Merge Sort Improvements

1. ***Merge Sort Improvements:*** Implement the three improvements to merge sort that are described as follows: Add a cutoff for small subarrays, test whether the array is already in order, and avoid the copy by switching arguments in the recursive code.

Use **insertion sort** for small subarrays. (If the subarray size is 7).

**Test whether the array is already in order**. We can reduce the running time to be linear for arrays that are already in order by adding a test to skip the call to merge() if a[mid] is less than or equal to a[mid+1]. With this change, we still do all the recursive calls, but the running time for any sorted subarray is linear.

**Eliminate the copy to the auxiliary array**. It is possible to eliminate the time (but not the space) taken to copy to the auxiliary array used for merging. To do so, we use two invocations of the sort method: one takes its input from the given array and puts the sorted output in the auxiliary array; the other takes its input from the auxiliary array and puts the sorted output in the given array.

**Input Format:**

* The input contains the elements that are separated by spaces. Each line of the input is treated as a separate test case.

**Output Format:**

* Print the output as specified in the sample test case. Print a blank line for each test case.

**Note:**

* Print a line “**Insertion sort method invoked...”** when there is a cutoff to insertion sort.
* Print a line **“Array is already sorted. So, skipped the call to merge...”** when the array is already sorted, means when you skip the call to merge method.
* Check for the test cases given in the folder.

**Sample Input #1:**

**1,2,4,4,5,7,6,8,9**

**4,56,89,-1,5,4,3,66,44,23,56,78,12,45,67,89,12,34,56,0**

**4,5,6,1,4**

**Sample Output #1:**

**Insertion sort method invoked...**

**Insertion sort method invoked...**

**Array is already sorted. So, skipped the call to merge...**

**[1, 2, 4, 4, 5, 6, 7, 8, 9]**

**Insertion sort method invoked...**

**Insertion sort method invoked...**

**Insertion sort method invoked...**

**Insertion sort method invoked...**

**[-1, 0, 12, 12, 23, 3, 34, 4, 4, 44, 45, 5, 56, 56, 56, 66, 67, 78, 89, 89]**

**Insertion sort method invoked...**

**[1, 4, 4, 5, 6]**