#include<iostream>

#include<string>

#include<stack>

using namespace std;

//Function to return precedence of operators

int prec(char c) {

    if(c == '^')

        return 3;

    else if(c == '/' || c=='\*')

        return 2;

    else if(c == '+' || c == '-')

        return 1;

    else

        return -1;

}

void infixToPostfix(string s) {

    stack<char> st;

    string result;

    for(int i = 0; i < s.length(); i++) {

        char c = s[i];

        if((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z') || (c >= '0' && c <= '9'))

            result += c;

        else if(+c == '(')

            st.push('(');

        else if(c == ')') {

            while(st.top() != '(')

            {

                result += st.top();

                st.pop();

            }

            st.pop();

        }

        else {

            while(!st.empty() && prec(s[i]) <= prec(st.top())) {

                result += st.top();

                st.pop();

            }

            st.push(c);

        }

    }

    while(!st.empty()) {

        result += st.top();

        st.pop();

    }

    cout << result << endl;

}

int main() {

    string exp = "a-b\*c-d/e+f";

    cout<<"Actual Inorder expression is:  a-b\*c-d/e+f"<<endl;

    cout<<"Post order expression is:";

    infixToPostfix(exp);

    return 0;

}