

#include <iostream>

#include <stack>

#include <string>

using namespace std;

bool isOperator(char c) {

    return c == '+' || c == '-' || c == '\*' || c == '/' || c == '^';

}

bool isOperand(char c) {

    return isdigit(c);

}

int precedence(char op) {

    if (op == '+' || op == '-') return 1;

    if (op == '\*' || op == '/') return 2;

    if (op == '^') return 3;

    return 0;

}

string infixToPostfix(string expression) {

    string postfix = "";

    stack<char> s;

    for (char c : expression) {

        if (c == ' ') continue;

        if (isOperand(c)) {

            postfix += c;

        } else if (c == '(') {

            s.push(c);

        } else if (c == ')') {

            while (!s.empty() && s.top() != '(') {

                postfix += s.top();

                s.pop();

            }

            s.pop(); // Remove '('

        } else if (isOperator(c)) {

            while (!s.empty() && precedence(s.top()) >= precedence(c)) {

                postfix += s.top();

                s.pop();

            }

            s.push(c);

        }

    }

    while (!s.empty()) {

        postfix += s.top();

        s.pop();

    }

    return postfix;

}

int main() {

    string expression = "sqrt(16)+2)/(sqrt(4)+2";

    string postfix = infixToPostfix(expression);

    cout << "Postfix expression: " << postfix << endl;

    return 0;

}

