CS 61B Discussion 10: Sorting Fall 2015

1 Sorting I

Show the steps taken by each sort on the following unordered list:

```
106, 351, 214, 873, 615, 172, 333, 564
```

(a) Quicksort (assume the pivot is always the first item in the sublist being sorted and the array is sorted in place). Circle the pivots on every step. 4

(b) Merge sort. Show the intermediate merging steps.

(c) LSD radix sort.

2 Sorting II

Match the sorting algorithms to the sequences, each of which represents several intermediate steps in the sorting of an array of integers.
Algorithms: Quicksort, merge sort, heapsort, MSD radix sort, insertion sort.

(a)
```
12, 7, 8, 4, 10, 2, 5, 34, 14
7, 8, 4, 10, 2, 5, 12, 34, 14
4, 2, 5, 7, 8, 10, 12, 14, 34
```

(b) `23, 45, 12, 4, 65, 34, 20, 43`
`4, 12, 23, 45, 65, 34, 20, 43`

(c) `12, 32, 14, 11, 17, 38, 23, 34`
`12, 14, 11, 17, 23, 32, 38, 34`

(d) `45, 23, 1, 65, 34, 3, 76, 25`
`23, 45, 1, 65, 3, 34, 25, 76`
`1, 23, 45, 65, 3, 25, 34, 76`

(e) `23, 44, 12, 11, 54, 33, 1, 41`
`54, 44, 33, 41, 23, 12, 1, 11`
`44, 41, 33, 11, 23, 12, 1, 54`

3 Runtimes

Fill in the best and worst case runtimes of the following sorting algorithms with respect to n, the length of the list being sorted. For radix sort, assume we are sorting integers and k is the average number of digits in the strings being sorted.

	Insertion sort	Selection sort	Merge sort	Heapsort	Radix sort
Worst case					
Best case					

4 Comparing Algorithms

(a) Sometimes insertion sort can be more efficient than merge sort. Give an example of an input array that demonstrates this.

(b) When might you decide to use radix sort over a comparison sort, and vice versa?