1. Go to csilm.usu.edu. Look at CS3/Priority Queue, activity 3 (Heap as a Priority Queue). Using the slider bar, increase the speed. As you insert nodes into the heap, notice the two forms of the heap. The physical form is an array that is shown at the bottom of the screen. The logical form is the tree itself.

Now, suppose you need a new operation for the priority queue. We will call it changeKey. The idea is this. You have a priority queue to give out scholarships based on ACT. Someone retakes the ACT test and their score improves or gets worse. You want to adjust the tree to accommodate this change. Since we can't search a priority queue easily, we will suppose that finding the subscript location of the value that changed is handled by someone else. Write changeKey(int newValueSubscript, int newValue).

2. For a heap as a max priority queue, which of the following are efficient to do? Explain your answer:
   a. find the largest
   b. find the smallest
   c. find a node with a particular value
   d. Sort the array

3. For each of the nodes in the heaps below, indicate the null pointer length. Merge the two leftist heaps below.

4. Merge the two skew heaps below.
Submit a single document on eagle or give it to me in class.

Notes

Turn in your written homework through Eagle in a .doc, .odt, .pdf format or you can bring a paper copy of the homework with you to class. It will be graded by randomly selecting a subset of problems to evaluate. Not every problem will be graded. Bring a copy of the answers to class so that we can discuss them.

Written homework provides an excellent framework for achieving the goals of obtaining a working knowledge of data structures, perfecting programming skills, and developing critical thinking strategies to aid the design and evaluation of algorithms. Since programming has a high overhead in terms of program entry and debugging, all important topics in this course cannot be covered via programming projects. Written homework exercises allow students to learn important material without a high time investment. Although the point value is low, the benefits are great. You can perfect your programming skills without spending hours at the computer and can get feedback on your thinking skills from your study partners. Students that consistently do quality homework, have far superior test scores. Because assignments are done as a group and any questions are discussed in class or during office hours, written solutions to the homework will not be provided.

*Note, these exercises may be done in groups of one, two, or three. If more than one person is involved, list all the names on one set of answers. Groups may change throughout the semester. Answers should not be compared with others not in your group. You will learn much more by working in a group than you will learn working by yourself.*