

3. Виконати вручну трансляцію арифметичного виразу у постфіксну нотація $(\sqrt{16}+2)/(\sqrt{4}+2)$.

Написати програму трансляції арифметичних виразів, які містять числа з фіксованою крапкою і операції +, -, *, /, ^, (,), f() у постфіксну нотацію на мові C/C++.

```
#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;
}

```

← ↻ 🔍 <https://cpp.sh> 🔍 ⭐ 🔄 📄 ⚙️ 🕒 🌐 ⋮

📁 Университет 📁 Інструменти 📁 Лекція 📁 Gmail 📁 XD 📁 Mod Maker 📁 Internet Meme Dat... 📁 Space Station 14 C... 📁 Проект - Presentati... 📁 Ukraine Higher Edu... 📁 Інші вкладки

C++ shell

online C++ compiler
about cpp.sh

```

31-     } else if (c == ')') {
32-         while (!s.empty() && s.top() != '(') {
33-             postfix += s.top();
34-             s.pop();
35-         }
36-         s.pop(); // Remove '('
37-     } else if (isOperator(c)) {
38-         while (!s.empty() && precedence(s.top()) >= precedence(c)) {
39-             postfix += s.top();
40-             s.pop();
41-         }
42-         s.push(c);
43-     }
44- }
45- while (!s.empty()) {
46-     postfix += s.top();
47-     s.pop();
48- }
49- return postfix;
50- }
51-
52- int main() {
53-     string expression = "sqrt(16)+2)/(sqrt(4)+2";
54-     string postfix = infixToPostfix(expression);
55-     cout << "Postfix expression: " << postfix << endl;
56-     return 0;
57- }
58-

```

Link to this code: [🔗](#) [\[copy\]](#) Run

[options](#) [compilation](#) [execution](#)

Postfix expression: 162+42+(

Normal program termination. Exit status: 0

C++ Shell 2.0 © cpp.sh 2014-2024 | [buy me a coffee](#)
old version still available [here](#) (for a limited time).