Sorting Algorithms Worksheet

1. Sort the sequence 3, 1, 4, 1, 5, 9, 6, 5 using insertion / bubble / selection sort.
2. What is the running time of insertion sort if all elements are equal?
3. Show the results of running Shellsort on the input 9, 8, 7, 6, 5, 4, 3, 2, 1 using the increments {1, 3, 7}.
4. What is the running time of Shellsort using the two-increment sequence {1, 2}?
5. Show how heapsort processes the input 142, 543, 123, 65, 453, 879, 572, 434, 111, 242, 811, 102.
6. What is the running time of heapsort for presorted input?
7. Rewrite heapsort so that it sorts only items that are in the range low – high, which are passed as additional parameters.
8. Sort 3, 1, 4, 1, 5, 9, 2, 6 using mergesort.
9. How would you implement mergesort without using recursion?
10. Determine the running time of mergesort for:
	1. Sorted input
	2. Reverse-ordered input
	3. Random input
11. Sort 3, 1, 4, 5, 9, 2, 6, 5, 3, 5 using quicksort.
12. Determine the running time of quicksort for:
	1. Sorted input
	2. Reverse-ordered input
	3. Random input

(from “Data Structures and Algorithm Analysis in Java”, 3rd edition, Mark Allen Weiss)