

```

#include <stdio.h>
int max_node = 15;
char tree[] = {'\0', 'D', 'A', 'F', 'E', 'B', 'R', 'T', 'G', 'Q', '\0', '\0',
               'V', '\0', 'J', 'L'};
int get_right_child(int index)
{
    if (tree[index] != '\0' && ((2 * index) + 1) <= max_node)
        return (2 * index) + 1;
    return -1;
}

int get_left_child(int index)
{
    if (tree[index] != '\0' && (2 * index) <= max_node)
        return 2 * index;
    return -1;
}

void preorder(int index)
{
    if (index > 0 && tree[index] != '\0')
    {
        printf(" %c ", tree[index]);
        preorder(get_left_child(index));
        preorder(get_right_child(index));
    }
}

void postorder(int index)
{
    if (index > 0 && tree[index] != '\0')
    {
        postorder(get_left_child(index));
        postorder(get_right_child(index));
        printf(" %c ", tree[index]);
    }
}

void inorder(int index)
{
    if (index > 0 && tree[index] != '\0')
    {
        inorder(get_left_child(index));
        printf(" %c ", tree[index]);
        inorder(get_right_child(index));
    }
}

int main()
{
    printf("Preorder:\n");
    preorder(1);
    printf("\nPostorder:\n");
    postorder(1);
    printf("\nInorder:\n");
    inorder(1);
    printf("\n");
    return 0;
}

```