James Vigilante

IT2660

Assignment 8 – MS and QS

12/9/2021

**Write a very short essay regarding the following topics:**

* Merge Sort and QuickSort are the fastest general sorting algorithms known for unconstrained data sets.
* Analyze both Merge Sort and QuickSort.
* What are the best case and worst case for each?
* When do each of them do well, and when do each of them do poorly?
* Is it ever useful to use a simpler but less efficient sorting algorithm like Bubble Sort? *(O(N^2))*
  + When and why?

The best case scenarios for both quick sort and merge sort are very different. Merge sort has a consistent speed on any size of data, while a quick sort works really well only on a smaller array. I view the merge sort as your daily driver: reliable, easy, just not terribly fast. On the other hand, quick sort is a drag racer. Incredibly speedy given a small track, but not practical for longer distances.

The worse case scenario for merge sort is one where every element has to be compared at least once. The best example of this that I could find online while doing research for this paper was this quote from stackOverflow: (paraphrasing) “The worst case scenario for a merge sort is an array that looks like this: {0,2,4,6,1,3,5,7}. When it splits into right/left, every element needs to be compared.”. On the other hand, the worse case scenario for quick sort is just having your program choose bad pivots, like the highest or lowest number in the array.

Bubble sort is a weirdly simple algorithm. It can be useful though, say when you’re trying to teach the concept of sorting algorithms or with extremely small data sets.