

Analysis of Priority Queue Implementations

Comparison of Default Java Priority Queue implementation against Custom Sorted Array Implementation

Time Complexity of 'insert()' operation: $O(N)$

Time Complexity of 'deleteMin()' operation: $O(1)$

Each implementation was run 10 times per value of N, and the results averaged

N Value	Java Implementation	Custom Implementation
5000	2 time units	46.4 time units
10000	4.8 time units	144.3 time units
20000	8.9 time units	952.2 time units
100000	31.2 time units	N/A
1000000	398.4 time units	N/A

N = 5000

Default Java Average: 2 time units

Custom Average: 46.4 time units

N = 10000

Default Java Average: 4.8 time units

Custom Average: 144.3

N = 20000

Default Java Average: 8.9 time units

Custom Average: 952.3 time units

The Java implementation appears to have a time complexity in this situation of $O(\log(N))$. This is shown in the decreasing time overhead as N increases.

The custom implementation appears to have a time complexity in this situation of $O(N\log(N))$.

The differences between the times and complexity of the two implementations is likely due to Java implementation using a binary heap which would greatly reduce the time complexity of the 'insert()' or 'offer()' operations.