Time complexity for OrderManager::Add
In the first approach, insert function takes O(n) time, because when an element is added all the other elements need to be shifted to make a room for it.
std::sort takes O(n log n) time because
In the second approach, std::lower_bound is actually a Binary Search method that repeatedly divides the search space in half to find a solution. Thus it takes O(log n) time. But it still takes O(n) to insert element into the position we found.

Time complexity for OrderManager::Update
It seems that there is no difference between two approaches in this part. A linear search is performed to find order_id. The time complexity is O(n) for both approaches. Updating order quantity is a constant-time operation.

The rationale of maintaining orders:
For approach 1, std::sort is called to sort all orders before printing out.
For approach 2, std::lower_bound is called when a new order is added and it makes sure that new order is placed at the right sorted position.