EECS 560: Lab 8 Report

Comparing Leftist Heap and Skew Heap

For this experiment, a leftist heap and skew heap were timed for how long it took to insert different amounts of values into them, and for how long to perform a mix of insert and delete min operations once they were full. They were timed for the amounts n = 50000, 100000, 200000, and 400000. Each value that was inserted into the heaps itself had a value between 1 and 4 \* n (the total number inserted in each trial). A for loop was created that would iterate over 5 different seeds for srand(). In each iteration of that loop, another for loop would time how long it took to insert the n values for each value of n, as well as time how long it took to perform 1/10th of n insert/delete min operations.

The results of the experiment are interesting. On average, it took a shorter amount of time to insert the n values into a leftist heap, while it took a shorter amount of time to perform the operations on a skew heap. This makes sense for the skew heap performing better in operations, because it has less steps in its merge function. The leftist heap has to compare the rank every time the merge function is called, where as the skew heap just swaps the left and right child every time. I suspect all of those extra rank calculations and comparisons added to the time of leftist heap operations. However both heaps do not seem to be following a logarithmic curve, which is what should be expected. Both perform very similarly, they are definitely comparable in performance as was stated in class. The timing results for each of the seeds as well as graphs of the average times are on the following pages.





