#include <stdio.h>

#include <stdlib.h>

struct node {

int data;

struct node\* left;

struct node\* right;

};

struct node\* createnode(int data) {

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

newnode->data = data;

newnode->left = NULL;

newnode->right = NULL;

return newnode;

}

struct node\* insert(struct node\* root, int data) {

if (root == NULL) {

return createnode(data);

}

if (data < root->data) {

root->left = insert(root->left, data);

} else if (data > root->data) {

root->right = insert(root->right, data);

}

return root;

}

void inorderTraversal(struct node\* root) {

if (root != NULL) {

inorderTraversal(root->left);

printf("%d ", root->data);

inorderTraversal(root->right);

}

}

int main() {

struct node\* root = NULL;

int n, i, value;

printf("Enter the number of elements to insert in the BST: ");

scanf("%d", &n);

for (i = 0; i < n; i++) {

printf("Enter value %d: ", i + 1);

scanf("%d", &value);

root = insert(root, value);

}

printf("In-order traversal of the BST: ");

inorderTraversal(root);

printf("\n");

return 0;

}

