

017 Sec

8. Write a Program

a) To construct binary search tree

b) To traverse the tree using all the methods
Preorder, Postorder, Inorder

c) To display the elements in the tree

```
#include <stdio.h>
```

```
#include <stdio.h>
```

```
struct Node {
```

```
    int data;
```

```
    struct Node *left, *right;
```

```
};
```

```
struct Node* createNode(int data) {
```

```
    struct Node* newNode = (struct Node*) malloc  
    (sizeof(struct Node));
```

```
    newNode->data = data;
```

```
    newNode->left = 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;
```

```
}
```



Date _____
Page _____

```

void inorder (struct Node* root) {
    if (root != NULL) {
        inorder (root->left);
        printf ("%d", root->data);
        inorder (root->right);
    }
}

```

```

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

```

```

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

```

```

void displayTree (struct Node* root) {
    printf ("Inorder Traversal:");
    inorder (root);
    printf ("\n");

    printf ("Preorder Traversal:");
    preorder (root);
    printf ("\n");

    printf ("Postorder Traversal:");
    postorder (root);
    printf ("\n");
}

```



```

}

```

```

int main() {

```

```

    struct Node* root = NULL;

```

```

    int n, value;

```

```

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

```

```

    scanf("%d", &n);

```

```

    printf("Enter the elements:");

```

```

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

```

```

        scanf("%d", &value);

```

```

        root = insert(root, value);

```

```

    }

```

```

    printf("Tree Traversal:");

```

```

    displayTree(root);

```

```

    return 0;

```

```

}

```

O/P

Binary Search Tree Operations

1. ~~Insert~~ Enter the number of elements to insert in the tree:

Enter the elements: 12, 13, 14, 8, 9, 10,

Inorder Traversal: 8, 9, 10, 12, 13, 14

Pre-order Traversal: 12, 8, 9, 10, 13, 14

Post-order Traversal: 10, 9, 8, 12, 14, 13