Problem 1.

[1, 16, 25, 31] + [-3, 0, 16, 27]
-3 < 1
[-3]
0 < 1
[-3, 0]
1 < 16
[-3, 0, 1]
16 <= 16
[-3, 0, 1, 16]
16 < 25
[-3, 0, 1, 16, 16]
25 < 27
[-3, 0, 1, 16, 16, 25]
27 < 31
[-3, 0, 1, 16, 16, 25, 27]
31 final
[-3, 0, 1, 16, 16, 25, 27, 31]

Problem 2.
[-1, -5, 67, -10, 21, 8, 4, 1]
[-5, -1, 67, -10, 21, 8, 4, 1]
[-5, -1, 67, -10, 21, 8, 4, 1]
[-10, -5, -1, 67, 21, 8, 4, 1]
[-10, -5, -1, 21, 67, 8, 4, 1]
[-10, -5, -1, 8, 21, 67, 4, 1]
[-10, -5, -1, 4, 8, 21, 67, 1]
[-10, -5, -1,1, 4, 8, 21, 67]

Problem 3
[-5, 42, 6, 19, 11, 25, 26, -3]
P = 25
S1 [-5, 6, 19, 11, -3] S2[42, 26]

P=6

[-5, -3] [19, 11]
[11, 19]
[-5, -3, 6, 11, 19]

S2[42, 26]

P = 26

S2[42]
[42]
[26, 42]
[-5, -3, 6, 11, 19, 25, 26, 42]

Problem 4 Shell Sort
[15, 14, -6, 10, 1, 15, -6, 0] h = 4
[15, 1]
[15, 14, -6, 10, 1, 15, -6, 0]
[14, 15]

[15, 14, -6, 10, 1, 15, -6, 0]
[-6, -6]
[15, 14, -6, 10, 1, 15, -6, 0]
[10, 0]

[15, 14, -6, 0, 1, 15, -6, 10]

h = 2
[15, 14, -6, 0, 1, 15, -6, 10]
[15, -6, 1, -6]
[15, 14, -6, 0, -6, 15, 1, 10]
[14, 0, 15, 10]
[0, -6, 10, -6, 14, 1, 15, 15]
h=1
[-6, -6, 0, 1, 10, 14, 15, 15]

Problem 5

Merge Sort and Quick Sorts have the same Omega time complexity, however I ranked Quicksort below Merge Sort because Quicksort Big O time complexity is worse than MergeSort. QuickSort having O(n^2) and Merge Sort having O(nlogn). Bubble Sort and Insertion Sort have the same Omega and Big O times complexities. The worst time ranking is for Selection Sort with its best time being O(n^2) and worst being O(n^2) as well.

1. Merge Sort
2. Quick Sort
3. Shell Sort
4. Insertion Sort
5. Bubble Sort
6. Selection Sort

Problem 10

As we increased the element size, the time increased as well. Not all the way through, because there were cases where some algorithms that had the same time complexity in best case scenario like Insertion Sort and BubbleSort, I saw that Insertion Sort performed significantly better than Bubble Sort.

Problem 12
Shell sort performed better