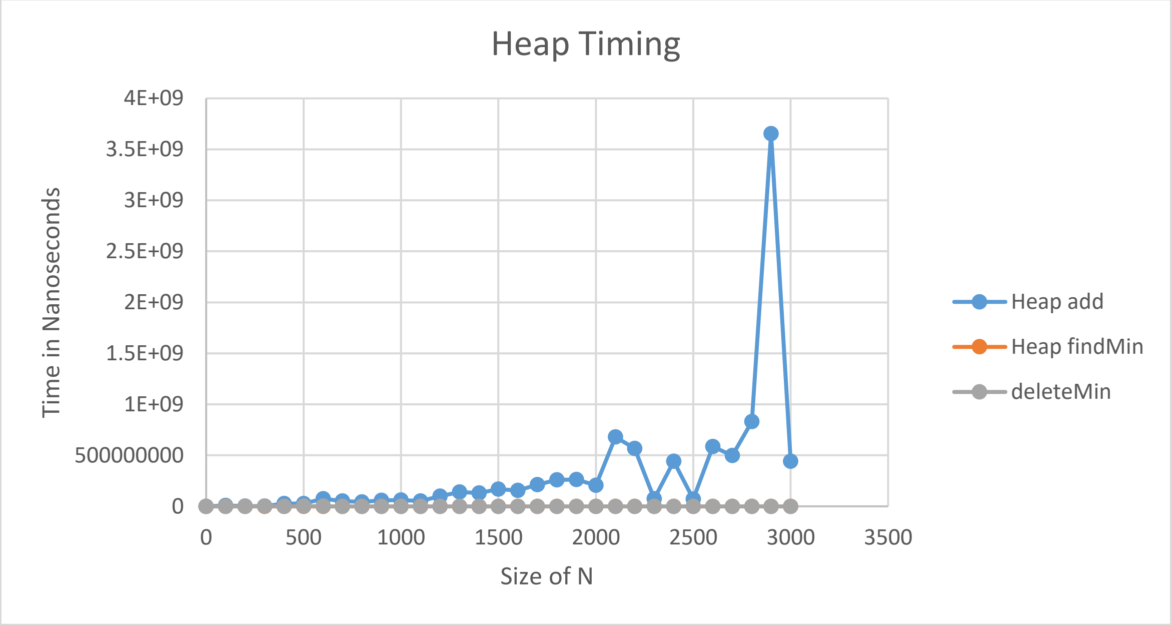
Chenxi Sun

Analysis Document for Assignment 11



1. I did the timing for the priority queue heap by generating a random arraylist of integers. This is done by shifting the index after creating an arraylist from 0 to size-1. A for loop is created and the heap is used to add the get method of the arraylist inside the for loop. I did the timing from size 0 to 3000 items added into the heap. In the chart above the Timing for add exhibit log N behavior so the big o for add is O(log N).
2. The cost for add is already determined from the plot above to be O(log N). The cost for deleteMin will also be O(log N). The reason for this is that although the deletion is constant, the cost of percolating the last element down through the heap will make it log N, This is also shown in the plot above. For findMin() the big O is constant as it just returns the first element in the array since the min value is the heap or the element in index 0.
3. The most important use of a priority queue might be a printer. So for printer if everything is put in a priority queue, those that are printing the minimum amount of pages should be processed first. The one that is printing the most amount should be printed last.
4. I spent about 5 hours on this assignment most of it in programming deleteMin().