

14  
(Q) Sorted array to BST.

Algo:

1. root = middle element of array.

2. root.left =  $\overset{\text{mid}}{\underset{\wedge}{(0 \text{ to } (mid-1))}}$

3. root.right = mid (mid+1 to high)

4. return root.

---

Node toBST (int [] a) {

    int mid = (a.length - 1) / 2;

    Node root = new Node(a[mid]);

    root.left = toBST(a, 0, mid - 1);

    root.right = toBST(a, mid + 1, a.length - 1);

    return root;

}

Node toBST (int [] a, int low, int high) {

    if (low > high) return null;

    int mid = (low + high) / 2;

    Node n = new Node(a[mid]);

    n.left = toBST(a, low, mid - 1);

    n.right = toBST(a, mid + 1, high);

    return n;

}