Due date: Friday, December 2 at 7:30 AM. You will need to submit this via GradeScope.

For this assignment, each answer must be contained within a single piece of paper. When you submit to GradeScope, you will need to inform the system which page of your scanned PDF contains the answer. *Do this even if your submission is a single page.*

Please review the syllabus and course reference for the expectations of assignments in this class. Remember that problem sets are not online treasure hunts. You are welcome to discuss matters with classmates, but remember the Kenny Loggins rule. Remember that you may not seek help from any source where not all respondents are subject to UC Irvine's academic honesty policy.

The goal of these problem sets is to get you to explore concepts from class. This should help prepare you for the exams and, more importantly, for understanding the big ideas from ICS 46.

1. Among Selection Sort, InsertionSort, MergeSort, QuickSort, and HeapSort, which algorithm would you use for each list-sorting situation below? Justify your answers.

   A question might have multiple right answers, but each definitely has at least one wrong answer.

   (a) The list has several hundred records. The records are quite long, but the keys are very short.

   (b) The list has about 45,000 records. It is necessary that the sort be completed reasonably quickly in all cases. There is barely enough memory to hold the 45,000 records.

   (c) The list has about 45,000 records, but it starts off only slightly out of order.

   (d) The list has about 25,000 records. It is desirable to complete the sort as quickly as possible on the average, but it is not critical that the sort be completed quickly in every single case.

2. (a) Describe a vector of distinct elements with size $n = 32$ such that if it is the initial input to HeapSort in the order presented, no other permutation of the input will cause the algorithm to use more comparisons to heapify the array. Explain why it does this. You do not (and should not) write out the entire array, but rather describe the order.

   (b) Describe a vector of distinct elements with size $n = 32$ such that it is a max-heap and, if it is the initial result of running heapify, no other heap created from the input will cause the algorithm to use more comparisons to complete the sorting stage of HeapSort. Explain why it does this. You do not (and should not) write out the entire array, but rather describe the order.

3. Please fill out course evaluations; it is important to me that you complete these. You do not need to submit anything to GradeScope for this part.