Workbook T2A1-B RfQ Response

Q1. Identify and explain the workings of TWO sorting algorithms and discuss and compare their performance/efficiency (i.e. Big O)

Bubble sort works by looking at the elements a pair at a time, swapping them round if the second one of the pair is larger than the first or if the element is in the wrong order. This continues until the list is sorted in smallest to largest or vice versa (Reyna A 2020). The worst case complexity of the bubble sort is O(n^2) as it involves nested loops. Bubble sort firstly interates through the data set in the outer while loop until the there is no more items to sort in the list or array. During this, the inner nested loop compares the paired values and works out the order by swapping them in order (Brandon J, 2020).

Merge sort breaks a list of items into many tiny lists. The first elements of both list is compared and forms a new list. It then merges all these lists into newly sorted ones, finally producing a single sorted list. Merge sort method is a O(log n) algorithm as each time the list of items is past through the loop, the set of items is sorted in half (Brandon J, 2020).

Though Merge sort uses more code than Bubble sort, I would recommend Merge sort for ACME Coporation marketplace web application as it is is more efficient. Bubble sort may not suit larger data sets as as it's often necessary to go through the list several times until the list is sorted in the required order. This can take up more time for the algorithmic logic to run the code and be less effecient especially if every item in the arrays need to be swapped.

Q2. Identify and explain the workings of TWO search algorithms and discuss and compare their performance/efficiency (i.e. Big O)

Linear search is used if the aim is to find one item in a million. It starts at the first left most item and compare it with each element from the list of arrays. If it's the right item it will stop searching otherwise, look at the next item. The time complexity of linear is O(n) as the time it would take for the search to complete can depend on how many elements are in the list of arrays.

Binary search through data that is already sorted. It starts by comparing the data set with the element to be searched in the middle of the list/array and follows the steps below:

1. If it matches, the index of the element is recorded and returned.

2. If it doesn't match, it will work out if the element to be searched is greater or less than the middle value.

3. When the element to be searched is greater than the middle value, then elements on the right side from the middle value will be picked. Values from the left side will be picked instead if the element to be searched requires to less than the middle value.

4. The half you don't need is disregarded. This is repeated until the item is found.

The worst case complexity of the Binary search is O(log n). Like Merge sort, Binary search algorithm halves the set of items each time the list of items is passed through the loop so it can be useful (Brandon J, 2020).

Binary searches allow faster searching and handy when dealing with large, sorted data sets. If the set of data is not sorted Binary search will not work. Though Linear search code is basic it isn't efficient as it might involve looking at every item on the list (Casey K, 2020).

Reference Source

Brandon J, 2020, CS Fundamentals - Big O (Part 1), Academy of Information Technology, https://ait.instructure.com/courses/3971/pages/cs-fundamentals-big-o?module_item_id=336423

Bredon J, 2020 CD Fundamentials - Big O (Part 2), Academy of Information Technology, https://ait.instructure.com/courses/3971/pages/cs-fundamentals-big-o?module_item_id=336423

Casey K, 5 Aug 2020, Let Us Understand Searching Algorithms, Codersera, https://codersera.com/blog/let-us-understand-searching-algorithms/

Reyna A, 25 Oct 2020, 5 Sorting algorithms every programmer should know, https://betterprogramming.pub/5-basic-sorting-algorithms-you-must-know-9ef5b1f3949c