CS 014 Assignment #4

Due on Tuesday 11/28/2017 at the *beginning* of the class

Instructions: Please include the following on the cover page of your assignment:

- Full name
- Student ID
- Your lab section number
- The name of the TA of your lab

Answer the following questions.

1. (15 points) For the array given below, show the state of the array after running each of the given sorting algorithms as described below. Assume that the sorting algorithms are designed to sort the array in ascending order.

   $$[8, 7, 9, 10, 4, 3, 1, 5, 6, 2]$$

   (a) (2 points) Bubble sort. Show the state of the array after running one iteration of the outer loop.

   (b) (2 points) Selection sort. Show the state of the array after running one iteration of the outer loop.

   (c) (2 points) Bottom-up merge sort algorithm. Show the state of the array after running the first iteration that merges elements into lists of size 2.

   (d) (4 points) Quick-sort algorithm. Assume that we use the median-of-three pivot selection technique. Show the state of the array after the first partition step is finisehd.

   (e) (5 points) In-place Heap-sort. Show the state of the array after the heap has been constructed and before any items have been removed from the heap.

2. (10 points) In some applications, we would like to keep the array sorted using different comparison functions. For example, we can keep an array of strings sorted using a case-sensitive and case-insensitive comparison functions. Instead of wasting the memory and duplicating the entire array, we can use an *index-based sorting algorithm.* In this algorithm, we keep the original array intact and create a separate *index array* of integers where each entry in the array points to an index in the array. Then, we sort the index array such that the indexes point to the array in the desired sort order. For example, for an input array $A = \{MN, NY, CA, OK\}$, the index array will be $I = \{2, 0, 1, 3\}$. $I[0] = 2$ points to $A[2] = CA$ which is the smallest element in A. Similarly, $I[1] = 0$ points to $A[0] = MN$ which is

the second smallest element in the array A. Write-down an insertion-sort-based sorting algorithm (pseudo code is fine) that takes as input an array A and a comparison function c, and returns an index array that sorts A based on the given comparison function. Your algorithm is *not* allowed to modify the values of A.

```
Index-Sort(A, n, c) {
...
}
```

Note: This assignment should be done individually. You can either deliver it on iLearn or hand it out at the *beginning* of the class. You can either handwrite it or type it on your favorite word processor and submit it as a PDF file. As an acknowledgment for your typing effort, you will get an extra 10% for typing it *neatly* without exceeding 100% of the final grade.