**PSA ASSIGNMENT 3**

YASH KHOPKAR

001850102

1. The complexity for each selection and Insertion sort is O(n^2).
2. I ran this sorting algorithms for 1000 (m) times for Random, ordered, reverse ordered, and partially ordered array each for Selection and Insertion Sort (with changing n value). The output is in milliseconds and listed in the table shown above.
3. Selection Sort is taking more time always than Insertion sort. In each iteration the value of n is getting changed and goes like 500, 1000, 2000, 4000, 8000, 16000.
4. The time required for sorting the arrays depends upon the value of n and number of repetitions done for taking average value of time.
5. The time difference for each value of n is gradually increasing for selection sort, whereas there is not much difference for insertion Sort.
6. From the results we can infer that Insertion sort gives best performance when the input array is Partially Sorted Array.
7. Insertion sort takes less time when the array is sorted as compared to partially sorted and unsorted array.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Ordered Array | | Reverse-Ordered Array | | Random Array | | Partially-Sorted Array | |
| N | Selection | Insertion | Selection | Insertion | Selection | Insertion | Selection | Insertion |
| 500 | 0.1241502 | 0.020610772 | 0.117383145 | 0.021836815 | 0.20086537 | 0.28957906 | 0.225984 | 0.28293896 |
| 1000 | 0.47144127 | 0.0192405 | 0.46628317 | 0.02316372 | 0.7057269 | 1.0611451 | 0.7484866 | 1.0682489 |
| 2000 | 1.9172223 | 0.019994406 | 1.7814615 | 0.024242714 | 2.6467528 | 4.4280667 | 2.6897972 | 4.3017507 |
| 4000 | 7.0724216 | 0.023066396 | 7.0109243 | 0.027530154 | 10.025859 | 17.172014 | 10.435096 | 17.654604 |
| 8000 | 27.933306 | 0.029038653 | 29.079866 | 0.033758078 | 46.188145 | 70.33202 | 48.912094 | 72.18191 |
| 1600.0E | 113.0301 | 0.042075578 | 112.9757 | 0.046995297 | 215.80513 | 3  3  1  .  8  8  4  8  3 |  |  |

SCREENSHOTS:







