ASSIGNMENT-11

1. Write a code to implement In-Order, Pre-Order and Post-Order Traversal in a Binary Search Tree

SOLUTION:

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

struct Node {
    int data;
    struct Node* left;
    struct Node* right;
};

struct Node* createNode(int value) {
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->data = value;
    newNode->left = newNode->right = NULL;
    return newNode;
}

struct Node* insert(struct Node* root, int value) {
    if (root == NULL) {
        return createNode(value);
    }
    if (value < root->data) {
        root->left = insert(root->left, value);
    }
    else if (value > root->data) {
        root->right = insert(root->right, value);
    }
    return root;
}

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

void preOrderTraversal(struct Node* root) {
    if (root != NULL) {
```

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

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

void freeTree(struct Node* root) {
    if (root != NULL) {
        freeTree(root->left);
        freeTree(root->right);
        free(root);
    }
}

int main() {
    struct Node* root = NULL;
    int n, value;
    printf("Enter the number of elements in the BST: ");
    scanf("%d", &n);
    printf("Enter the elements separated by space: ");
    for (int i = 0; i < n; i++) {
        scanf("%d", &value);
        root = insert(root, value);
    }
    printf("\nIn-order traversal: ");
    inOrderTraversal(root);
    printf("\nPre-order traversal: ");
    preOrderTraversal(root);
    printf("\nPost-order traversal: ");
    postOrderTraversal(root);
    freeTree(root);
    return 0;
}
```

OUTPUT:

```
Enter the number of elements in the BST: 8
Enter the elements separated by space: 5 9 8 3 6 11 15 10

In-order traversal: 3 5 6 8 9 10 11 15
Pre-order traversal: 5 3 9 8 6 11 10 15
Post-order traversal: 3 6 8 10 15 11 9 5
```