#include <stdio.h>

#include <stdlib.h>

struct Node {

int value;

struct Node\* left;

struct Node\* right;

};

struct Node\* newNode(int value) {

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

node->value = value;

node->left = NULL;

node->right = NULL;

return node;

}

struct Node\* insert(struct Node\* root, int value) {

if (root == NULL) {

return newNode(value);

}

if (value < root->value) {

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

} else {

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

}

return root;

}

void inorder(struct Node\* root) {

if (root != NULL) {

inorder(root->left);

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

inorder(root->right);

}

}

void preorder(struct Node\* root) {

if (root != NULL) {

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

preorder(root->left);

preorder(root->right);

}

}

void postorder(struct Node\* root) {

if (root != NULL) {

postorder(root->left);

postorder(root->right);

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

}

}

int main() {

int choice, value;

struct Node\* root = NULL;

while (1) {

printf("\n1. Insert value\n2. Inorder traversal\n3. Preorder traversal\n4. Postorder traversal\n5. Exit\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch (choice) {

case 1:

printf("Enter value to insert: ");

scanf("%d", &value);

root = insert(root, value);

break;

case 2:

printf("Inorder: ");

inorder(root);

printf("\n");

break;

case 3:

printf("Preorder: ");

preorder(root);

printf("\n");

break;

case 4:

printf("Postorder: ");

postorder(root);

printf("\n");

break;

case 5:

exit(0);

default:

printf("Invalid choice!\n");

}

}

return 0;

}

1. Insert value

2. Inorder traversal

3. Preorder traversal

4. Postorder traversal

5. Exit

Enter your choice: 1

Enter value to insert: 45

1. Insert value

2. Inorder traversal

3. Preorder traversal

4. Postorder traversal

5. Exit

Enter your choice: 1

Enter value to insert: 79

1. Insert value

2. Inorder traversal

3. Preorder traversal

4. Postorder traversal

5. Exit

Enter your choice: 1

Enter value to insert: 65

1. Insert value

2. Inorder traversal

3. Preorder traversal

4. Postorder traversal

5. Exit

Enter your choice: 2

Inorder: 45 65 79

1. Insert value

2. Inorder traversal

3. Preorder traversal

4. Postorder traversal

5. Exit

Enter your choice: 3

Preorder: 45 79 65

1. Insert value

2. Inorder traversal

3. Preorder traversal

4. Postorder traversal

5. Exit

Enter your choice: 4

Postorder: 65 79 45

1. Insert value

2. Inorder traversal

3. Preorder traversal

4. Postorder traversal

5. Exit

Enter your choice: 5