1. Sorting with a Priority Queue (PQ-sort) consists of performing \( n \) insert(e) (number and type of PQ operations) PQ operations followed by \( n \) removeMin() (number and type of PQ operations) PQ operations.

2. When a PQ is implemented with an unsorted sequence (list), the PQ-sort algorithm is referred to as selection-sort (a type of sorting algorithm) and runs in time \( O(n^2) \). In this case, the most expensive PQ operations are the removeMin() operations.

3. When a PQ is implemented with a sorted sequence (list), the PQ-sort algorithm is referred to as insertion-sort (a type of sorting algorithm) and runs in time \( O(n^2) \). In this case, the most expensive PQ operations are the insert(e) operations.

4. When a PQ is implemented with a heap, which is in turn realized by an array-based structure of binary trees, the insert(e) operation takes time \( O(\log n) \) and the removeMin() operation takes time \( O(\log n) \). In this case, the PQ-sort algorithm is referred to as heap-sort (a type of sorting algorithm) and runs in time \( O(n \log n) \).

5. **Bonus.** A(n) **Adaptable** priority queue allows updating the key of any entry and arbitrarily removing any entry. **Location-aware** entries are used to facilitate this functionality.