include<stdio.h>

#include<stdlib.h>

struct node

{

int key;

struct node \*left, \*right;

};

struct node \*newNode(int item)

{

struct node \*temp = (struct node \*)malloc(sizeof(struct node));

temp->key = item;

temp->left = temp->right = NULL;

return temp;

}

void traversetree(struct node \*root)

{

if (root != NULL)

{

traversetree(root->left);

printf("%d \t", root->key);

traversetree(root->right);

}

}

struct node\* search(struct node\* root, int key)

{

if (root == NULL || root->key == key)

return root;

if (root->key < key)

return search(root->right, key);

return search(root->left, key);

}

struct node\* insert(struct node\* node, int key)

{

if (node == NULL) return newNode(key);

if (key < node->key)

node->left = insert(node->left, key);

else if (key > node->key)

node->right = insert(node->right, key);

return node;

}

int main(){

struct node \*root = NULL;

root = insert(root, 23);

insert(root, 15);

insert(root, 12);

insert(root, 17);

insert(root, 32);

insert(root, 29);

insert(root, 45);

printf("The tree is :\n");

traversetree(root);

printf("\nSearching for 12 in this tree ");

if(search(root , 12))

printf("\nelement found");

else

printf("\nelement not found");

return 0;

}