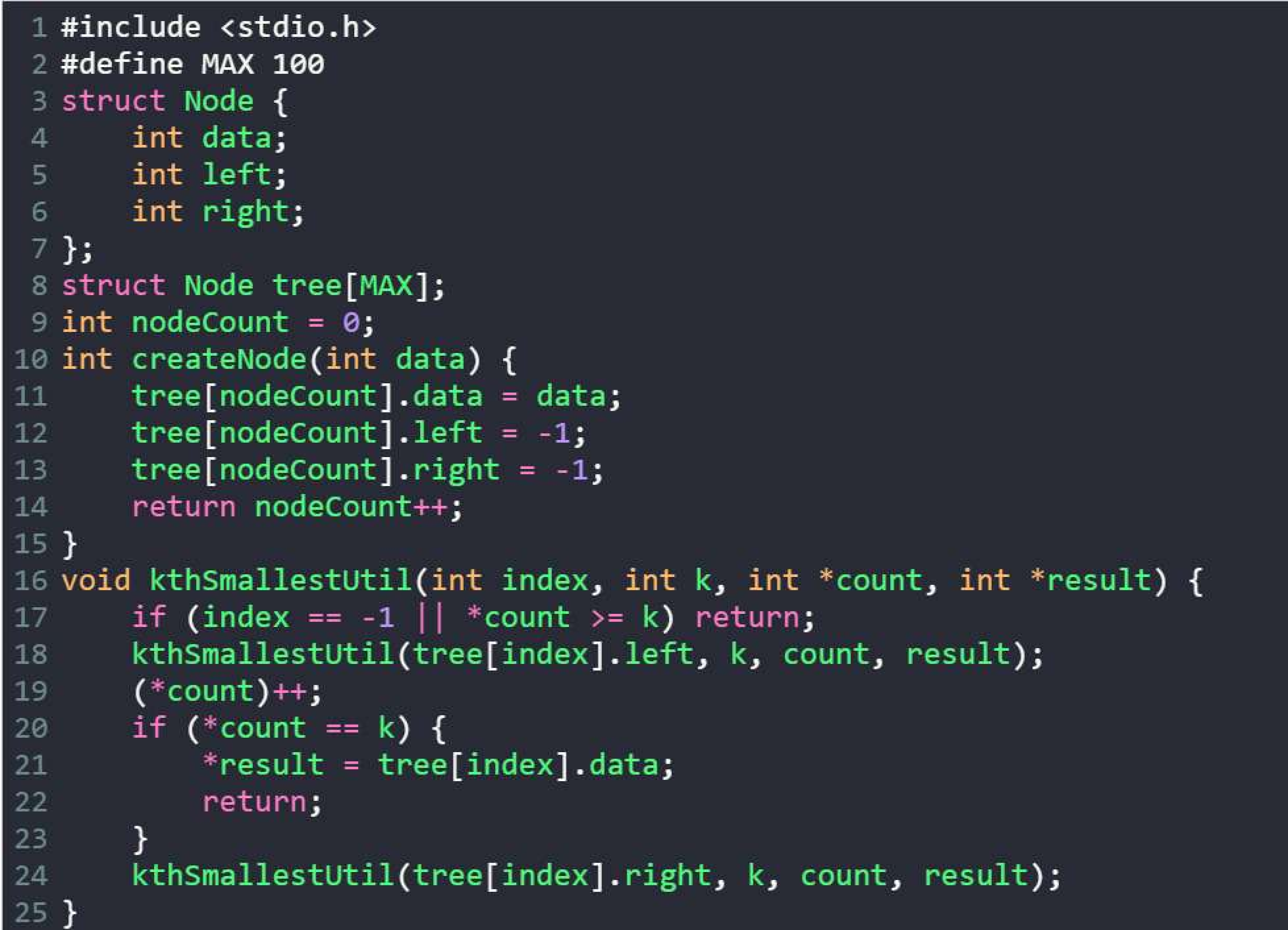
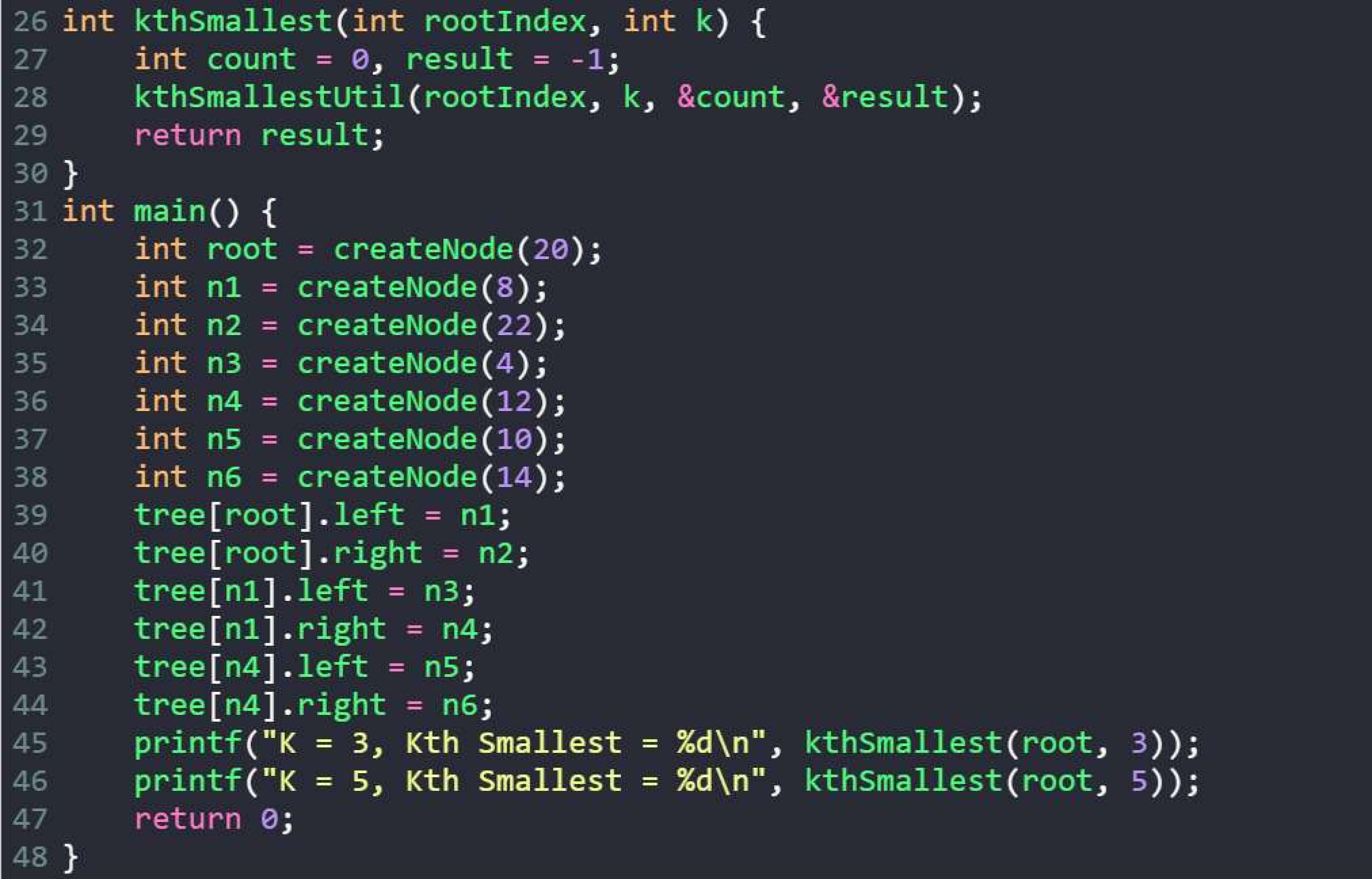
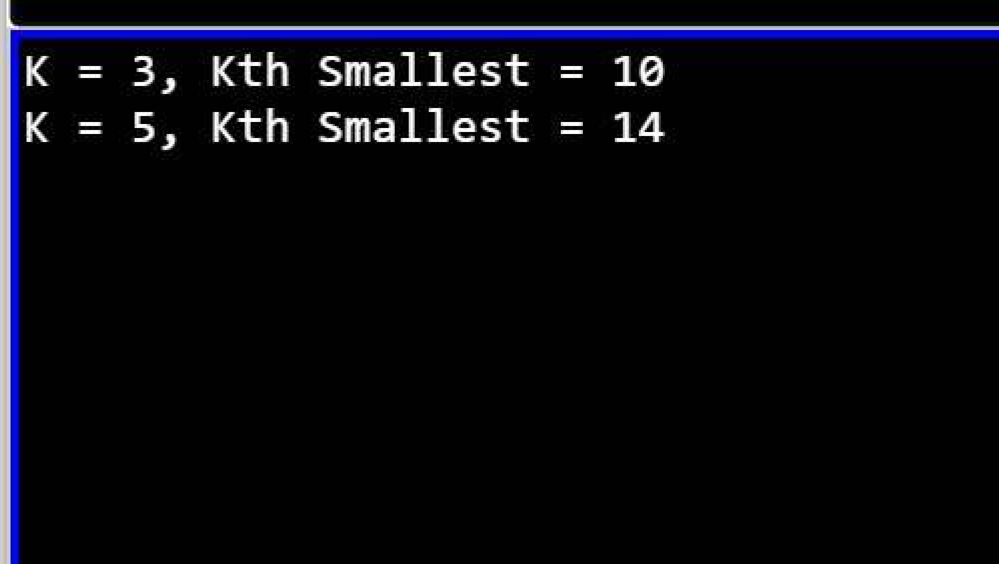


```c
#include <stdio.h>
#define MAX 100
struct Node {
    int data;
    int left;
    int right;
};
struct Node tree[MAX];
int nodeCount = 0;
int createNode(int data) {
    tree[nodeCount].data = data;
    tree[nodeCount].left = -1;
    tree[nodeCount].right = -1;
    return nodeCount++;
}
void kthSmallestUtil(int index, int k, int *count, int *result) {
    if (index == -1 || *count >= k) return;
    kthSmallestUtil(tree[index].left, k, count, result);
    (*count)++;
    if (*count == k) {
        *result = tree[index].data;
        return;
    }
    kthSmallestUtil(tree[index].right, k, count, result);
}
```



```c
int kthSmallest(int rootIndex, int k) {
    int count = 0, result = -1;
    kthSmallestUtil(rootIndex, k, &count, &result);
    return result;
}
int main() {
    int root = createNode(20);
    int n1 = createNode(8);
    int n2 = createNode(22);
    int n3 = createNode(4);
    int n4 = createNode(12);
    int n5 = createNode(10);
    int n6 = createNode(14);
    tree[root].left = n1;
    tree[root].right = n2;
    tree[n1].left = n3;
    tree[n1].right = n4;
    tree[n4].left = n5;
    tree[n4].right = n6;
    printf("K = 3, Kth Smallest = %d\n", kthSmallest(root, 3));
    printf("K = 5, Kth Smallest = %d\n", kthSmallest(root, 5));
    return 0;
}
```



```
K = 3, Kth Smallest = 10
K = 5, Kth Smallest = 14
```