|  |  |
| --- | --- |
| |  | | --- | | **Sorting Algorithm Report** | |

**Selection Sort:**

1. I have run this algorithm for two different types of array

* Random Array(Average Case)
* Sorted Array(Best Case)

**Table for random Array input Array**

|  |  |
| --- | --- |
| No of runs | Execution Time |
| 1 | 571059218 |
| 2 | 568884870 |
| 3 | 647453253 |
| 4 | 648242419 |
| 5 | 647710504 |
| 6 | 646482416 |
| 7 | 646446423 |
| 8 | 741228970 |
| 9 | 692595675 |
| 10 | 667718625 |
| Average time Taken | 647782237 |

**Table for Sorted input Array**

|  |  |
| --- | --- |
| No of runs | Execution Time |
| 1 | 628497812 |
| 2 | 614844158 |
| 3 | 671383062 |
| 4 | 694766527 |
| 5 | 665242432 |
| 6 | 658033374 |
| 7 | 669272432 |
| 8 | 658003374 |
| 9 | 664242422 |
| 10 | 634847158 |
| Average time Taken | 655913275 |

**Bubble sort:**

1. I have run this algorithm for two different types of array

* Random Array(Average Case)
* Sorted Array(Best Case)

|  |  |
| --- | --- |
| No of runs | Execution Time |
| 1 | 3052451362 |
| 2 | 2697120809 |
| 3 | 2690096950 |
| 4 | 3760200410 |
| 5 | 2610425636 |
| 6 | 2607576253 |
| 7 | 2610737242 |
| 8 | 2628238472 |
| 9 | 2609096189 |
| 10 | 2773341682 |
| Average time Taken | 2803928500 |

**Table for random Array input Array**

**Table for Sorted input Array**

|  |  |
| --- | --- |
| No of runs | Execution Time |
| 1 | 1470 |
| 2 | 1483 |
| 3 | 1566 |
| 4 | 1693 |
| 5 | 1392 |
| 6 | 1401 |
| 7 | 1525 |
| 8 | 1560 |
| 9 | 1510 |
| 10 | 1453 |
| Average time Taken | 1505 |

**Insertion Sort:**

1. I have run this algorithm for two different types of array

* Random Array(Average Case)
* Sorted Array(Best Case)

**Table for random Array input Array**

|  |  |
| --- | --- |
| No of runs | Execution Time |
| 1 | 384360590 |
| 2 | 386451198 |
| 3 | 376189221 |
| 4 | 375347226 |
| 5 | 376818237 |
| 6 | 375840283 |
| 7 | 375914061 |
| 8 | 375787609 |
| 9 | 376480321 |
| 10 | 378873414 |
| Average time Taken | 378206216 |

**Table for Sorted input Array**

|  |  |
| --- | --- |
| No of runs | Execution Time |
| 1 | 2519 |
| 2 | 2500 |
| 3 | 3044 |
| 4 | 2680 |
| 5 | 2384 |
| 6 | 2358 |
| 7 | 3936 |
| 8 | 2744 |
| 9 | 3953 |
| 10 | 4195 |
| Average time Taken | 2931 |

* Which Algorithm perform best under which condition?

Bubble Algorithm was taking less execution time in case of sorted array because there is a flag in algorithm which checks if no swap occurs then loop breaks

* Which are the worse

Selection sort was worse. It was taking much time in case of Random Array as well as Sorted Array. Because there is no way in to get out from loop before the condition becomes false.