Program 10 (BST)

```
typedef struct binary_node {
        int data;
        struct binary node* left;
        struct binary node* right;
} node;

void insert(node** root, int d){
        if (*root == NULL) {
            (*root) = (node*) malloc(sizeof(node));
            (*root) -> left = NULL;
            (*root) -> data = d;
            (*root) -> right = NULL;
        }
        else {
            if (d < (*root) -> data) {
                insert(&(*root) -> left, d);
            }
            else
                insert(&(*root) -> right, d);
        }
}

void inorder(node *root) {
        if (root == NULL){
            return;
        }
        else {
```

```
printf(
inorder(root->left);
printf("%d", root->data);
inorder(root->right);
}

void preorder(node *root){
    if(root == NULL){
        return;
    }
    printf("%d", root->data);
    preorder(root->left);
    preorder(root->right);

void postorder(node *root){
    if(root == NULL){
        return;
    }
    printf(
    postorder(root->left);
    postorder(root->right);
    printf("%d", root->data);
}

bool search(node *root, int key){
    if(root == NULL){
        return false;
    }
```

```
if (root -> data == key) {
    return true;
}
else if (key < root -> data) {
    return search(root -> left, key);
}
else {
    return search(root -> right, key);
}
}
```