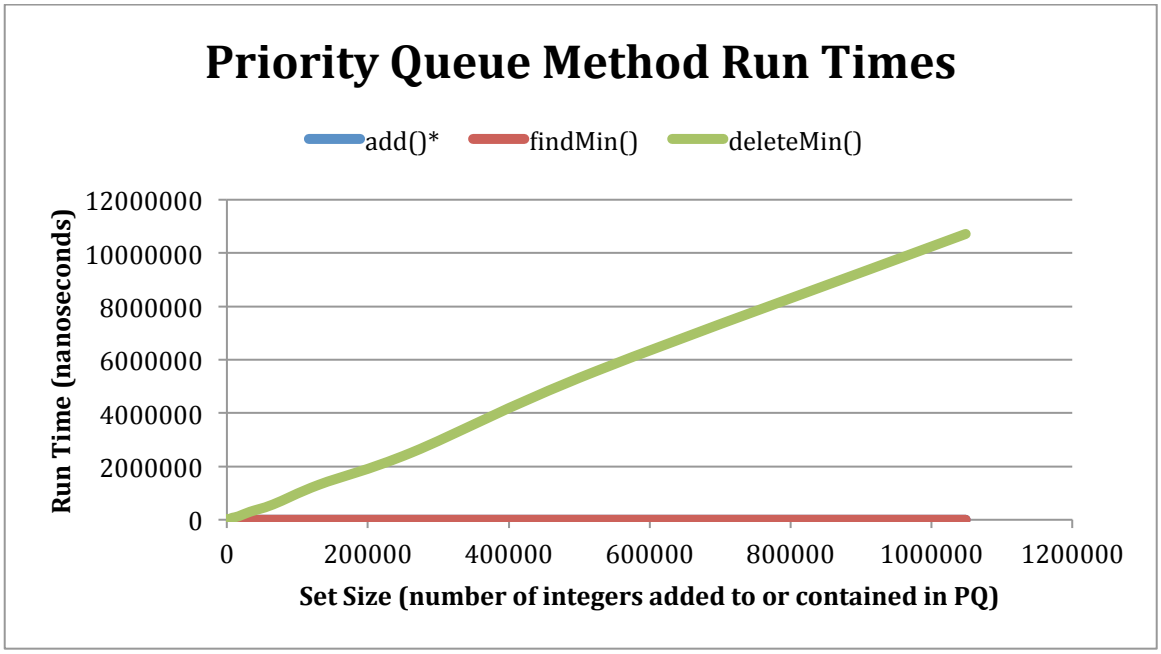
CS2420 Assignment11 Analysis
Daxton Wilson
U0264580

1. Experiment Steps:
   1) Initialize a priority queue with size N.
   2) Run and time methods, add(), deleteMin() and findMin() for each of the different set sizes. Add will be done in random order for each set size N, each time adding one element to the PQ and timing the single element addition.
   3) Plot each of the different methods' runtimes on the same plot to compare and contrast run-time performance and evaluate complexity.
   4) Compare and contrast the different methods with your expected results.



*Add() not shown in plot because it is overlapped by findMin(), both of which have a complexity of O(c).
(For analysis and interpretation see question 2)

2. The run times are as expected for each of the different methods timed. As expected, add() has a complexity of O(C). Remove() was expected to have the greatest complexity with a complexity of O(logN) for both average and worst-case because of the percolate down method to ensure that we have a min-heap. Percolate down on average requires traversal down to the bottom two levels in the heap because they contain (2^n and 2^n-1 / sum(2^n)) of the elements, roughly 65-75% of the elements in the heap. FindMin() has a complexity of O(c) because it only ever looks up the first element in our heap regardless of set size, just as we expected.

3. An example of a min-max heap can be found in sports statistics. We can keep track of where all the players or teams in a given sport rank in each category and easily find both the best and worst. It will also allow us to access this data quickly. If we want to find the top 5 scoring teams in the NBA for example, we simply deleteMax() 5 times because it returns the team at the max to give us a top 5 in-order ranking (same situation applies for deleteMin() for the worst teams).

4. I spent 5 hours on this assignment.