Colleen Lemak

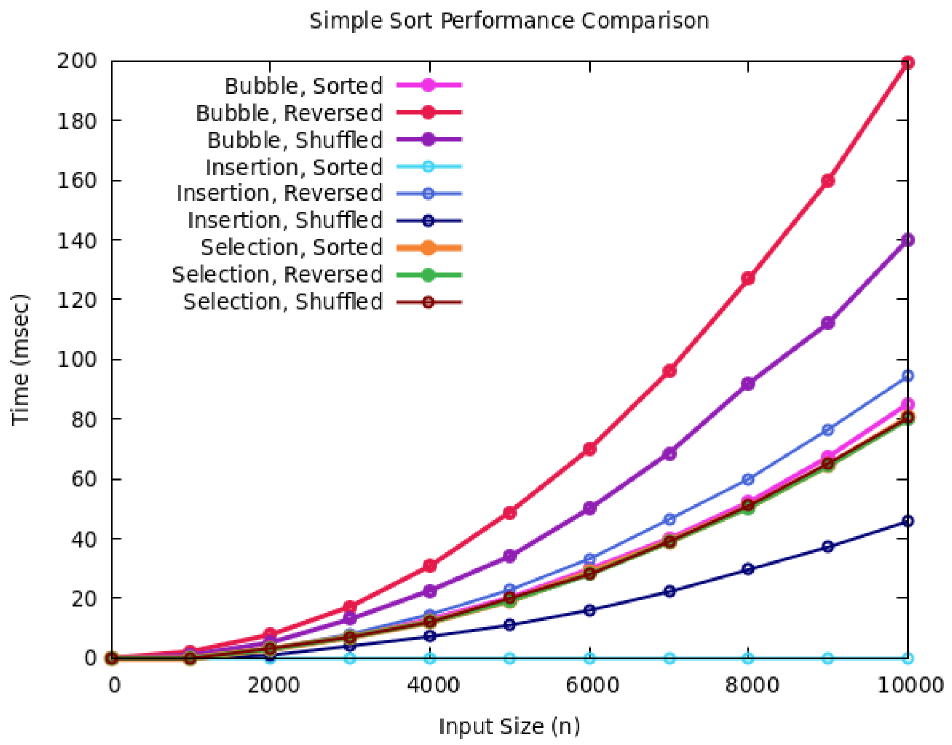
Professor Bowers

CPSC 223 Section 2

16 September 2021

HW 1 Write Up

For bubble, sorted (in pink), the time it took to completely "sort" the array was about the same time as selection, sorted (in orange). This makes sense as both algorithms have O(N) swaps to make, but given the array is already sorted and it needs a pass, it takes about 80 msec; this amount of time was also taken for selection, reversed (in green), and selection shuffled (in maroon). The selection sort algorithm is very consistent with its times, no matter how the array is ordered, probably in part to how it checks each position for a swap, making very little time difference in terms of performance time, if any at all. In insertion, sorted (in teal), the test recorded 0 msec; I think this value is different from other algorithms using a sorted array because insertion does not detect any need to slide over any values, thus it does not need much time at all to complete the test. Insertion, reversed (in light blue), is about 50 msec faster than insertion, shuffled (in dark navy), and I think this is due to the sliding effect, where a randomized array would take longer to compare and slide over numbers than a reversed list because a random set needs more moves to slide numbers, as there is not any order present. Bubble, reversed (in red), takes about 60 msec longer to run than bubble, shuffled (in purple), as these two take the longest. The bubble sort method requires comparing two numbers many times in one pass, and this is very time consuming as if it is not in order, the biggest number could take a while to bubble to the end of the array. For example, the reversed array yields 200 msec because it bubbles the first number all the way to the end, and repeats that for the entire length, as the starting index will always be bigger than the next positions. I think the results came out as expected with the tests, and below is a plot of the simple-sort performance.


Simple Sort Performance Comparison
Time (msec)
Input Size (n)
Bubble, Sorted
Bubble, Reversed
Bubble, Shuffled
Insertion, Sorted
Insertion, Reversed
Insertion, Shuffled
Selection, Sorted
Selection, Reversed
Selection, Shuffled