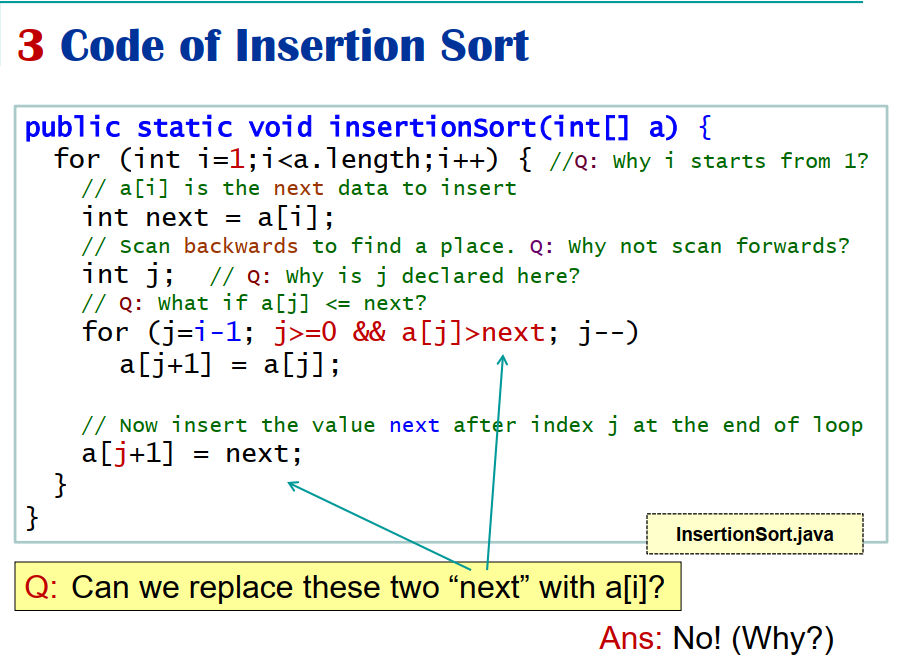
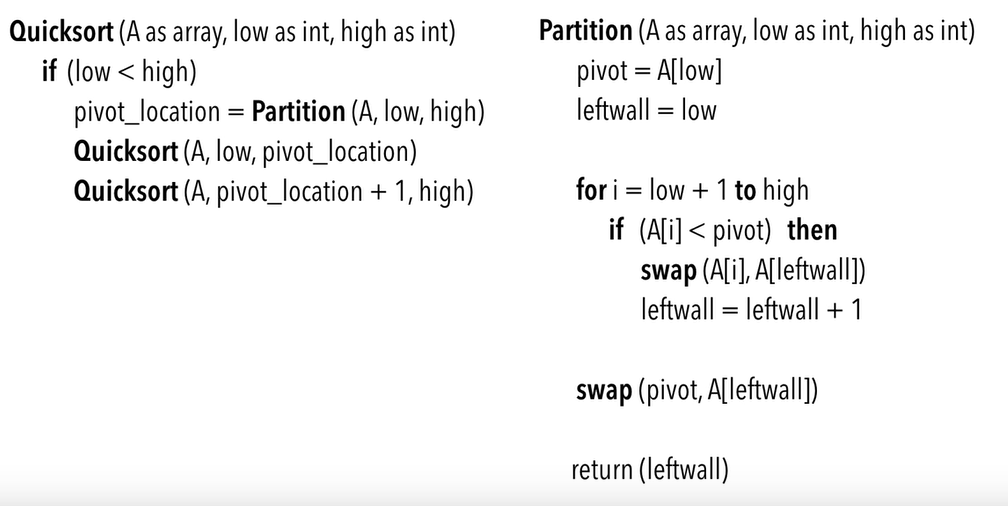
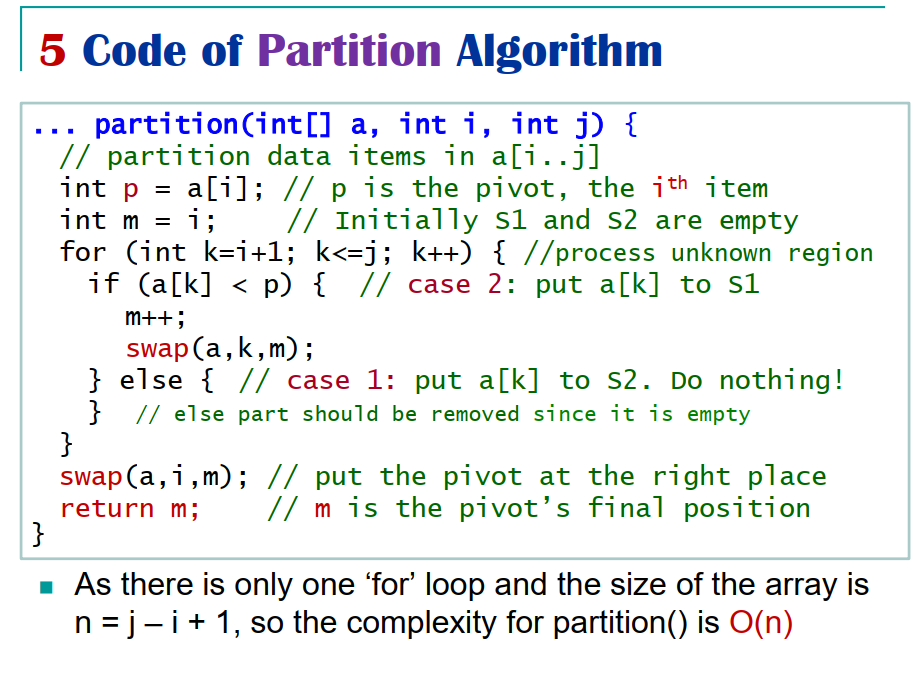


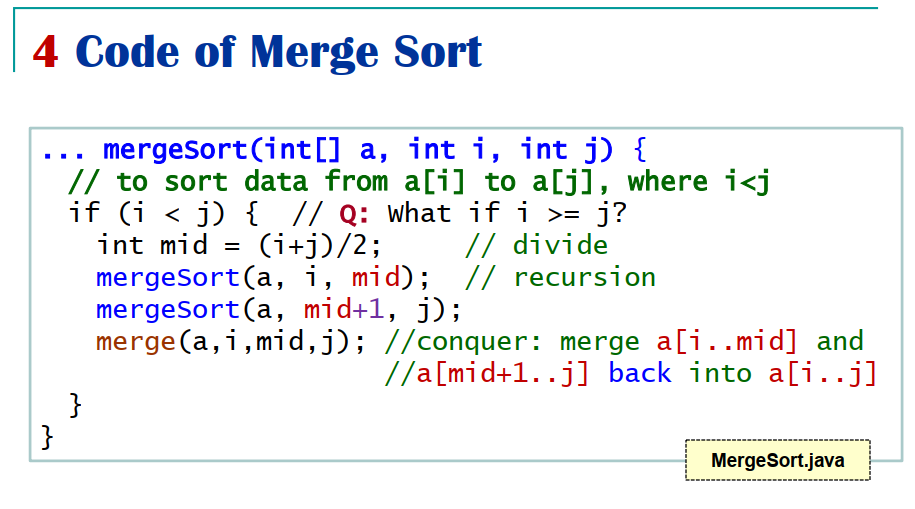
**Find largest element, shift it to last, repeat until array ends**

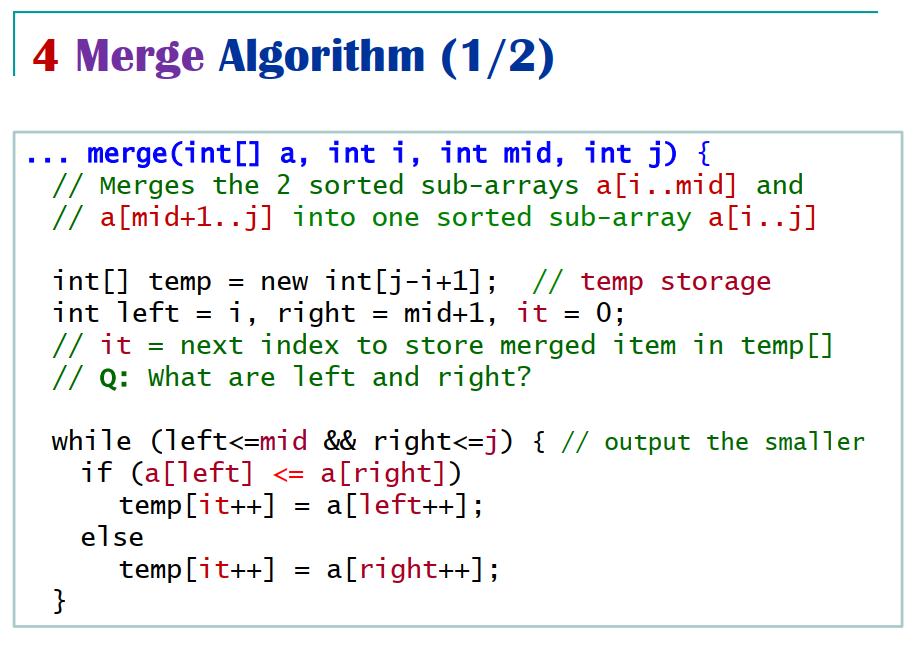


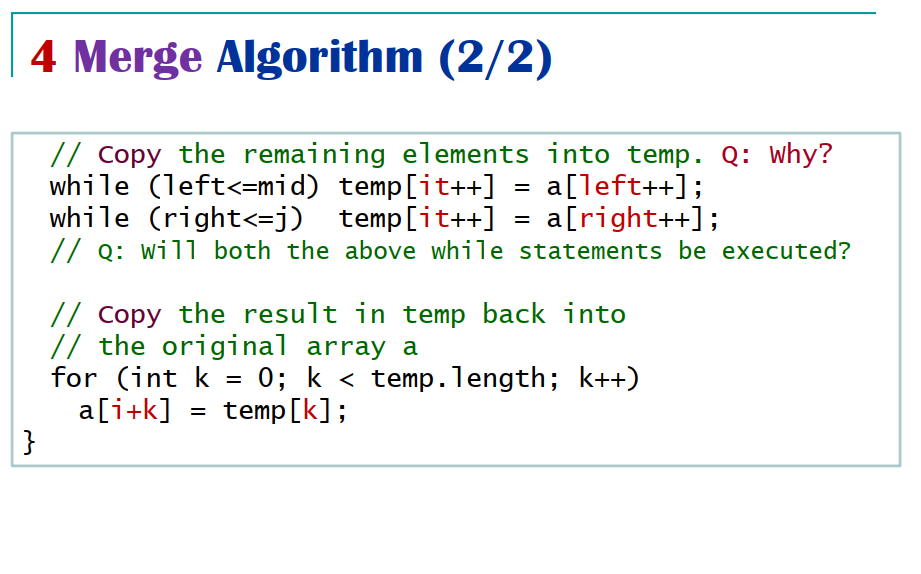
* **Find a non sorted element, check if its smaller than bigger than arr[pos-1]. If not, keep swapping with arr[pos-1] until the left element is smaller, it is then sorted. Move on till last element is reached**
* **Good for almost sorted arrays**

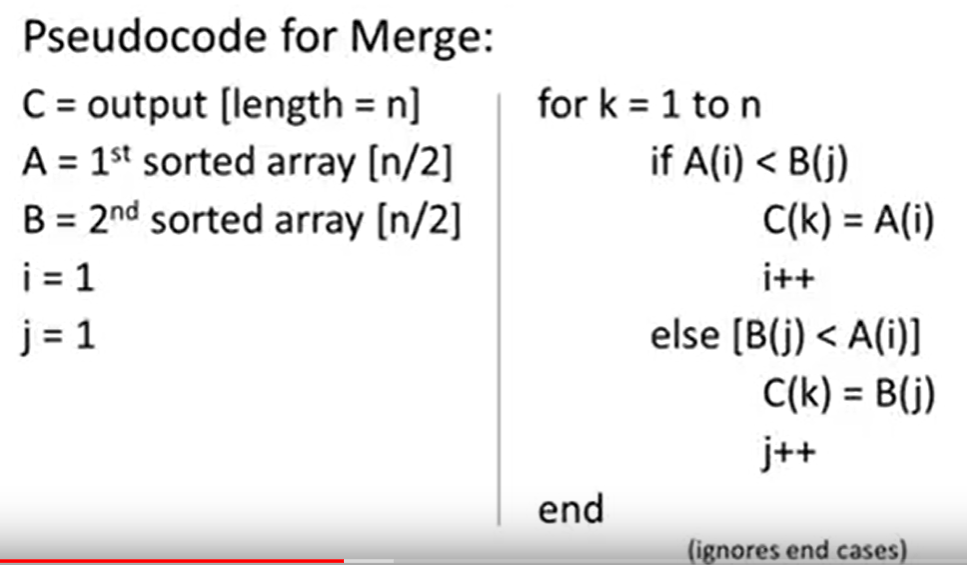
**Quick Sort**

* Sort first, last and middle elements
* Choose middle element as pivot (high chance)
* Ove pivot to the end
* Keep swapping to get the bigger elements on one side and smaller elements on the other, until you can fit pivot in the centre
* 
* 

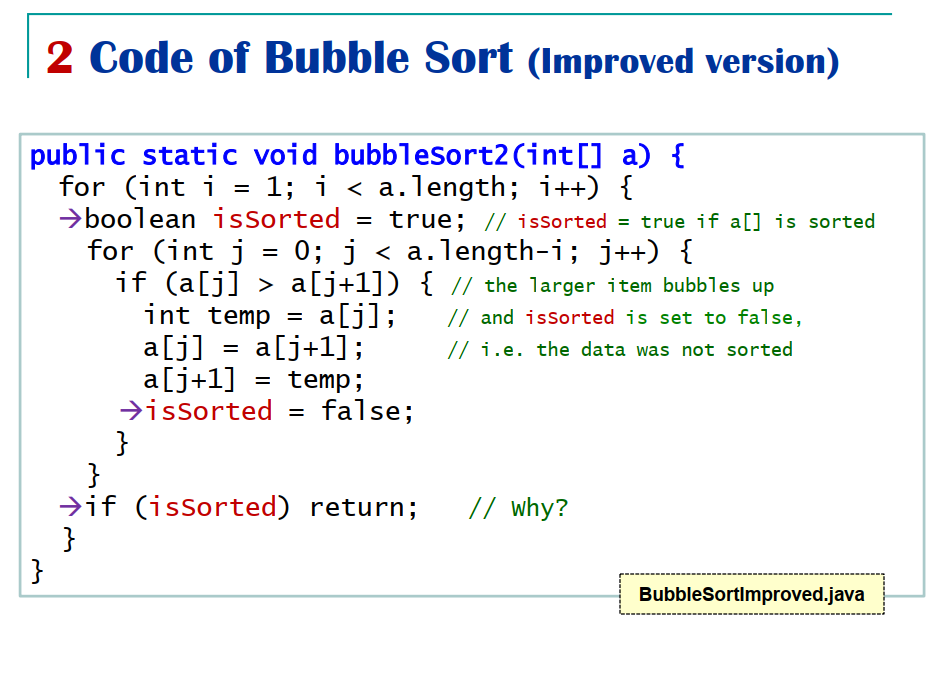








* **Merge is very good for large data sets (recursive), and the best overall**
* **Splits until groups of two, compare each group and select till all in order**
* **Use insertion for smaller data sets**



**Bubble sort can potentially be up to O(N) in best case.**

**Stable sorting and safe.**

**Radix Sort**

* Sorting and group based on characters
* Good for strings (ascii comparisons) and long digit numbers
* Non fixed
* Sorts in O(n) time

