#include <stdio.h>

#include <stdlib.h>

typedef struct node

{

    int data;

    struct node \*left;

    struct node \*right;

} Node;

Node \*newNode(int data)

{

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

    node->data = data;

    node->left = NULL;

    node->right = NULL;

    return (node);

}

void printInorder(Node \*node)

{

    if (node == NULL)

        return;

    printInorder(node->left);

    printf("%d ", node->data);

    printInorder(node->right);

}

void printPreorder(Node \*node)

{

    if (node == NULL)

        return;

    printf("%d ", node->data);

    printPreorder(node->left);

    printPreorder(node->right);

}

void printPostorder(Node \*node)

{

    if (node == NULL)

        return;

    printPostorder(node->left);

    printPostorder(node->right);

    printf("%d ", node->data);

}

int main()

{

    Node \*root = newNode(1);

    root->left = newNode(2);

    root->right = newNode(3);

    root->left->left = newNode(4);

    root->left->right = newNode(5);

    printf("\nPreorder traversal of binary tree is \n");

    printPreorder(root);

    printf("\nInorder traversal of binary tree is \n");

    printInorder(root);

    printf("\nPostorder traversal of binary tree is \n");

    printPostorder(root);

    return 0;

}

