S69: Project.

Insertion:

|  |  |  |
| --- | --- | --- |
|  | BST (nano seconds) | Ordered Array (nano seconds) |
| 1000 (Average) | 321 | 10095 |
| 10000 (Average) | 3608 | 475458 |
| 100000 (Average) | 243779 | 1059724 |
| 1000000 (Worst) | Took too long | Took too long |
|  |  |  |
|  |  |  |

Searching:

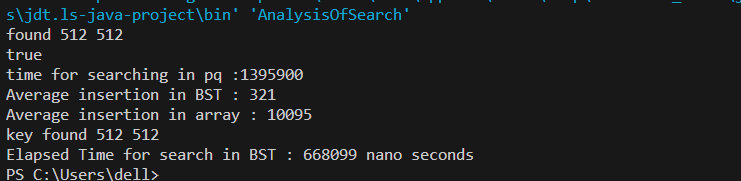
|  |  |  |
| --- | --- | --- |
|  | BST (nano seconds) | Ordered array (nano seconds) |
| 1000 (Average) | 668099 | 139590 |
| 10000 (Average) | 411599 | 949399 |
| 100000 (Average) | 18617300 | 12621400 |
| 1000000 (worst) | 27684500 | Took too long |

From above observations we can say that BST performs better for insertion and searching operations.

As the data set size increases the time for performing insertion and searching operation also increases.

Theoretically, Ordered array can be used for small data set as it is easy to insert elements in sorted way. But as data size increases BST will perform better.

1000 elements output:



10000 elements output:

