```
preOrder traversal of the constructed tree is
(6, 3)
(2, 5)
(1, 4)
(4, 2)
(3, 7)
(5, 6)
(7, 1)
Inorder traversal of the constructed tree is
(1, 4)
(2, 5)
(3, 7)
(4, 2)
(5, 6)
(6, 3)
(7, 1)
postOrder traversal of the constructed tree is
(7, 1)
(5, 6)
(3, 7)
(4, 2)
(1, 4)
(2, 5)
(6, 3)
InOrder traversal of the copied tree is
(1, 4)
(2, 5)
(3, 7)
(4, 2)
(5, 6)
(6, 3)
(7, 1)
The point (4, 2) is in the tree at index: 4
                (6, 3)
            (2, 5)  (7, 1)
        (1, 4)  (4, 2)  -   -
    -   -   (3, 7)  (5, 6)  -   -   -   -
-   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -
```

```
The tree after deleting (4, 2) is
                (6, 3)
            (2, 5)  (7, 1)
        (1, 4)  (5, 6)  -   -
    -   -   (3, 7)  -   -   -   -   -
-   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -

Successor of root is: 2
Predecessor of root is: 4
The node count is: 6
The size is: 31
The maximum is: (7, 1)
The minimum is: (1, 4)
The nodes in ascending order:
(1, 4)
(2, 5)
(3, 7)
(5, 6)
(6, 3)
(7, 1)
The nodes in descending order:
(7, 1)
(6, 3)
(5, 6)
(3, 7)
(2, 5)
(1, 4)
```

The Code is at: https://github.com/Flips2001/Data-Structure-and-Algorithm/tree/0e75d67f6eb444642694cde337f891b1867afb79/BinarySearchTree/src