Hyoin Lee (20102711)

**Step 1** – Knowledge Step 1 – Knowledge Question (40-70 words) In your own words, describe what sorting is in general.

Sorting is kind of a process of data rearrangement into a specific order or based on specific criteria mainly to make the data more organized, making it easier to search, retrieve, and analyse. For example, binary search is only available on a sorted database.

**Step 2** – Knowledge Question (60-100 words) Research sorting algorithms. Describe advantages and disadvantages for at least three different sorting algorithms. Please provide references for external resources.

**Bubble Sort**

The bubble sort algorithm works by repeatedly swapping adjacent elements that are not in order until the whole list of items is in sequence. In this way, items can be seen as bubbling up the list according to their key values.

* Advantage: Easy to implement and requires minimal space.
* Disadvantage: Inefficient for large lists, suitable for academic purposes.

**Selecting Sort**

The selection sort works by repeatedly going through the list of items, each time selecting an item according to its ordering and placing it in the correct position in the sequence.

* Advantage: Performs well on small lists, no extra storage needed.
* Disadvantage: Inefficient for large lists and sensitive to initial order.

**Insertion Sort**

The insertion sort repeatedly scans the list of items, each time inserting the item in the unordered sequence into its correct position.

* Advantage: Simple and space-efficient, good for small lists.
* Disadvantage: Inefficient for large lists, best for few items.

**Quick Sort**  
The quick sort works on the divide-and-conquer principle. First, it partitions the list of items into two sublists based on a pivot element. All elements in the first sublist are arranged to be smaller than the pivot, while all elements in the second sublist are arranged to be larger than the pivot. The same partitioning and arranging process is performed repeatedly on the resulting sublists until the whole list of items are sorted.

* Advantage: Best for large lists, sorts in place with no extra storage.
* Disadvantage: Worst-case performance similar to other inefficient sorts, but generally most effective and widely used.

**Merge Sort**Merge sort is one of the most efficient sorting algorithms. It is based on the divide-and-conquer strategy. Merge sort continuously cuts down a list into multiple sublists until each has only one item, then merges those sublists into a sorted list.

* Advantage: Can be used with linked lists without taking up any more space
* Disadvantages: For small datasets, merge sort is slower than other sorting algorithms

References

<https://sciencing.com/the-advantages-of-heap-sort-12749895.html>

[https://www.simplilearn.com/tutorials/data-structure-tutorial/merge-sort-algorithm#what\_are\_the\_advantages\_of\_the\_merge\_sort](https://www.simplilearn.com/tutorials/data-structure-tutorial/merge-sort-algorithm%23what_are_the_advantages_of_the_merge_sort)

**Step 3** – Knowledge Question (20-50 words) In your own words, describe why you generally need comparison operators to successfully sort a list of objects. In addition, describe how you could sort a list of objects without adding comparison operators.

Comparison operators allow us to determine the relative order of those objects based on certain criteria or attributes.   
To sort a list of objects without adding comparison operators, we could define custom comparison functions or methods for the objects which would evaluate specific attributes or properties of the objects and determine their relative order based on those criteria.

**Step 4** – Reason why I chose ‘Quick sort’.

I find ‘Quick Sort’ can sort a large list of players based on their scores quickly, which is important in games where I might have many players to deal with. Also, It doesn’t need to use a lot of extra memory to do its job which means it’s more memory efficient. Last but not least, it is easy to understand and implement.