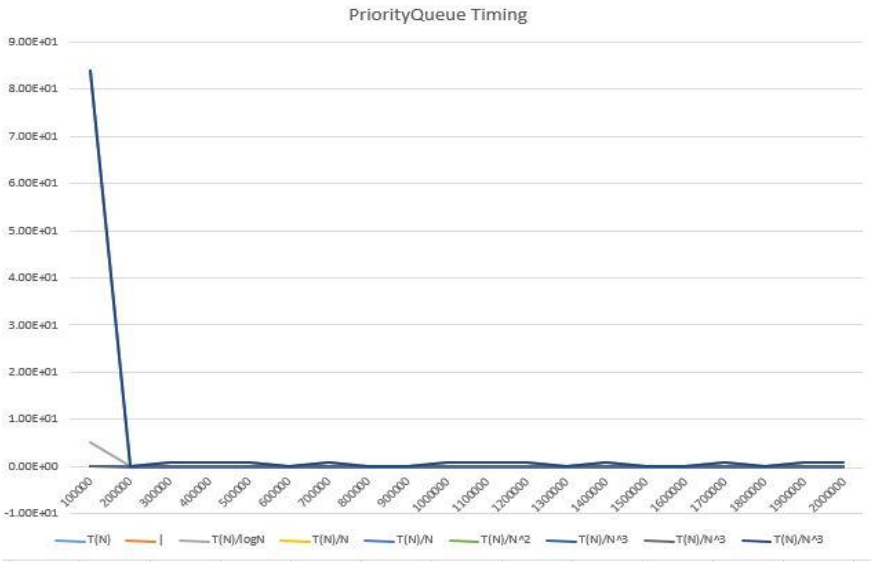
When you are satisfied that your program is correct, write a brief analysis document. The analysis document is 20% of your Assignment 11 grade. Ensure that your analysis document addresses the following.

1. Design and conduct an experiment to assess the running-time efficiency of your priority queue. Carefully describe your experiment, so that anyone reading this document could replicate your results. Plot the results of your experiment. Since the organization of your plot(s) is not specified here, the labels and titles of your plots(s), as well as, your interpretation of the plots is critical

For my experiment I looped through each of the functions adding values to lists of the priority queue. Using timing tests, I then printed out the results of each function: add, findMin, and deleteMin.



2. What is the cost of each priority queue operation (in Big-O notation)? Does your implementation perform as you expected? (Be sure to explain how you made these determinations.)

The cost of add os O(c): percolate up (average of 2.6 compares)

The cost of findMin is O(c): just return the root

The cost of deleteMin is (logN): percolate down (rarely terminates before near the bottom of the tree)

My implementation did perform in conjunction with these expected results as the complexity was similar to the definitions for these functions.

3. Briefly describe at least one important application for a priority queue. (You may consider a priority queue implemented using any of the three versions of a binary heap that we have studied: min, max, and min-max)

One important application of my priority queue would be add. The add method percolates up than all the nodes below it. If the new item is the smallest in the set, the cost is O(log N).

4. How many hours did you spend on this assignment?

I spent around 15 hours on this assignment

Programming partners are encouraged to collaborate on the answers to these questions. However, each partner must write and submit his/her own solutions.

Upload your solution (.pdf, only) here by 11:59pm on Wednesday, November 16.