**Chapter 17 Concept Quiz**

Determine whether each of the following statements is true or false. If a statement is false, please explain why.

1. The implementation of bubble sort relies on finding smallest or largest item in a linear data structure.
2. When the majority of items in a linear data structure are already sorted, bubble sort requires fewer comparisons than selection sort.
3. The inner loop of the selection sort is to find the smallest or largest item in the unsorted portion of the list.
4. In selection sort, as in bubble sort, the sorted portion of the list can sometimes grow by more than one item in a single iteration.
5. Like bubble sort and selection sort, the inner loop of insertion sort iterates through the unsorted portion of the list.
6. When the majority of items in a linear data structure are already sorted, insertion sort requires fewer outer loop iterations than selection sort.
7. When the majority of items in a linear data structure are already sorted, shell sort requires fewer outer loop iterations than insertion sort.
8. Quick sort is an iterative sorting algorithm.
9. For a list in descending order that needs to be sorted in ascending order, quicksort does not necessarily have an advantage compared to insertion sort, particularly when using a simple pivot selection strategy.
10. Sorting is done during division process for merge sort.

**Answers to the Quiz**

1. **False**. The implementation of bubble sort does not explicitly rely on finding the smallest or largest item in a linear data structure. Instead, it repeatedly compares adjacent elements and swaps them if they are in the wrong order.
2. **True**.
3. **True**.
4. **False**. In selection sort, the sorted portion of the list always grows by exactly one element per iteration
5. **False**. The inner loop of insertion sort is used to traverse the sorted portion of the list, not the unsorted portion, in order to find the correct position for the current element being inserted.
6. **False**. The outer loop of both insertion sort and selection sort always iterates through all elements of the list, regardless of how sorted the data is. The difference lies in the work done inside the loops; insertion sort performs fewer shifts when the list is mostly sorted, but the number of outer loop iterations remains the same.
7. **True**.
8. **False**. Quick sort is primarily a recursive sorting algorithm.
9. **True**.
10. **False**. The sorting happens during the merge step.