Q3-1. Say in words how you will approach the problem.

We can first group together data one way (by book), and then sort within the groupings another way (by word). So we can use the "value-to-key conversion" design pattern. The basic idea is to move part of the value into the intermediate key to form a composite key, and let the MapReduce execution framework handle the sorting.

Instead of emitting the word as the key, we would emit the word and the book as a composite key:

(book, word) → (count)

The sensor reading itself now occupies the value. We must define the intermediate key sort order to first sort by the book (the left element in the pair) and then by the word(the right element in the pair). We must also implement a custom partitioner so that all pairs associated with the same sensor are shuffled to the same reducer.

Properly orchestrated, the key-value pairs will be presented to the reducer in the correct sorted order:

(book1, word1) → [(count)] (book1, word2) → [(count)] (book1, word3) → [(count)] ...