#5: The Algorithm given belongs to the linear complexity level O(N)

#10: A selection sort algorithm will sort an array by repeatedly finding the Minimum element from the unsorted part and putting it at the beginning of the array. The following is the state of the array after each outer pass of the selection sort

First array: int[] numbers3 = {8, 5, –9, 14, 0, –1, –7, 3};

{8, 5, -9, 14, 0, -1, -7, 3} 1st **Original Array**

{-9, 5, 8, 14, 0, -1, -7, 3} 2nd **Search for the smaller element from 0 index to 7th index and swap the smaller value to the 0 index 8 <-> -9**

{-9, -7, 8, 14, 0, -1, 5, 3} 3rd **Search for the smaller element from 1st index to 7th index and swap the smaller value to the 1st index 5 <-> -7**

{-9, -7, -1, 14, 0, 8, 5, 3} 4th **Search for the smaller element from 2nd index to 7th index and swap the smaller value to the 2nd index 8 <-> -1**

{-9, -7, -1, 0, 14, 8, 5, 3} 5th **Search for the smaller element from 3rd index to 7th index and swap the smaller value to the 3rd index 14 <-> 0**

{-9, -7, -1, 0, 3, 8, 5, 14} 6th **Search for the smaller element from 4th index to 7th index and swap the smaller value to the 4th index 14 <-> 3**

{-9, -7, -1, 0, 3, 5, 8, 14} 7th **Search for the smaller element from 5th index to 7th index and swap the smaller value to the 5th index 8 <-> 5**

{-9, -7, -1, 0, 3, 5, 8, 14} **Final sorted array**

Second array: int[] numbers4 = {15, 56, 24, 5, 39, -4, 27, 10};

{15, 56, 24, 5, 39, -4, 27, 10} 1st **Original Array**

{-4, 56, 24, 5, 39, 15, 27, 10} 2nd **Search for the smaller element from 0 index to 7th index and swap smaller value to the 0 index** **15 <-> -4**

{-4, 5, 24, 56, 39, 15, 27, 10} 3rd **Search for the smaller element from 1st index to 7th index and swap the smaller value to the 1st index 56 <-> 5**

{-4, 5, 10, 56, 39, 15, 27, 24} 4th **Search for the smaller element from 2nd index to 7th index and swap the smaller value to the 2nd index 24 <-> 10**

{-4, 5, 10, 15, 39, 56, 27, 24} 5th **Search for the smaller element from 3rd index to 7th index and swap the smaller value to the 3rd index 56 <-> 15**

{-4, 5, 10, 15, 24, 56, 27, 39} 6th **Search for the smaller element from 4th index to 7th index and swap the smaller value to the 4th index 39 <-> 24**

{-4, 5, 10, 15, 24, 27, 56, 39} 7th **Search for the smaller element from 5th index to 7th index and swap the smaller value to the 5th index 56 <-> 27**

{-4, 5, 10, 15, 24, 27, 39, 56} 8th **Search for the smaller element from 6th index to 7th index and swap the smaller value to the 6th index 56 <-> 39**

{-4, 5, 10, 15, 24, 27, 39, 56} **Final sorted array**