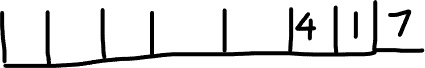
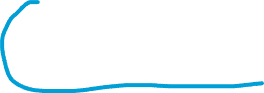
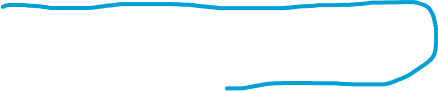
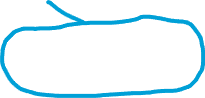
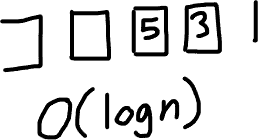
The splitting process takes on a tree

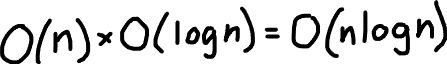
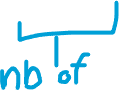
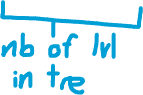
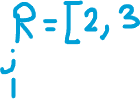
Shape time complexity is O(logn)



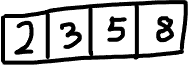
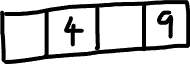
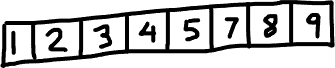
Put pointer on each subarray, compare their first element to find smallest, put the element in R. move pointer of the smallest to next element, compare again….

Once a pointer reaches the end of its subarray, put the remaining elements

of the other subarray in R since its already sorted



The number in blue represents the number of add operation performed for a comparison. So we notice it will always give 8 the total number of elements in the original array



For a single level in the constructed tree, we need to add in total n elements to temporary arrays before moving to the next level.