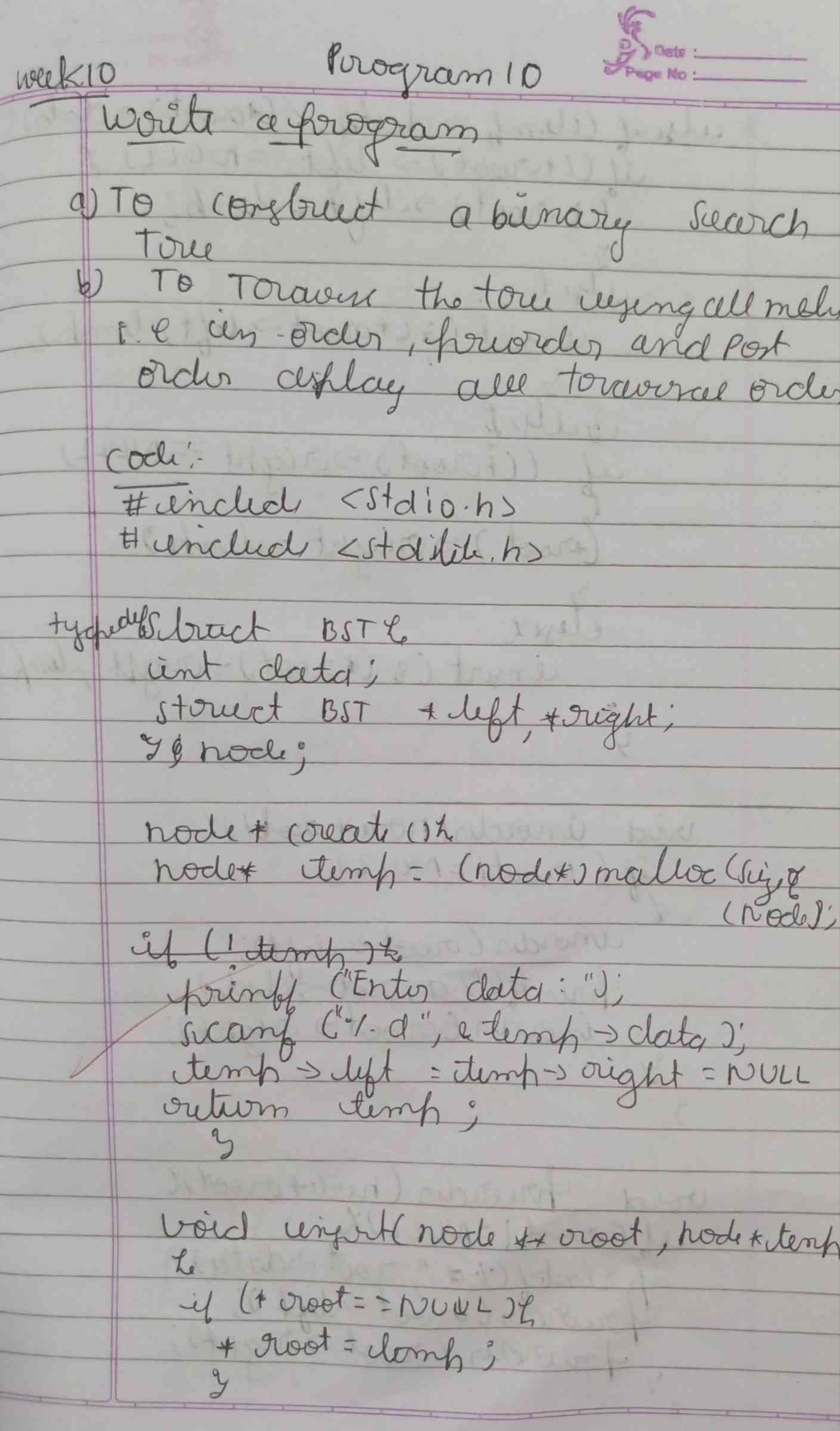
**LABORATORY PROGRAM – 10**

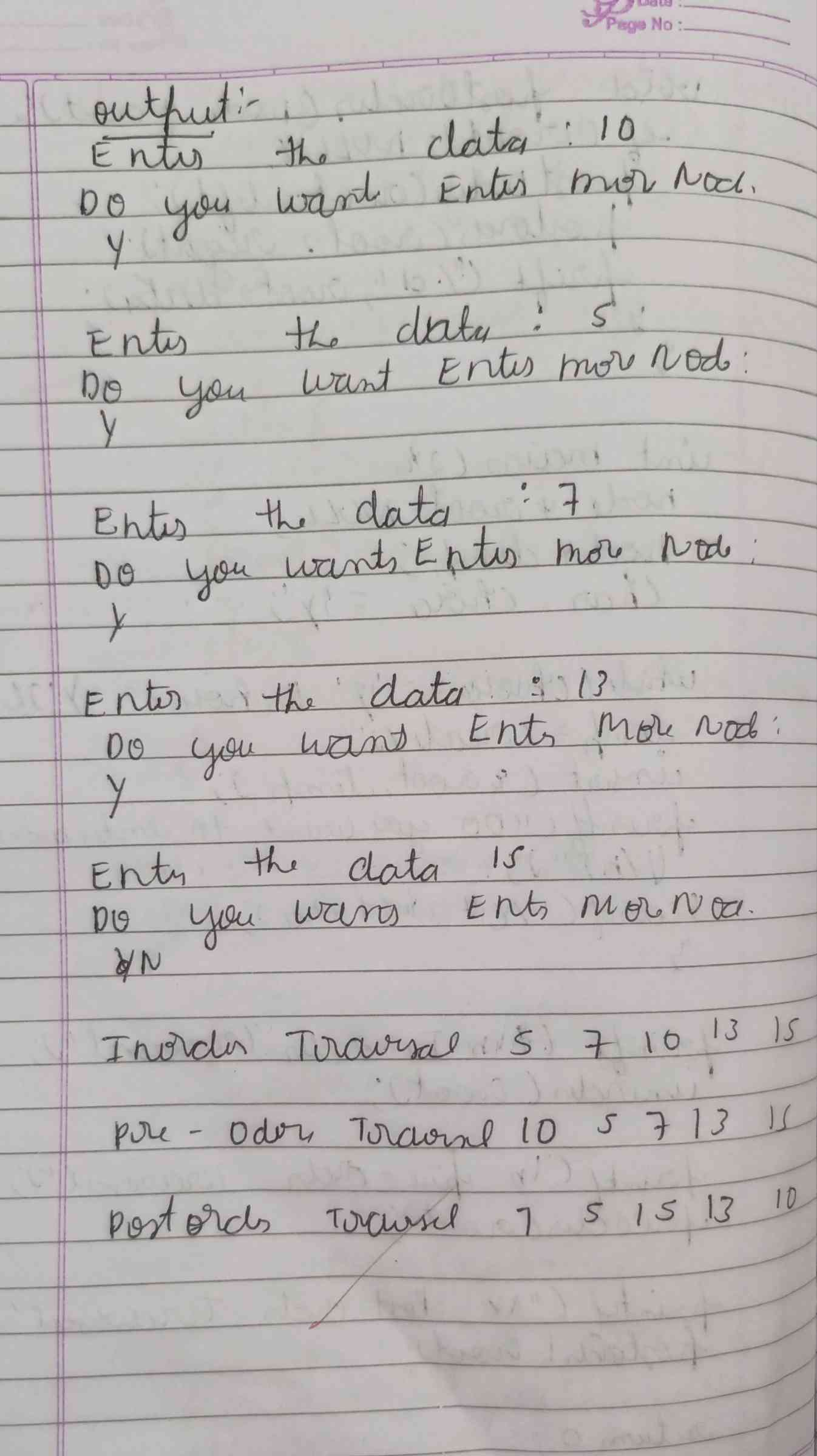
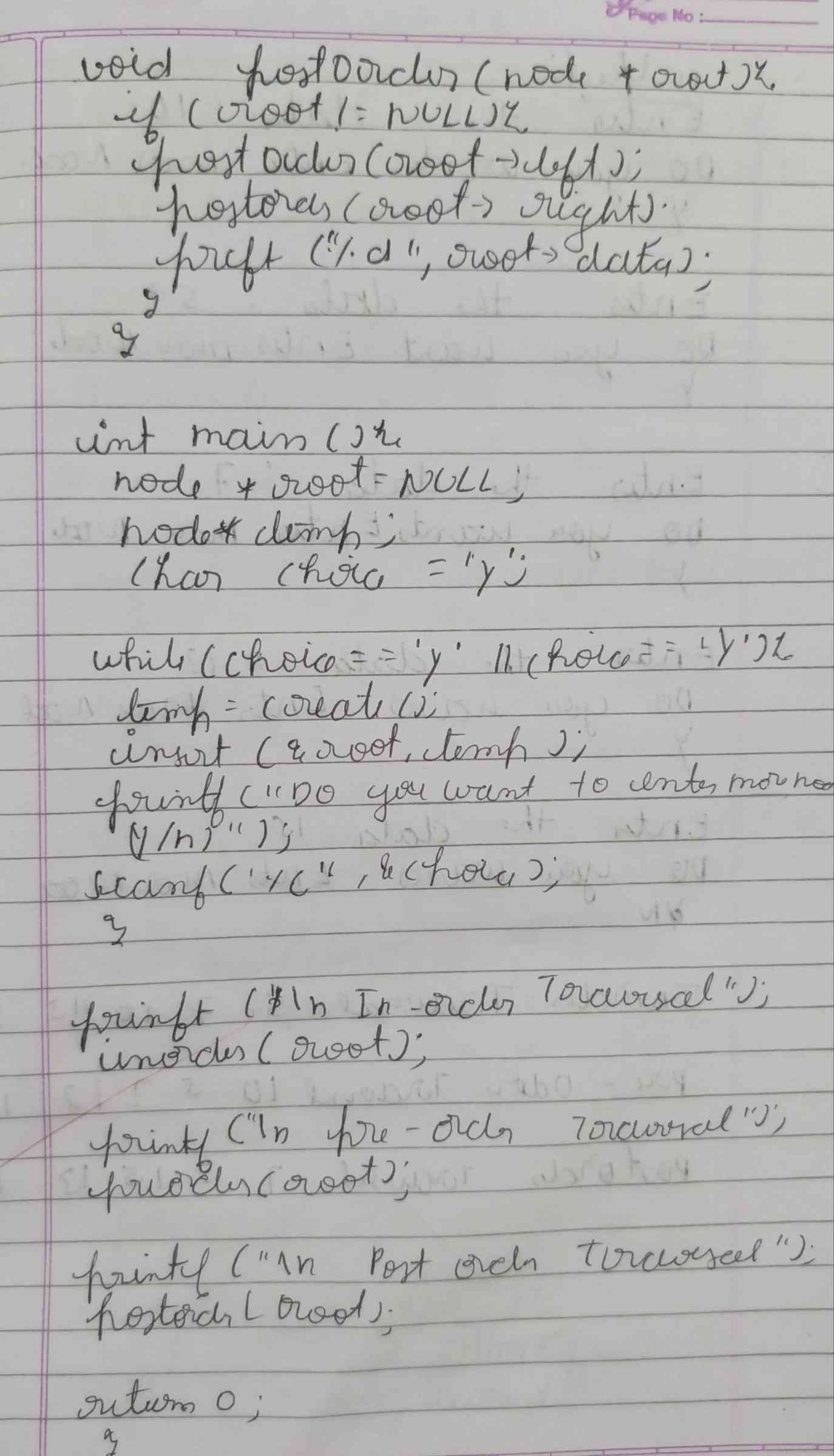
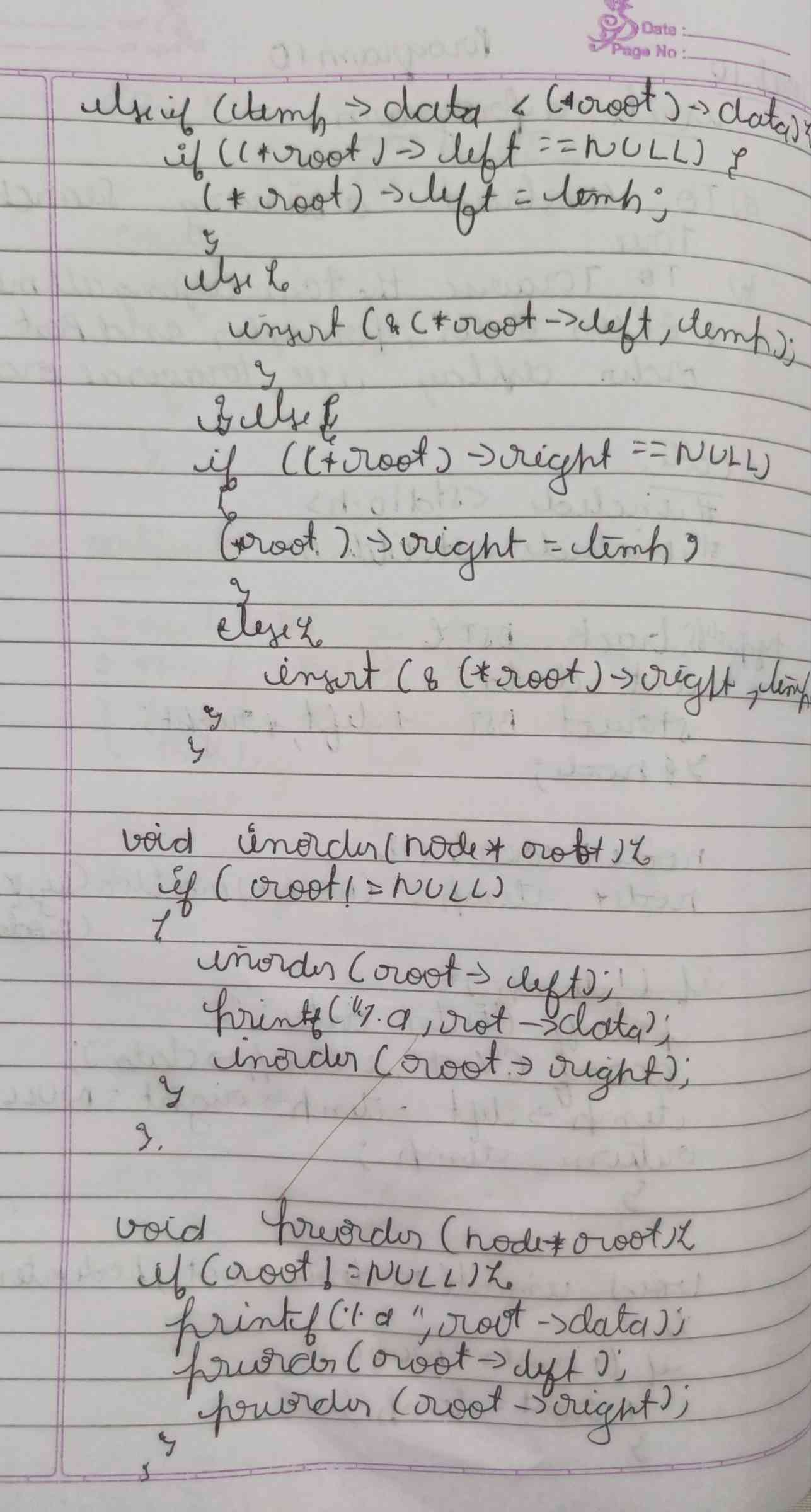
Write a program a) To construct a binary Search tree.

b) To traverse the tree using all the methods i.e., in-order, preorder and post order

c) To display the elements in the tree

**OBSERVATION :**

****

****

**CODE :**

**#include <stdio.h>**

**#include <stdlib.h>**

**typedef struct BST**

**{**

**int data;**

**struct BST \*left, \*right;**

**} node;**

**node \*createNode()**

**{**

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

**printf("Enter the data : ");**

**scanf("%d", &temp->data);**

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

**return temp;**

**}**

**void insert(node \*\*root, node \*temp)**

**{**

**if (\*root == NULL)**

**{**

**\*root = temp;**

**}**

**else if (temp->data < (\*root)->data)**

**{**

**if ((\*root)->left == NULL)**

**{**

**(\*root)->left = temp;**

**}**

**else**

**{**

**insert(&(\*root)->left, temp);**

**}**

**}**

**else**

**{**

**if ((\*root)->right == NULL)**

**{**

**(\*root)->right = temp;**

**}**

**else**

**{**

**insert(&(\*root)->right, temp);**

**}**

**}**

**}**

**void inorder(node \*root)**

**{**

**if (root != NULL)**

**{**

**inorder(root->left);**

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

**inorder(root->right);**

**}**

**}**

**void preorder(node \*root)**

**{**

**if (root != NULL)**

**{**

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

**preorder(root->left);**

**preorder(root->right);**

**}**

**}**

**void postOrder(node \*root)**

**{**

**if (root != NULL)**

**{**

**postOrder(root->left);**

**postOrder(root->right);**

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

**}**

**}**

**int main()**

**{**

**node \*root = NULL;**

**node \*temp;**

**char choice = 'y';**

**while (choice == 'y' || choice == 'Y')**

**{**

**temp = createNode();**

**insert(&root, temp);**

**printf("Do you want to Enter More node : \n");**

**scanf(" %c", &choice);**

**}**

**printf("\n Inorder Traversal");**

**inorder(root);**

**printf("\n Pre-Order Traversal");**

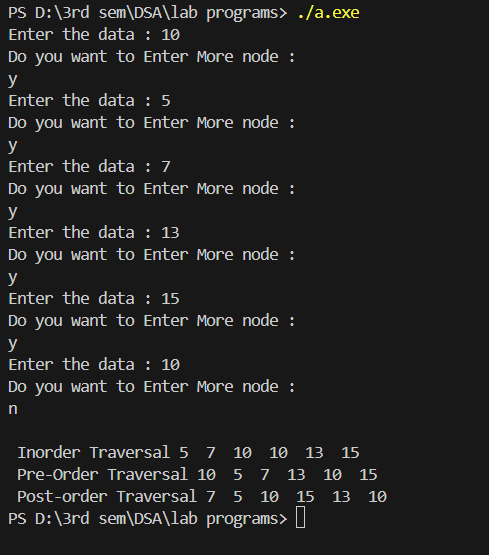
**preorder(root);**

**printf("\n Post-order Traversal");**

**postOrder(root);**

**}**

**OUTPUT :**

****