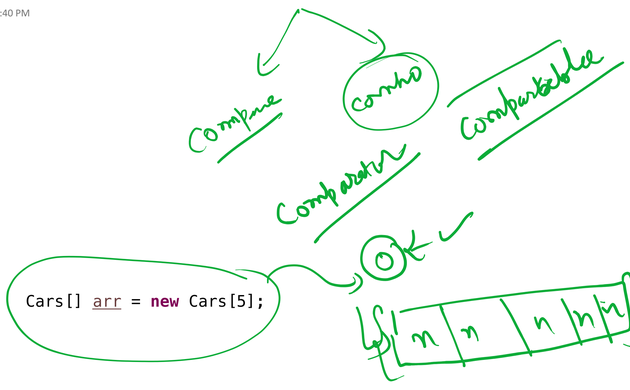
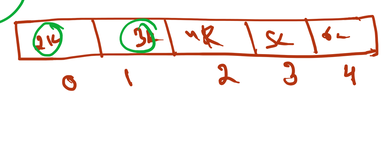
Lec-40

19/12/24 8:40 PM



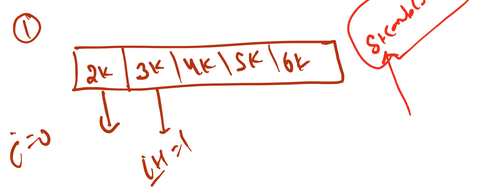

```
public static <T>void sort(T []arr) {
    for (int trun = 1; trun < arr.length; trun++) {
        for (int i = 0; i <arr.length-trun ; i++) {
            if(arr[i]>arr[i+1]) {
                T temp=arr[i];
                arr[i]=arr[i+1];
                arr[i+1]=temp;
            }
        }
    }

}
```

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

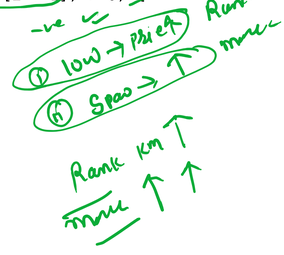


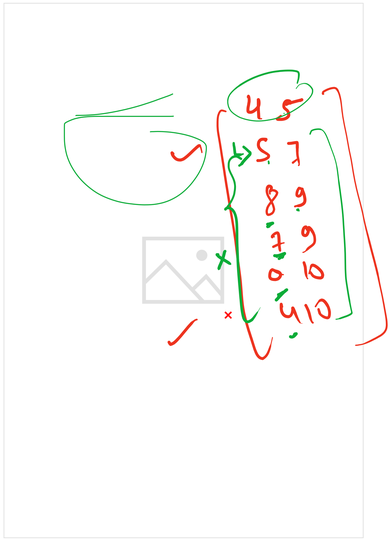

```
@Override
    public int compareTo(Cars o) {
        // TODO Auto-generated method stub
        return 0;
    }
```

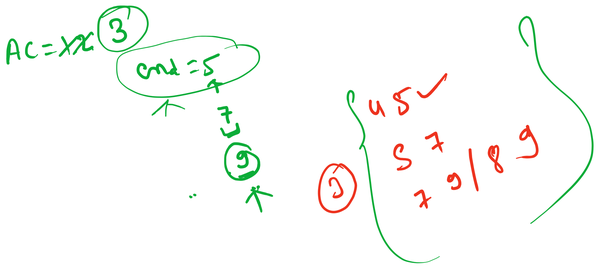
```
arr[0] = new Cars(200, 10, "White");// P S C
arr[1] = new Cars(1000, 20, "Black");
arr[2] = new Cars(345, 3, "Yellow");
arr[3] = new Cars(34, 89, "Grey");
arr[4] = new Cars(8907, 6, "Red");
```



```
public static <T extends Comparable<T>> void sort(T[] arr) {
    for (int trun = 1; trun < arr.length; trun++) {
        for (int i = 0; i < arr.length - trun; i++) {
            if (arr[i].compareTo(arr[i + 1]) > 0) {
                T temp = arr[i];
                arr[i] = arr[i + 1];
                arr[i + 1] = temp;
            }
        }
    }
}
```








```
Arrays.sort(arr, new Comparator<Pair>() {
            @Override
            public int compare(Pair o1, Pair o2) {
                return o1.et - o2.et;
            }

        });
        int activitie = 1;
        int end = arr[0].et;
        for (int i = 1; i < arr.length; i++) {
            if (arr[i].st >= end) {
                activitie++;
                end = arr[i].et;
            }
        }
        System.out.println(activitie);
```