1 Sorting

1.1 Practice

Sort the following data using different sorting algorithm.
62 7 29 31 67 18 93

1. Insertion Sort
   62 7 29 31 67 18 93
   7 62 29 31 67 18 93
   7 29 62 31 67 18 93
   7 29 31 62 67 18 93
   7 18 29 31 62 67 93
   7 18 29 31 62 67 93

2. Selection Sort
   7 62 29 31 67 18 93
   7 18 29 31 67 62 93
   7 18 29 31 67 62 93
   7 18 29 31 67 62 93
   7 18 29 31 62 67 93
   7 18 29 31 62 67 93
   7 18 29 31 62 67 93

3. Heap Sort (show both heapify and sorting)
   Heapify:
   62 7 29 31 67 18 93
   62 7 29 31 67 18 93
   62 7 29 31 67 18 93
   62 7 29 31 67 18 93 (four leaves)
   62 7 93 31 67 18 29 (29 sink)
   62 67 93 31 7 18 29 (7 sink)
   93 67 62 31 7 18 29 (62 sink)
   Sort:
   67 31 62 29 7 18 93 (93 popped)
   62 31 18 29 7 67 93 (67 popped)
   31 29 18 7 62 67 93 (62 popped)
   29 7 18 31 62 67 93 (62 popped)
   18 7 29 31 62 67 93 (31 popped)
   7 18 29 31 62 67 93 (18 popped)
   7 18 29 31 62 67 93 (7 popped)

4. Merge Sort
   67 31 62 29 | 7 18 93
   67 31 | 62 29 | 7 18 | 93
   31 67 | 29 62 | 7 18 | 93
   29 31 62 67 | 7 18 93
   7 18 29 31 62 67 93

5. Quick Sort (always choose the first element as pivot)
   31 62 29 7 18 | —67— | 93
   29 7 18 | —31— | 62 | —67— | 93
   7 18 | —29— | —31— | 62 | —67— | 93
   —7— | 18 | —29— | —31— | 62 | —67— | 93

1.2 Scenarios

Sort the following data under different scenario using the **most appropriate sort**, and give your reasoning and runtime. There may be more than one correct answer.

1. (1, 2, 3, 4, 5, 6, 7)
   Insertion sort has O(N) runtime if the input is completely sorted.

2. A random generated integer data set with arbitrary size (could be really large).
   Ex. (65, 97, 85, 31, 72, 5, 98)
   If we know that the input is relatively large, (arbitrary size), the memory size should be under consideration. It is possible that the machine memory is not enough to make multiple copies of the data set, so a in-place sort will work better in this case.
   There might be another case that one machine can't take over even one full data set. What would happen? Think about only take part of the data and sort it first. Then, merge it with the rest of the data set. That's merge sort.

3. Kevin collects your QuickCheck worksheet and tries to sort it by names. What kind of sort would he use?
   Ex. (Erik, Louis, Velocity, Erika, Alieen, Brain)
   Since this is a "paper work" and the TAs might not want to print copies and waste papers, in-place sorting is preferable. An example is that Kevin find four TAs and split the papers in five groups. Each person sorts their stack and merge them back together.