

Excerpt from "Data Structures in Java" by Standish

how to sort using
priority queues

Non-decreasing
functions

```
void priorityQueueSort(ComparisonKey[ ] A) {  
    int i; // let i be an integer array index variable  
    int n = A.length; // let n be the length of the array A to be sorted  
    PriorityQueue PQ = new PriorityQueue( ); // let PQ be initially empty  
    for (i = 0; i < n; i++) PQ.insert(A[i]); // put A's items into PQ  
    for (i = n-1; i >= 0; i--) A[i] = PQ.remove( ); // remove PQ's items  
                                                    // and put them in A  
}
```

Program 5.2 A Priority Queue Sorting Method

worst case

$$\sum_{i=0}^{n-1} f(i) \leq \sum_{i=0}^{n-1} f(n-1) = n \cdot f(n-1) \leq \boxed{n f(n)}$$
$$\sum_{i=0}^{n-1} g(i+1) \leq \sum_{i=0}^{n-1} g(n) = \boxed{n g(n)}$$

$$\begin{aligned} \text{Total} &\leq O(1 + n \cdot f(n) + n \cdot g(n)) = O(1 + n(f(n) + g(n))) \\ &= O(n(f(n) + g(n))) = \\ &= \boxed{O(n \cdot \max(f(n), g(n)))} \end{aligned}$$