INFO 6205-PROGRAM STRUCTURE AND ALGORITHMS ASSIGNMENT-3

Sai Kashyap Cheruku
NU-ID- 002756849

Task:

The task at hand is to observe the running times of a Insertion Sort algorithm, for random, partially sorted, sorted and Reverse-sorted Arrays.

The timer used in the above analysis is to be configured by the programmer, by writing the methods to return system-tick count and mean millisecond time taken for completing an experiment.

The programmer is also required to write a main class for carrying out the benchmarking process.

Relationship Conclusion:

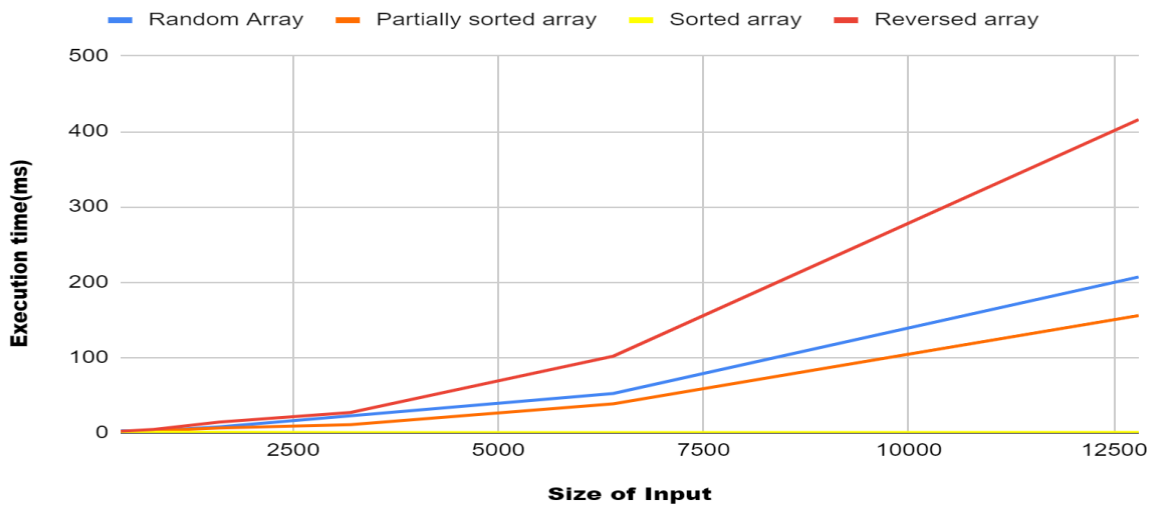
The Insertion Sort algorithm has an average time complexity of $O(N^2)$.

From the benchmark outputs, it is observed that the quadratic nature of insertion sort algorithm is somewhat agnostic to a sorted input. In case of the other input variations, the quadratic nature is visibly evident.
The most time consuming variant is a reverse-sorted array input in which the elements are inserted at their opposite ends to form the sorted output. Such an approach towards an already sorted(without considering the direction of growth) would prove extremely inefficient.
The partially sorted array input performance lies in between the random and reverse-sorted input performances of the algorithm. This is due to the fact that the traversing and swapping operation is only performed on the second half of the array.

Evidence:

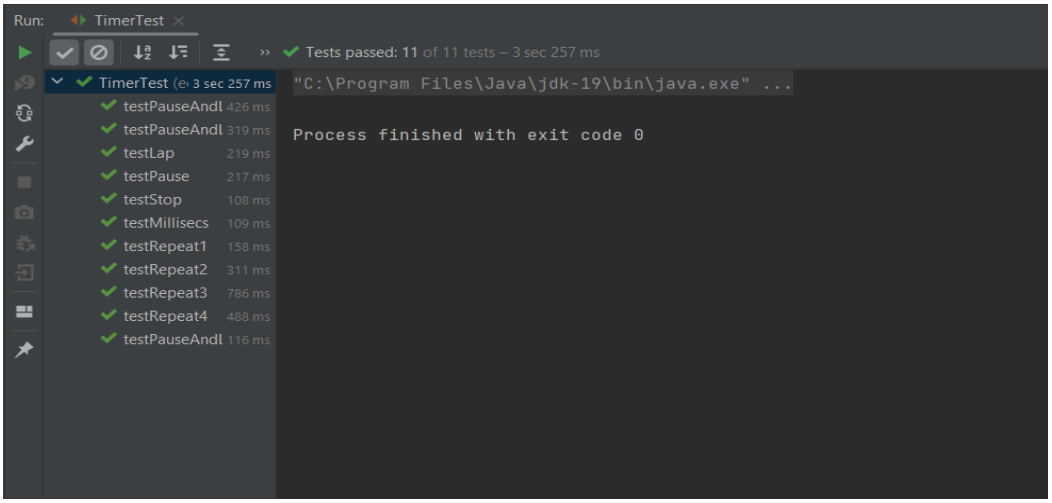
N	Random Array	Partially sorted	Sorted array	Reversed array
400	2.8	1.975	1.25	2.225
800	2.95	2.9	0.8	4.575
1600	8.025	6.475	0.9	14.3
3200	22.575	10.85	0.35	27.125
6400	52.3	38.55	0.375	101.7
12800	206.95	155.7	0.525	415.825

Graphical Representations:

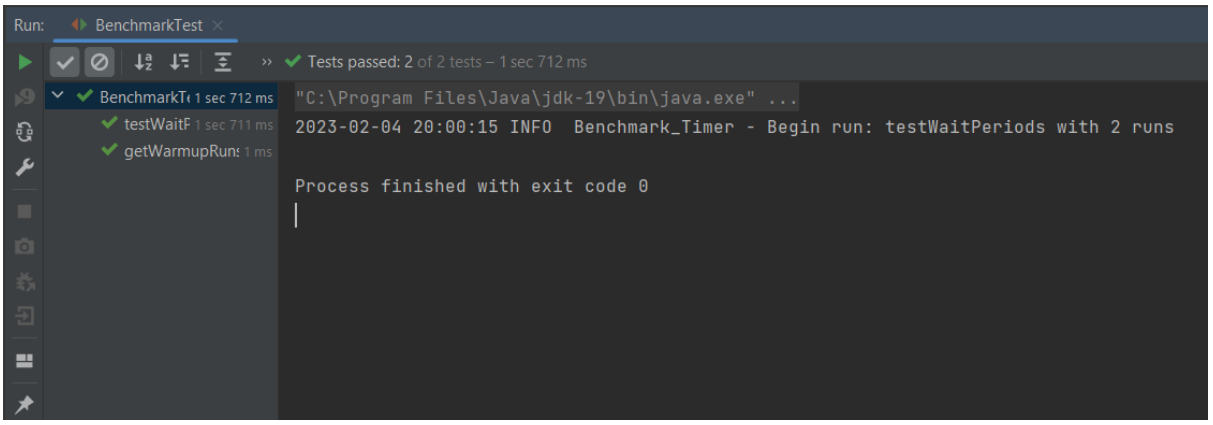


Unit Tests:

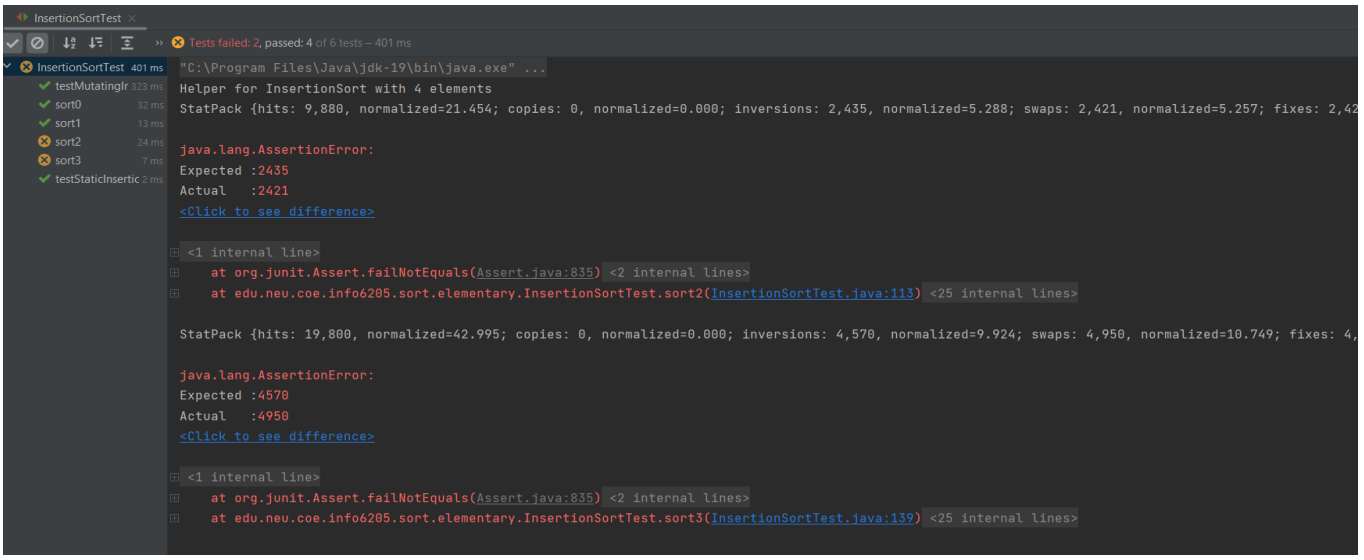
Timer test:



Benchmark Test:



Insertion sort Test:



Main program output:

Started benchmarking procedure
Random Array Sorting 1 n = 400
2023-02-04 19:21:37 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 2.8
Partially Sorted Array Sorting 1 n = 400
2023-02-04 19:21:37 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 1.975
Fully Sorted Array Sorting 1 n = 400
2023-02-04 19:21:37 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 1.25
Reverse Sorted Array Sorting 1 n = 400
2023-02-04 19:21:37 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 2.225
Random Array Sorting 2 n = 800
2023-02-04 19:21:38 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs

Mean lap time: 2.95
Partially Sorted Array Sorting 2 n = 800
2023-02-04 19:21:38 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 2.9
Fully Sorted Array Sorting 2 n = 800
2023-02-04 19:21:38 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 0.8
Reverse Sorted Array Sorting 2 n = 800
2023-02-04 19:21:38 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 4.575
Random Array Sorting 3 n = 1600
2023-02-04 19:21:38 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 8.025
Partially Sorted Array Sorting 3 n = 1600
2023-02-04 19:21:38 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 6.475
Fully Sorted Array Sorting 3 n = 1600
2023-02-04 19:21:39 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 0.9
Reverse Sorted Array Sorting 3 n = 1600
2023-02-04 19:21:39 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 14.3
Random Array Sorting 4 n = 3200
2023-02-04 19:21:39 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 22.575
Partially Sorted Array Sorting 4 n = 3200
2023-02-04 19:21:40 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 10.85
Fully Sorted Array Sorting 4 n = 3200
2023-02-04 19:21:41 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 0.35
Reverse Sorted Array Sorting 4 n = 3200
2023-02-04 19:21:41 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 27.125
Random Array Sorting 5 n = 6400
2023-02-04 19:21:42 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 52.3
Partially Sorted Array Sorting 5 n = 6400
2023-02-04 19:21:44 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 38.55
Fully Sorted Array Sorting 5 n = 6400
2023-02-04 19:21:46 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 0.375
Reverse Sorted Array Sorting 5 n = 6400
2023-02-04 19:21:46 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 101.7
Random Array Sorting 6 n = 12800
2023-02-04 19:21:51 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs

Mean lap time: 206.95
Partially Sorted Array Sorting 6 n = 12800
2023-02-04 19:22:00 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 155.7
Fully Sorted Array Sorting 6 n = 12800
2023-02-04 19:22:07 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 0.525
Reverse Sorted Array Sorting 6 n = 12800
2023-02-04 19:22:07 INFO Benchmark_Timer - Begin run: Insertion Sort with 40 runs
Mean lap time: 415.825
Benchmarking completed successfully