ASSIGNMENT 5

Part 2- : Syntax Parsing in Programming Languages

PROGRAM :-

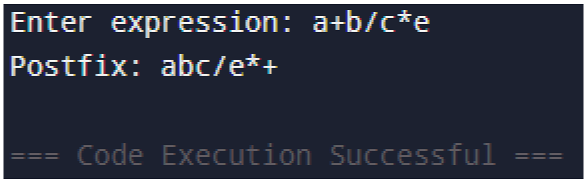
```
#include <iostream>
#include <stack>
using namespace std;

int prec(char c) {
    if (c == '^') return 3;
    if (c == '*' || c == '/') return 2;
    if (c == '+' || c == '-') return 1;
    return -1;
}

string infixToPostfix(string s) {
    stack<char> st;
    string res = "";
    for (char c : s) {
        if (isalnum(c)) res += c;
        else if (c == '(') st.push(c);
        else if (c == ')') {
            while (!st.empty() && st.top() != '(') {
                res += st.top(); st.pop();
            }
            if (!st.empty()) st.pop();
        } else {
            while (!st.empty() && prec(st.top()) >= prec(c) && c != '^') {
                res += st.top(); st.pop();
            }
            st.push(c);
        }
    }
    while (!st.empty()) { res += st.top(); st.pop(); }
    return res;
}

int main() {
    string exp;
    cout << "Enter expression: ";
    cin >> exp;
    cout << "Postfix: " << infixToPostfix(exp);
    return 0;
}
```

OUTPUT :-



```
Enter expression: a+b/c*e
Postfix: abc/e*+


=== Code Execution Successful ===
```