

## Java

```
/**
 * Definition for a binary tree node.
 * public class TreeNode {
 *     int val;
 *     TreeNode left;
 *     TreeNode right;
 *     TreeNode(int x) { val = x; }
 * }
 */
class Solution {
    public List<Integer> distanceK(TreeNode root, TreeNode target, int k) {

    }
}
```

---

## JavaScript

```
/**
 * Definition for a binary tree node.
 * function TreeNode(val) {
 *     this.val = val;
 *     this.left = this.right = null;
 * }
 */
/**
 * @param {TreeNode} root
 * @param {TreeNode} target
 * @param {number} k
 * @return {number[]}
 */
```

```
*/  
var distanceK = function(root, target, k) {  
  
};  
-----
```

## C++

```
/**  
 * Definition for a binary tree node.  
 * struct TreeNode {  
 *     int val;  
 *     TreeNode *left;  
 *     TreeNode *right;  
 *     TreeNode(int x) : val(x), left(NULL), right(NULL) {}  
 * };  
 */  
class Solution {  
public:  
    vector<int> distanceK(TreeNode* root, TreeNode* target, int k) {  
  
    }  
};  
-----
```

## C#

```
/**  
 * Definition for a binary tree node.  
 * public class TreeNode {  
 *     public int val;  
 *     public TreeNode left;  
 *     public TreeNode right;  
 * }
```

```
*     public TreeNode right;
*     public TreeNode(int x) { val = x; }
* }
*/
public class Solution {
    public IList<int> DistanceK(TreeNode root, TreeNode target, int k) {

    }
}
```

---