Names:\_\_\_\_\_\_\_key\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Period:\_\_\_\_\_\_

**Computer Science 2 AP – Salute to Sorts**

Use the following sort algorithms to sort the original array in **ascending** order to determine total # of passes, swaps, comparisons, and Order of Magnitude (“Big O”). The contents of the array after the 2nd pass will be displayed in both ascending and descending order.

The original array will always be the same: **{75, -3, 99, 25, 52, 83, 12, 7, 38}**

**Monte Carlo sort**: Passes:\_varies\_ Swaps: varies Comparisons varies Efficiency: O(\_2N\_\_\_\_)

*NOTE: Contents of the array after the 2nd pass not applicable to this algorithm.*

**Linear sort**: Passes:\_8\_\_\_\_ Swaps: \_21\_\_\_\_ Comparisons:\_36\_\_\_\_ Efficiency: O(\_\_\_N2\_\_)

*What are the contents of the array after the 2nd pass?*

Ascending: \_\_\_\_[-3, 7, 99, 75, 52, 83, 25, 12, 38] \_

Descending: \_\_\_ [99, 83, -3, 25, 52, 75, 12, 7, 38]

**Bubble sort**: Passes:\_7\_\_\_\_ Swaps: \_\_21\_\_\_ Comparisons:\_\_35\_\_\_ Efficiency: O(\_\_N2\_)

*What are the contents of the array after the 2nd pass?*

Ascending: \_\_\_\_\_\_\_ [-3, 25, 52, 75, 12, 7, 38, 83, 99]\_\_\_\_\_\_\_\_\_\_

Descending: \_\_\_\_\_\_ [99, 75, 52, 83, 25, 12, 38, 7, -3]\_\_\_\_\_\_\_\_\_\_

**Selection sort**: Passes:\_\_8\_\_\_ Swaps: \_\_\_8\_\_ Comparisons:\_\_36\_\_\_ Efficiency: O(\_N2\_\_)

*What are the contents of the array after the 2nd pass?*

Ascending: \_\_\_ [-3, 7, 99, 25, 52, 83, 12, 75, 38]\_\_\_\_\_\_\_\_\_

Descending: \_\_ [99, 83, 75, 25, 52, -3, 12, 7, 38]\_\_\_\_\_

**Insertion sort**: Passes:\_\_8\_\_\_ Swaps: \_\_(varies) Comparisons:\_21\_\_\_ Efficiency: O(\_N2\_)

*What are the contents of the array after the 2nd pass?*

Ascending: \_\_\_\_ [-3, 75, 99, 25, 52, 83, 12, 7, 38]\_\_\_\_\_\_

Descending: \_\_\_ [99, 75, -3, 25, 52, 83, 12, 7, 38]\_\_\_\_\_\_\_\_\_\_\_

**Merge sort**: Passes:\_\_varies Swaps: \_varies Comparisons:varies Efficiency: O(\_N log2N\_\_\_\_)

*What are the contents of the array after the 2nd pass?*

Ascending: \_\_does not apply\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Descending: \_does not apply\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_