Arda Önal

21903350

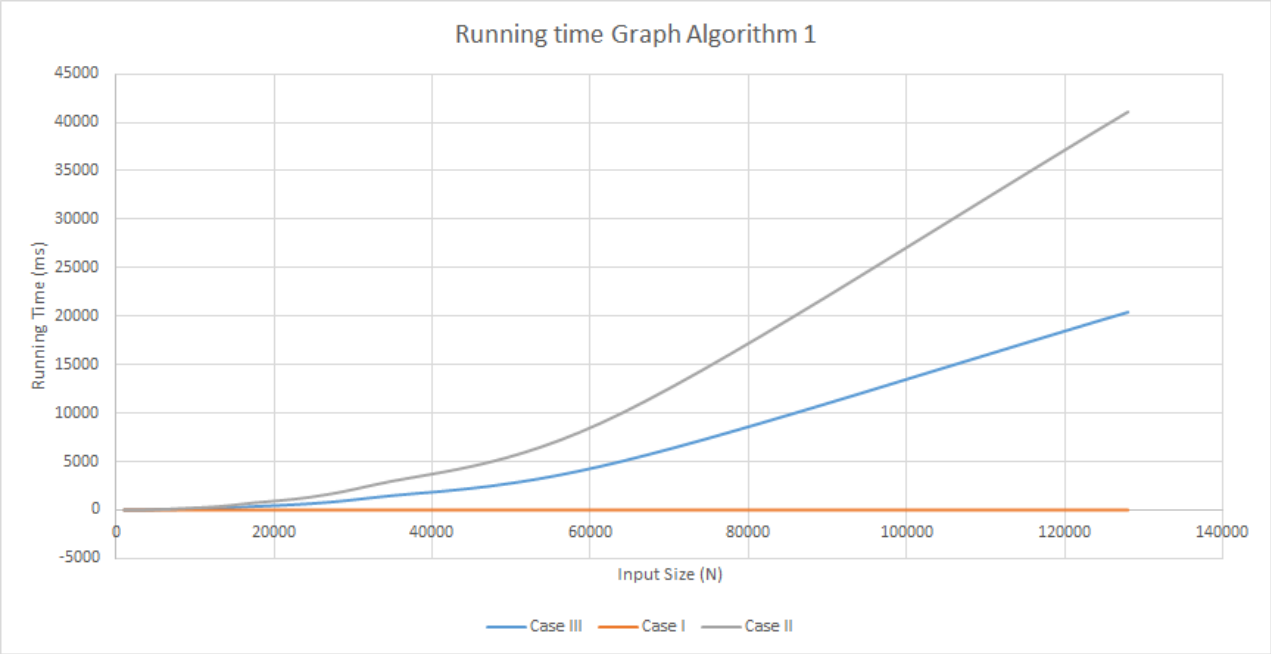
CS 201-01

CS 201, Fall 2020

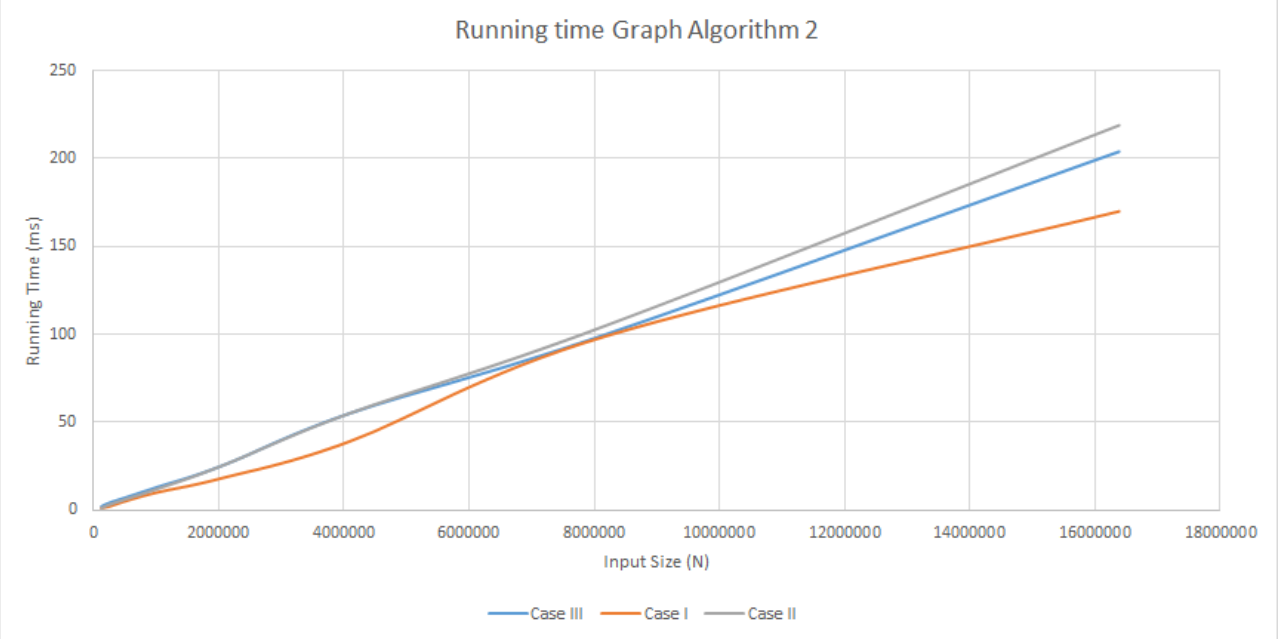
Homework Assignment 2

2.

Algorithm 1:



Algorithm 2:



Case(i): all numbers in arr1 are smaller than arr2

Case(ii): all numbers in arr2 are smaller than arr1

Case(iii): there is no such ordering between arr1 and arr2

3.

Algorithm 1:

Best case: Case(i)

Average case: Case(iii)

Worst case: Case(ii)

Corresponding worst case time complexity: $O(N^2)$

Algorithm 2:

Best case: Case(i)

Average case: Case(iii)

Worst case: Case(ii)

Corresponding worst case time complexity: O(N)

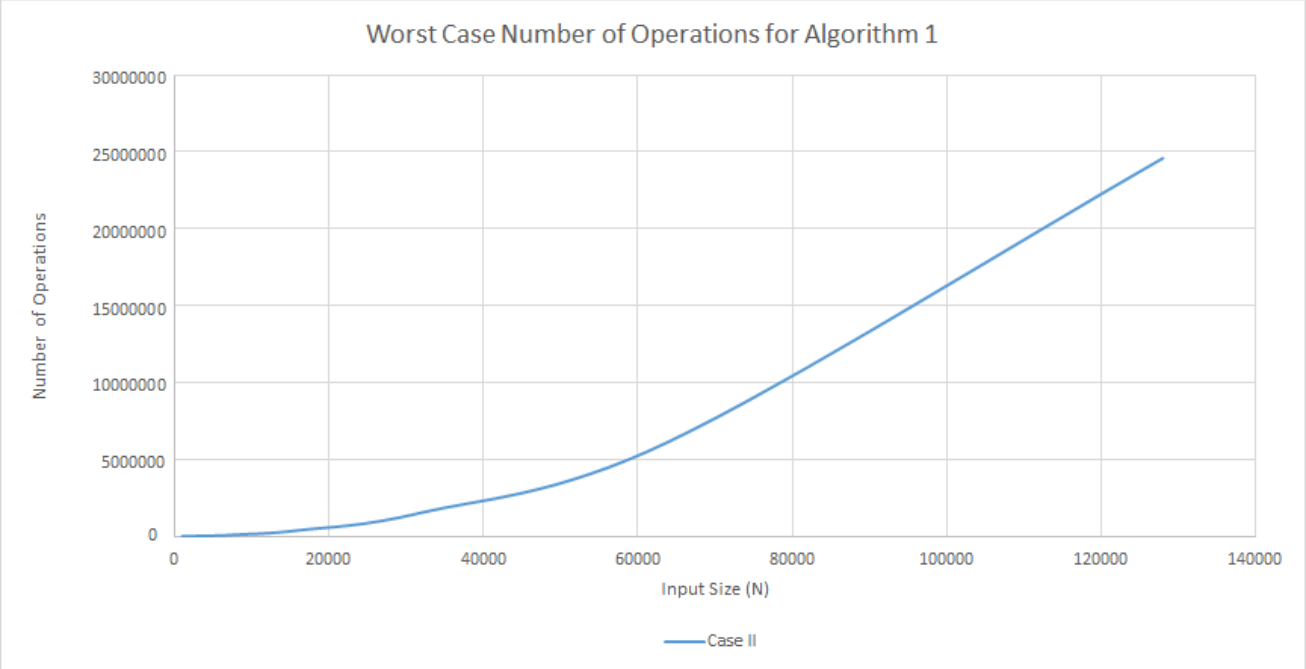
4.

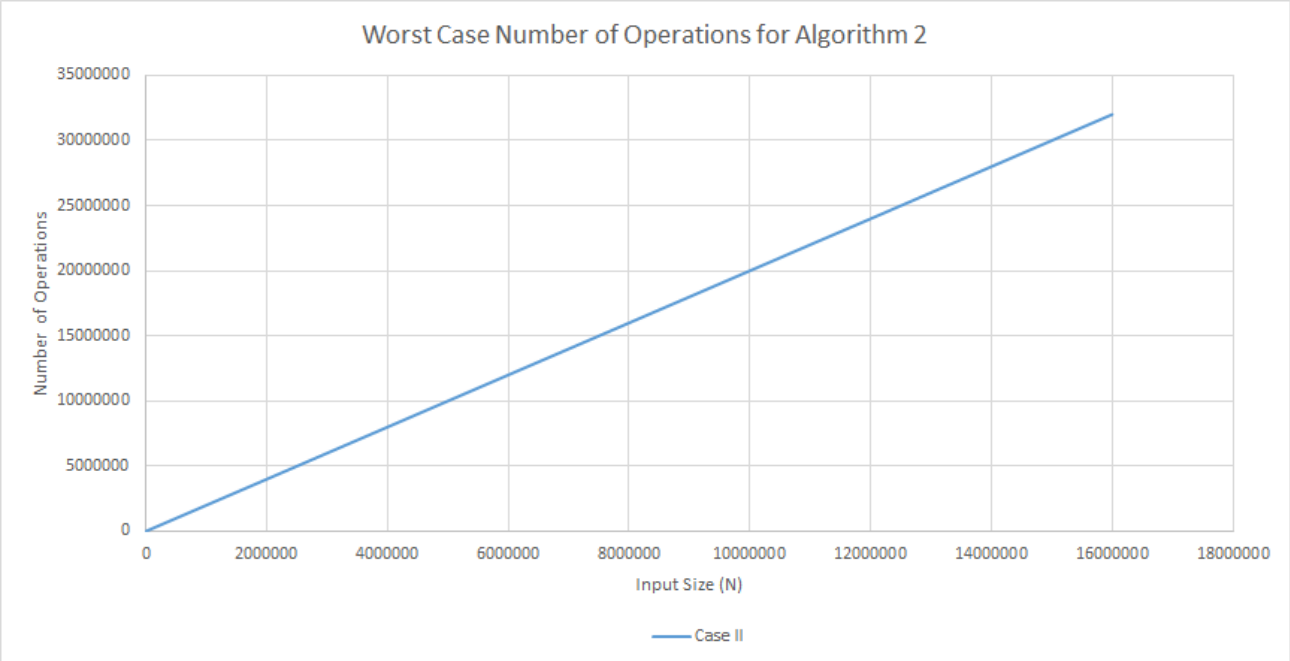
Processor: Intel® Core™ i5-4460 CPU @ 3.20 GHz

RAM: 16.0 GB

Operating system: Windows 8.1 Pro 64 bit

5. Expected worst case growth rate graphs:





6. Comparison of the expected growth rates obtained in step 5 and the worst case results obtained in step 3:

If we look at algorithm 1, the running time graph of the worst case looks like an $O(N^2)$ time complexity. This is because it is curved. If we think about how many operations the worst case

does, we find that approximately N^2 operations are made and the expected growth rate plot looks like an $O(N^2)$ time complexity graph. Therefore, the result I have found for algorithm 1 in the step 3 is very similar to the results in step5. If we look at algorithm 2, the runing time graph of the worst case looks like an O(N) time complexity. This is because it is linear. If we think about how many operations the worst case does, we find hat it does 2N operations and the expected growth rate plot looks like an O(N) time complexity graph. Therefore, the result I have found for algorithm 2 in the step3 is very similar to the result in step5. Finally, if we compare the first and the second algorithm, except the best case of algorithm 1, algorithm 2 is much faster than algorithm 1 on average and worst cases.