**Program Structures & Algorithms**

**Spring 2022**

**Assignment No. 2**

Name: Sushmita Maity

(NUID): 001092534

**Task  
1)** To implement *repeat()*, *getClock()*, and *toMillisecs()* of *Time class, which is invoked in Benchmark\_Timer class*.

2) Implementing sort method using helper class.

3) Drawing conclusion for random, ordered, partially ordered and reverse ordered array and time taken to sort them using the above implemented method.

* **Output screenshot**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

* **Relationship Conclusion**

1. **Reverse ordered Array takes the maximum time for sorting, following by Randomly ordered array, partially ordered array and then ordered array.**
2. **For ordered array we can conclude that it takes approximately O(n) times, while for reverse ordered array O(n^2) time.**
3. **As we can observe from the graphs, as the array input increases the graph imitates a quadratic growth graph, therefore we can say that for non-ordered arrays, time = (array input)^2.**

* **Evidence / Graph**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Random Array | Ordered Array | Partialy Ordered Array | Reverse Ordered Array |
| Array | Random Arr Time | Ordered Arr Time | Partially Ordered Time | Reversed Arr Time |
| 50 | 0.00401581 | 4.91E-04 | 4.80E-04 | 5.16E-04 |
| 100 | 0.00505212 | 7.84E-04 | 9.03E-04 | 6.72E-04 |
| 200 | 0.0025165 | 0.00124755 | 0.00137593 | 0.00107899 |
| 400 | 0.00430665 | 0.00153642 | 0.00150477 | 0.00177874 |
| 800 | 0.00825466 | 0.00291174 | 0.00294285 | 0.00328126 |
| 1600 | 0.07190395 | 0.00563587 | 0.00591497 | 0.00613369 |
| 3200 | 0.01115812 | 0.01136864 | 0.01116502 | 0.02359319 |

|  |
| --- |
|  |

* **Unit tests result**

Benchmark Test Screenshot :

Graphical user interface, text

Description automatically generated

Insertion Sort Test Screenshot :

Text

Description automatically generated

Timer Test screenshot:

Text

Description automatically generated