**Sorting** **Algorithm Analysis**

Sorting Analysis of Binary Search Tree and Red Black on running different input on different sorting order.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Input Size** | **Sorting order** | **Sorting using Binary Search Tree (in ms)** | | **Sorting using Red-Black Tree (in ms)** | |
| 50000 | 0 | 51 | Duplicates: 2 | 41 | Case 1: 25634  Case 2: 9726  Case 3: 19401  Left Rotate: 14585  Right Rotate: 14542  Duplicates: 0 |
| -1 | 4649 | Duplicates: 0 | 33 | Case 1: 49966  Case 2: 0  Case 3: 49971  Left Rotate: 0  Right Rotate: 49971  Duplicates: 0 |
| 1 | 4798 | Duplicates: 0 | 28 | Case 1: 49966  Case 2: 0  Case 3: 49971  Left Rotate: 49971  Right Rotate: 0  Duplicates: 0 |
| 100000 | 0 | 94 | Duplicates: 4 | 82 | Case 1: 51359  Case 2: 19375  Case 3: 38725  Left Rotate: 28952  Right Rotate: 29148  Duplicates: 0 |
| -1 | 25111 | Duplicates: 0 | 68 | Case 1: 99964  Case 2: 0  Case 3: 99969  Left Rotate: 0  Right Rotate: 99969  Duplicates: 0 |
| 1 | 28989 | Duplicates: 0 | 65 | Case 1: 99964  Case 2: 0  Case 3: 99969  Left Rotate: 99969  Right Rotate: 0  Duplicates: 0 |
| 170000 | 0 | 198 | Duplicates: 11 | 174 | Case 1: 87250  Case 2: 33108  Case 3: 66230  Left Rotate: 49592  Right Rotate: 49746  Duplicates: 9 |
| -1 | 89657 | Duplicates: 0 | 92 | Case 1: 169963  Case 2: 0  Case 3: 169968  Left Rotate: 0  Right Rotate: 169968  Duplicates: 0 |
| 1 | 98719 | Duplicates: 0 | 105 | Case 1: 169963  Case 2: 0  Case 3: 169968  Left Rotate: 169968  Right Rotate: 0  Duplicates: 0 |
| 250000 | 0 | Taking too much time to execute |  | 231 | Case 1: 128299  Case 2: 48421  Case 3: 97123  Left Rotate: 72767  Right Rotate: 72777  Duplicates: 14 |
| -1 | Taking too much time to execute |  | 131 | Case 1: 249961  Case 2: 0  Case 3: 249967  Left Rotate: 0  Right Rotate: 249967  Duplicates: 0 |
| 1 | Taking too much time to execute |  | 134 | Case 1: 249961  Case 2: 0  Case 3: 249967  Left Rotate: 249967  Right Rotate: 0  Duplicates: 0 |
| 1000000 | 0 | Taking too much time to execute |  | 1182 | Case 1: 513086  Case 2: 193984  Case 3: 388071  Left Rotate: 290927  Right Rotate: 291128  Duplicates: 247 |
| -1 | Taking too much time to execute |  | 436 | Case 1: 999957  Case 2: 0  Case 3: 999963  Left Rotate: 0  Right Rotate: 999963  Duplicates: 0 |
| 1 | Taking too much time to execute |  | 568 | Case 1: 999957  Case 2: 0  Case 3: 999963  Left Rotate: 999963  Right Rotate: 0  Duplicates: 0 |
| 2500000 | 0 | Taking too much time to execute |  | 3829 | Case 1: 1282523  Case 2: 484946  Case 3: 968808  Left Rotate: 727185  Right Rotate: 726569  Duplicates: 1403 |
| -1 | Taking too much time to execute |  | 1310 | Case 1: 2499952  Case 2: 0  Case 3: 2499960  Left Rotate: 0  Right Rotate: 2499960  Duplicates: 0 |
| 1 | Taking too much time to execute |  | 1152 | Case 1: 2499952  Case 2: 0  Case 3: 2499960  Left Rotate: 2499960  Right Rotate: 0  Duplicates: 0 |
| 5000000 | 0 | Taking too much time to execute |  | 9108 | Case 1: 2565221  Case 2: 969705  Case 3: 1939502  Left Rotate: 1454350  Right Rotate: 1454857  Duplicates: 5764 |
| -1 | Taking too much time to execute |  | 2137 | Case 1: 4999950  Case 2: 0  Case 3: 4999958  Left Rotate: 0  Right Rotate: 4999958  Duplicates: 0 |
| 1 | Taking too much time to execute |  | 2123 | Case 1: 4999950  Case 2: 0  Case 3: 4999958  Left Rotate: 4999958  Right Rotate: 0  Duplicates: 0 |

Analysis:

1. Red-Black tree is taking much less time to sort same input as compared to Binary Search Tree.
2. Binary Search is taking very long time to execute large input data while Red- Black is working smoothly on large input data too.
3. For both Red-Black Tree and Binary Search Tree, sorting time depends on input value size. If the input size increases, sorting time increase.
4. For Red Back Tree random input order is taking much more time to sort input than decreasing and increasing input sorting order.
5. For Binary Search Tree Random input order is taking much less time to sort increasing and decreasing input sorting order.