {
    // TODO Auto-generated method stub
    for (int turn = 1; turn < arr.length; turn++) {
        for (int i = 0; i < arr.length - turn; i++) {
            if (arr[i].compareTo(arr[i+1]) > 0) {
                T temp = arr[i];
                arr[i] = arr[i + 1];
                arr[i + 1] = temp;
            }
        }
    }
```







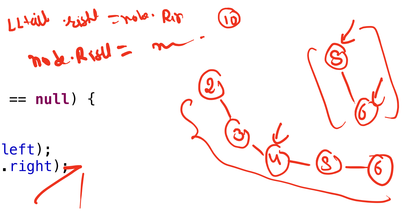


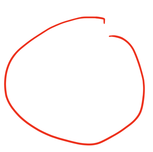




```
public TreeNode MakeLL(TreeNode root) {
        if (root == null) {
            return null;
        }
        if (root.left == null && root.right == null) {
            return root;
        }
        TreeNode leftll_Tail = MakeLL(root.left);
        TreeNode rightll_Tail = MakeLL(root.right);

    }
```





```
public TreeNode MakeLL(TreeNode root) {
        if (root == null) {
            return null;
        }
        if (root.left == null && root.right == null) {
            return root;
        }
        TreeNode leftll_Tail = MakeLL(root.left);
        TreeNode rightll_Tail = MakeLL(root.right);

        leftll_Tail.right = root.right;
        root.right = root.left;
        root.left = null;

    }
```