Name: _____ Period: _____ Date: _____

SORTING ALGORITHM WORKSHEET

ASSIGNMENT:

For each sorting algorithm, you will determine its efficiency by counting the number of loop iterations it takes to fully sort an integer array of various sizes when the data is organized in the worst possible order (in reverse), the best case (the data is already sorted), and the average case using a randomly generated list. Graphing the results will help reveal the "Big O" efficiency of each algorithm.

FOR DATA SORTED IN REVERSE ORDER: (USUALLY WORST CASE SCENARIO)

Sort Algorithm	Number of Elements							
	500	1000	1500	2000	2500	3000	3500	4000
Selection								
Insertion								
Merge								

FOR DATA SORTED IN OPTIMAL ORDER: (USUALLY BEST CASE SCENARIO)

Sort Algorithm	Number of Elements							
	500	1000	1500	2000	2500	3000	3500	4000
Selection								
Insertion								
Merge								

FOR DATA SORTED IN RANDOM ORDER: (USUALLY AVERAGE CASE SCENARIO)

Sort Algorithm	Number of Elements							
	500	1000	1500	2000	2500	3000	3500	4000
Selection								
Insertion								
Merge								

SORTING ALGORITHM SUMMARY:

Sorting Algorithm	General Efficiency (Big O)	Iterations Performed		
		Best Case	Worst Case	Average Case
Selection				
Notes:				
Insertion				
Notes:				
Merge				
Notes:				

Graphing the Worst Case