LEET CODE – Two Sum IV - Input is a BST

Given the root of a binary search tree and an integer k, return true if there exist two elements in the BST such that their sum is equal to k, or false otherwise.

CODE:

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

 bool findTarget(struct TreeNode* root, int k) {
if (root == NULL) {
return false;
}

void inOrderTraversal(struct TreeNode* root, int* arr, int* index) {
if (root == NULL) {
return;
}
inOrderTraversal(root->left, arr, index);
arr[(*index)++] = root->val;
inOrderTraversal(root->right, arr, index);
}

int numNodes = 0;
struct TreeNode* temp = root;
struct TreeNode* stack[100];
int stackSize = 0;

while (temp != NULL || stackSize > 0) {
while (temp != NULL) {
stack[stackSize++] = temp;
temp = temp->left;
}
temp = stack[--stackSize];
numNodes++;
temp = temp->right;
}

int* arr = (int*)malloc(numNodes * sizeof(int));
int index = 0;
inOrderTraversal(root, arr, &index);

int left = 0;
```

```
int right = numNodes - 1;

while (left < right) {
int sum = arr[left] + arr[right];
if (sum == k) {
free(arr);
return true;
} else if (sum < k) {
left++;
} else {
right--;
}
}

free(arr);
return false;
}
```

OUTPUT :