

11 Yashraj Deepak Devrat

```
#include <iostream>
#include <string.h>
using namespace std;

struct node
{
    char data;
    node *left;
    node *right;
};

class tree
{
    char prefix[20];

public:
    node *top;
    void expression(char[]);
    void display(node *);
    void non_rec_postorder(node *);
    void del(node *);
};

class stack1
{
```

```
node *data[30];  
int top;
```

```
public:  
    stack1()  
    {  
        top = -1;  
    }  
    int empty()  
    {  
        if (top == -1)  
            return 1;  
        return 0;  
    }  
    void push(node *p)  
    {  
        data[++top] = p;  
    }  
    node *pop()  
    {  
        return (data[top--]);  
    }  
};  
void tree::expression(char prefix[])  
{  
    char c;  
    stack1 s;  
    node *t1, *t2;  
    int len, i;
```

```

len = strlen(prefix);
for (i = len - 1; i >= 0; i--)
{
    top = new node;
    top->left = NULL;
    top->right = NULL;
    if (isalpha(prefix[i]))
    {
        top->data = prefix[i];
        s.push(top);
    }
    else if (prefix[i] == '+' || prefix[i] == '*' || prefix[i] == '-' ||
prefix[i] == '/')
    {
        t2 = s.pop();
        t1 = s.pop();
        top->data = prefix[i];
        top->left = t2;
        top->right = t1;
        s.push(top);
    }
}
top = s.pop();
}

void tree::display(node *root)
{
    if (root != NULL)
    {
        cout << root->data;
        display(root->left);
    }
}

```

```

        display(root->right);
    }
}
void tree::non_rec_postorder(node *top)
{
    stack1 s1, s2; /*stack s1 is being used for flag . A NULL
data implies that the right subtree has not been visited */
    node *T = top;
    cout << "\n";
    s1.push(T);
    while (!s1.empty())
    {
        T = s1.pop();
        s2.push(T);
        if (T->left != NULL)
            s1.push(T->left);
        if (T->right != NULL)
            s1.push(T->right);
    }
    while (!s2.empty())
    {
        top = s2.pop();
        cout << top->data;
    }
}
void tree::del(node *node)
{
    if (node == NULL)
        return;
    /* first delete both subtrees */

```

```

    del(node->left);
    del(node->right);
    /* then delete the node */
    cout <<endl<<"Deleting node : " << node->data<<endl;
    free(node);
}
int main()
{
    char expr[20];
    tree t;

    cout <<"Enter prefix Expression : ";
    cin >> expr;
    cout << expr;
    t.expression(expr);
    //t.display(t.top);
    //cout<<endl;
    t.non_rec_postorder(t.top);
    t.del(t.top);
    // t.display(t.top);
}

```

Google

SE(AI&DS) DSA Laboratory

Online C++ Compiler

programiz.com/cpp-programming/online-compiler/

Programiz

C++ Online Compiler

betway

Valid from 00:00AM IST on 26/03/2022 until 11:59PM IST on 05/04/2022. For selected betway customers only. Promotion is only valid 3-10pm IST every day during the promotion period. Opt in required. Min deposit ₹20 & Min Play through ₹20 on selected Live Casino table to qualify. 50x bonus amount needs to be played through. Customer can claim this reward maximum 5 times per day. Other conditions apply.

18+ T&C's apply

mga

MGA/20C/130/2006

+18

play the responsible way

PLAY

Learn Python App

main.cpp

Run

Clear

```
1 //36_Vedant Kulkarni
2
3
4 #include <iostream>
5 #include <string.h>
6 using namespace std;
7
8
9 struct node
10 {
11     char data;
12     node *left;
13     node *right;
14 };
15 class tree
16 {
17     char prefix[20];
18
19
20 public:
21     node *top;
```

Output

Clear

```
/tmp/r4CT5DEC6K.o
Enter prefix Expression : ++-a*bc/def
+--a*bc/def
abc*-de/-f+
Deleting node : a
Deleting node : b
Deleting node : c
Deleting node : *
Deleting node : -
Deleting node : d
Deleting node : e
Deleting node : /
Deleting node : -
```

Windows Taskbar

22:59 21-05-2022