This week’s data structure was stacks and queues and its algorithm was sorting methods. These two as a pair are strong when dealing with large data sets. Stacks and queues give a user methods in which to append, prepend, or pop items depending on how the set needs to be accessed, and the sorting methods can, of course, organize all of your data. This week’s text mention that stacks are useful for last-in first-out application and queues are useful in first-in first-out applications.

There are weaknesses as well, however. With sorting specifically, there are many ways to sort data. Some of the algorithms used are very efficient for small data sets, but when used on larger ones they take a very long time to complete. This can be a problem if, let’s say, you were developing for Amazon and needed to sort all the items in their catalog.

Both stacks and queues, as well as this week’s sorting algorithms are useful with linked lists and would probably be mostly used in real-world applications as such. Another application of the sorting methods learned could be in combination with the search methods learned last week. A sorted list could potentially make a search quicker!