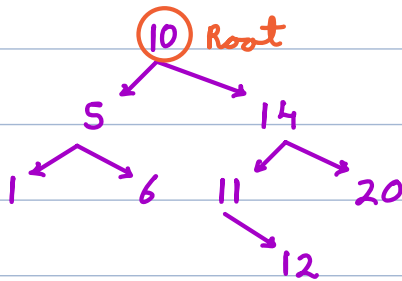


Q → Given a BST & a +ve integer K.

Find K^{th} smallest element in BST.



$K=3 \rightarrow \text{Ans} = \underline{6}$

$K=5 \rightarrow \text{Ans} = \underline{11}$

Bruteforce → Travel the tree then sort the data & find K^{th} element.

inorder

Left Node Right

$TC = \underline{O(N)}$

$SC = \underline{O(H)}$

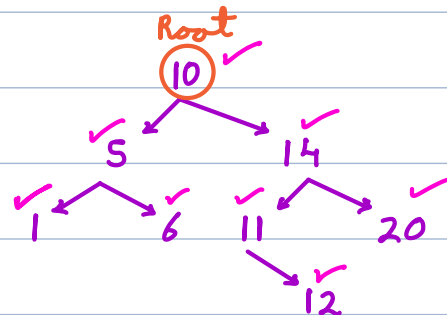
→ Inorder traversal of BST is sorted.

Morris Inorder Traversal

$SC = \underline{O(1)}$

When is 10 req.?

After 6



o/p → 1 5 6 10 11 12 14 20

```
void inorder (root) {
```

```
    cur = root
```

```
    while (cur != null) {
```

```
        if (cur.left == null) {
```

```
            print (cur.data)
```

```
            cur = cur.right
```

```
        } else {
```

```
            p = cur.left
```

```
            while (p.right != null && p.right != cur) {
```

```
                p = p.right
```

```
            }
```

```
            if (p.right == null) {
```

```
                p.right = cur
```

```
                cur = cur.left
```

```
            } else {
```

```
                p.right = null
```

```
                print (cur.data)
```

```
                cur = cur.right
```

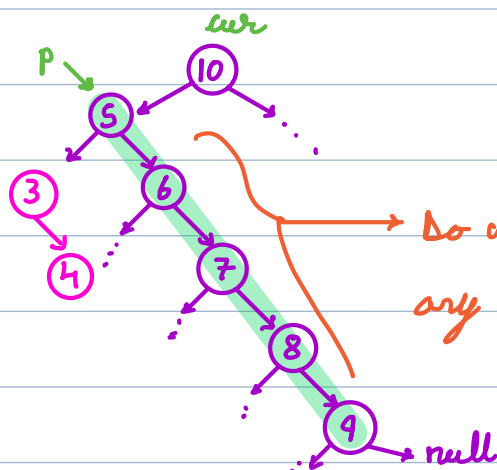
```
            }
```

```
        }
```

```
    }
```

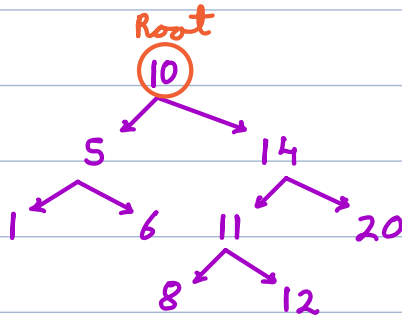
SC = O(1)

TC = O(N)



Do we travel these for any other current? No

Q → Given a binary tree. For a given node print path from root to current node.



path(12) = 10 → 14 → 11 → 12

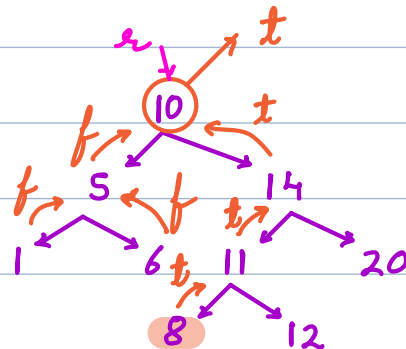
path(5) = 10 → 5

```

boolean findPath (root, x) {
    if (root == null) return false
    if (root.data == x) {
        list.add(root)
        return true
    }
    res = findPath (root.left, x) OR
        findPath (root.right, x)

    if (res) list.add (root)

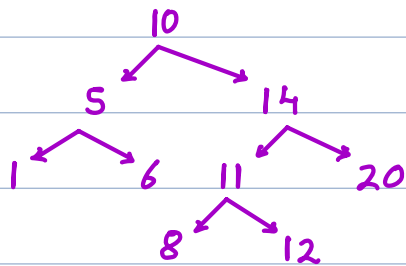
    return res
}
    
```



list → [8 11 14 10]

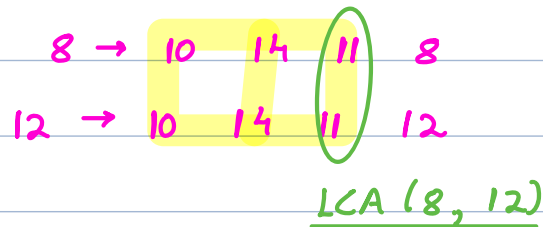
print ans in reverse order wrt list.

$$TC = \underline{O(N)} \quad SC = \underline{O(H)}$$



lowest common Ancestor

→ All nodes from root till cur node.



$$LCA(1, 14) \rightarrow 10$$

$$LCA(8, 20) \rightarrow 14$$

$$LCA(8, 14) = 14$$

$$LCA(6, 6) = 6$$

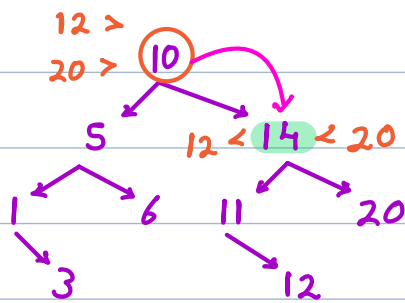
Sol → $LCA(x, y)$

- 1) Find path from root to x & root to y.
- 2) Travel till we have common ancestor.
- 3) Ans = last common ancestor.

$$TC = O(N + N + H) \rightarrow \underline{O(N)}$$

$$SC = \underline{O(H)}$$

Q → Find LCA x & y in the given BST.



$$\text{LCA}(12, 20) = 14$$

$\text{cur} = \text{root}$

while ($\text{cur} \neq \text{null}$) {

if ($\text{cur}.\text{data} > x$ && $\text{cur}.\text{data} > y$)

$\text{cur} = \text{cur}.\text{left}$

else if ($\text{cur}.\text{data} < x$ && $\text{cur}.\text{data} < y$)

$\text{cur} = \text{cur}.\text{right}$

else

return cur

}

$$\text{TC} = \underline{O(H)} \quad \text{SC} = \underline{O(1)}$$