

# Sunil Yadav (NUID: 001492711)

## Program Structures & Algorithms

### Fall 2020

### Assignment 2

For Part 1 and part 2 code has been pushed to github.com

Link: <https://github.com/Sunil-Y/INFO6205/tree/assignment2>

#### • Task

Measure the running times of Insertion sort, using four different initial array ordering situations:

1. Ordered integer array
2. Random ordered integer array
3. Partially – Ordered integer array
4. Reversed – Ordered integer array

#### • Output

##### 1. Ordered integer array:

for No of Integers = 2000, Average time: 0.011111111111111111 ms  
for No of Integers = 4000, Average time: 0.033333333333333333 ms  
for No of Integers = 8000, Average time: 0.066666666666666667 ms  
for No of Integers = 16000, Average time: 0.1 ms  
for No of Integers = 32000, Average time: 0.266666666666666666 ms

##### 2. Random ordered integer array:

for No of Integers = 2000, Average time: 8.5 ms  
for No of Integers = 4000, Average time: 21.933333333333334 ms  
for No of Integers = 8000, Average time: 93.1 ms  
for No of Integers = 16000, Average time: 376.5 ms  
for No of Integers = 32000, Average time: 2163.2333333333333 ms

### 3. Partially – Ordered integer array:

for No of Integers = 2000, Average time: 5.366666666666666 ms  
for No of Integers = 4000, Average time: 15.833333333333334 ms  
for No of Integers = 8000, Average time: 62.733333333333334 ms  
for No of Integers = 16000, Average time: 249.76666666666668 ms  
for No of Integers = 32000, Average time: 1230.9666666666667 ms

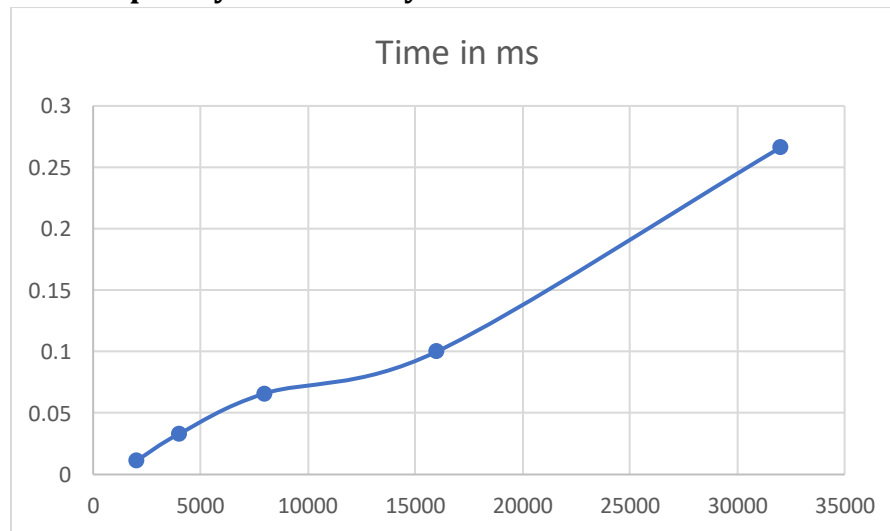
### 4. Reversed – Ordered integer array:

for No of Integers = 2000, Average time: 12.866666666666667 ms  
for No of Integers = 4000, Average time: 40.03333333333333 ms  
for No of Integers = 8000, Average time: 163.13333333333333 ms  
for No of Integers = 16000, Average time: 667.5 ms  
for No of Integers = 32000, Average time: 3327.6666666666665 ms

## • Relationship conclusion

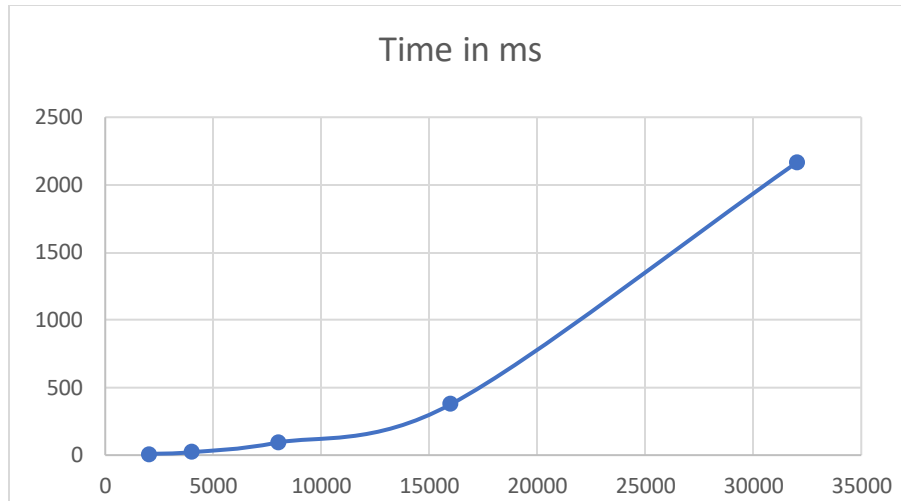
Observations regarding the order of growth:

### 1. For completely sorted array:



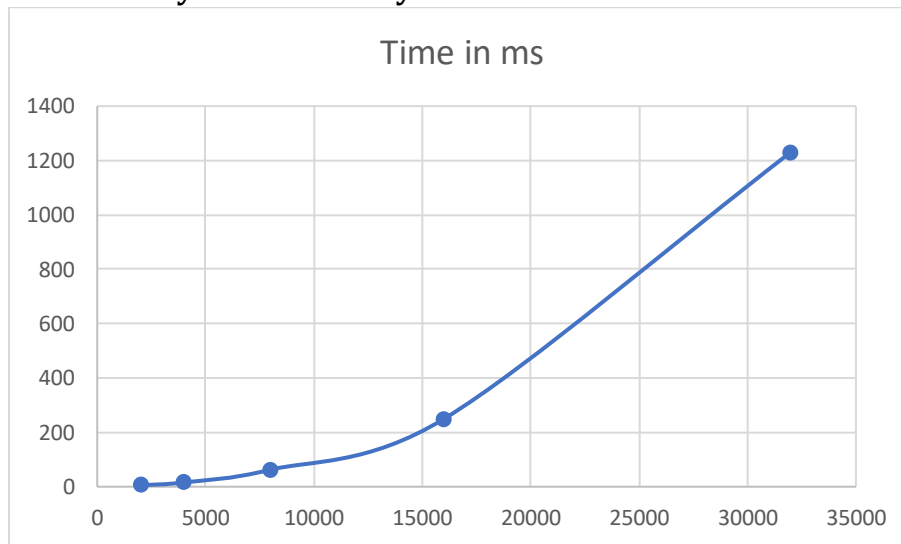
For best case time complexity of Insertion sort for sorted array:  $O(n)$

### 2. For Random ordered integer array:



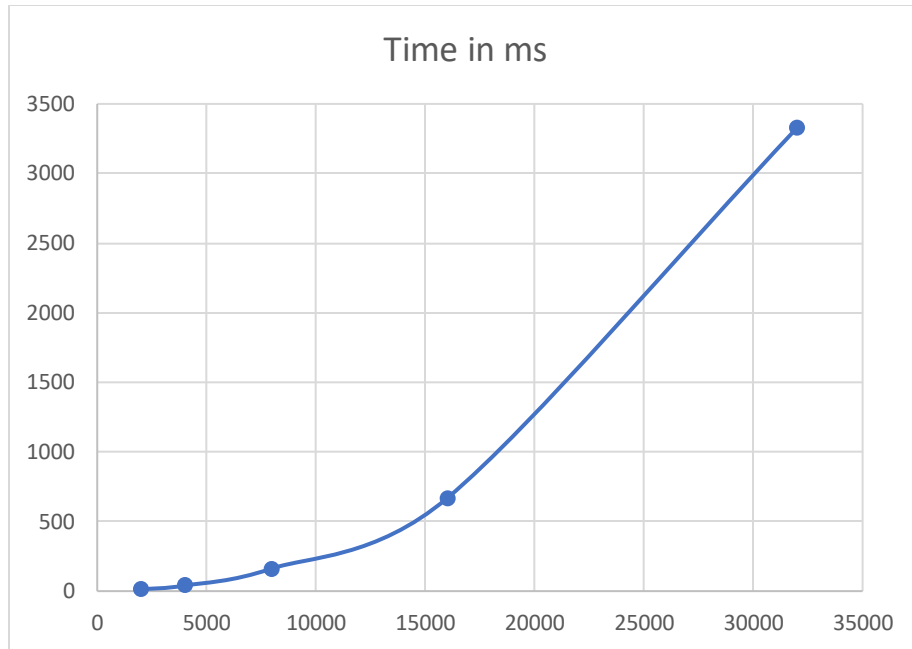
Time complexity for random ordered integer array:  $O(n^2)$

**3. For Partially Ordered array:**



Time complexity for random ordered integer array:  $O(n^2)$

**4. For Reversed Ordered array:**



Time complexity for random ordered integer array:  $O(n^2)$

## • Evidence to support relationship

```
C:\Users\yadav\.jdk\openjdk-15\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.2.2\lib\idea_r
```

```
2020-09-28 21:20:15 INFO Benchmark_Timer - Begin run: Benchmarking for insertion Sort: No. of Integers = 2000 with 30 runs
2020-09-28 21:20:15 INFO Benchmark_Timer - Begin run: Benchmarking for insertion Sort: No. of Integers = 2000 with 30 runs
2020-09-28 21:20:15 INFO Benchmark_Timer - Begin run: Benchmarking for insertion Sort: No. of Integers = 2000 with 30 runs
2020-09-28 21:20:16 INFO Benchmark_Timer - Begin run: Benchmarking for insertion Sort: No. of Integers = 2000 with 30 runs
```

```
--- =====
for No of Integers = 2000
Average time for : Sorted Array = 0.1333333333333333 ms
Average time for : Random Array = 8.5 ms
Average time for : Partially Ordered Array = 5.366666666666666 ms
Average time for : Reversed Ordered Array = 12.866666666666667 ms
```

```
--- =====
for No of Integers = 4000
Average time for : Sorted Array = 0.03333333333333333 ms
Average time for : Random Array = 21.933333333333334 ms
--- =====
```

```
--- =====
for No of Integers = 16000
Average time for : Sorted Array = 0.1 ms
Average time for : Random Array = 376.5 ms
Average time for : Partially Ordered Array = 249.76666666666668 ms
Average time for : Reversed Ordered Array = 667.5 ms
```

```
2020-09-28 21:21:12 INFO Benchmark_Timer - Begin run: Benchmarking for insertion Sort: No. of Integers = 32000 with 30 runs
2020-09-28 21:21:12 INFO Benchmark_Timer - Begin run: Benchmarking for insertion Sort: No. of Integers = 32000 with 30 runs
2020-09-28 21:22:22 INFO Benchmark_Timer - Begin run: Benchmarking for insertion Sort: No. of Integers = 32000 with 30 runs
2020-09-28 21:23:03 INFO Benchmark_Timer - Begin run: Benchmarking for insertion Sort: No. of Integers = 32000 with 30 runs
```

- **Screenshot of Unit test passing**

```
✓ BenchmarkTest (edu.neu.coe.info6205.util) 1 s 526 ms C:\Users\yadav\.jdk\openjdk-15\bin\java.exe ...
  ✓ testWaitPeriods 1 s 526 ms 2020-09-28 22:21:17 INFO Benchmark_Timer - Begin run: testWaitPeriods with 2 runs
  ✓ getWarmupRuns 0 ms Process finished with exit code 0
```

```
✓ TimerTest (edu.neu.coe.info6205.util) 2 s 597 ms C:\Users\yadav\.jdk\openjdk-15\bin\java.exe ...
  ✓ testPauseAndLapResume0 242 ms
  ✓ testPauseAndLapResume1 333 ms Process finished with exit code 0
  ✓ testLap 222 ms
  ✓ testPause 221 ms
  ✓ testStop 107 ms
  ✓ testMillisecs 110 ms
  ✓ testRepeat1 156 ms
  ✓ testRepeat2 313 ms
  ✓ testRepeat3 783 ms
  ✓ testPauseAndLap 110 ms
```