Connor Welch

11/1/16

EECS 560

Lab 7 Report

For lab 7, performance times were recorded for our min max and min 5 heap. This was accomplished by using a seed to generate a consistent list of random numbers. A timing was recorded from each data structure for its build method and multiple delete and insert operations. 5 seeds were used for number generation as well as 4 sizes of n coming to a total of 20 time recordings for both the build operations and the delete and insert operations. For both the min max heap and min 5 heap, a bottom up approach was used for building the heaps. Both of the build heap methods for these structures utilized recursive methods to accomplish the task. When looking at the recorded times for these data structures we can analyze the performance between the two and compare them.

For the min max heap, the performance of the build and the operations is very similar. As n became larger so did the time values. However, the difference in the times are very small. With an n the size of 400,000 was only five times as slow as with an n of 50,000. When comparing sizes of n at 50,000 and 100,000 we can see that the amount of time it took didn't even increase by a factor of 2. It is also worth noting that the build operation performance was consistently ten times slower than the delete and insert operations

For the min 5 heap, there is a much more drastic difference between the build and insert and delete operations. The build times do not increase at a drastic rate, but the insert and delete operations do. When comparing n sizes of 400,000 and 50,000 there is nearly a 10

second difference between the insert and delete operations. When comparing sizes of n of 50,000 and 100,000, there was still a relatively drastic increase among the operations.

Analyzing both structures together we can determine which of these heaps perform more efficiently. When comparing the build operations of the structures, we can see that starting off the Min 5 heap has almost a twice as fast build time but as n increase it becomes faster than a factor of 2. However, when comparing the insert and delete operations we can see that the min max heap has a significantly faster time. When n is at its largest, the min max heap can still perform the operations in a fraction of a second, while the min 5 heap takes over 11 seconds to complete its task.

Based on the comparisons from the two data structures we can come to a conclusion on their performance. The min max heap data structure is much better at performing delete and insertion operations. The min 5 heap is better for quickly building a list of data. However, with the difference in the build times still relatively small, we can conclude that a min max heap makes a better data structure in a general setting.