Колтунов Кирилл Константинович Выйти

ВШЭ АиСД 2021. STL!

12 янв 2022, 18:06:28
старт: 7 дек 2021, 20:00:00
финиш: 18 дек 2021, 03:59:00
длительность: 10д. 7ч.
начало: 7 дек 2021, 20:00:00
конец: 18 дек 2021, 03:59:00

A. Постфиксная нотация (0.15)

Ограничение времени	0.1 секунд
Ограничение памяти	64.0 Мб
Ввод	стандартный ввод
Вывод	стандартный вывод

Дано выражение, содержащее некоторые числа, круглые скобки, знаки сложения, вычитания, умножения и деления и записанное в инфиксной записи. Напишите программу, переводящую это выражение в постфиксную нотацию (обратную польскую запись).

Формат ввода

На вход программе подается некоторое выражение, записанное в инфиксной записи, где каждый оператор или операнд отделен пробелом от предыдущего.

Формат вывода

Программа должна вывести одну строку – результат перевода выражения в постфиксную нотацию.

Пример 1

Ввод	Вывод
`7 * 8 + ( 3 - 5 )`	`7 8 * 3 5 - +`

Пример 2

Ввод	Вывод
`1 + 2 / 5 + 10 - 8 + 12 - 17 + 45 - 123 * 2 + 15 - 94 / 7`	`1 2 5 / + 10 + 8 - 12 + 17 - 45 + 123 2 * - 15 + 94 7 / -`

Примечания

Система оценки

Группа	Баллы	Доп. ограничения	Необх. группы	Комментарий
		кол-во знаков n		
0	1	–	–	Тест из условия.
1	1	$n \leq 10$	0	
2	4	$n \leq 60$	0 – 1	
3	2	$n \leq 1000$	0 – 2	

Группа	Баллы	Доп. ограничения	Необх. группы	Комментарий
4	2	$n \leq 12000$	0 – 3	**Offline-проверка**

Набрать здесь | Отправить файл

```cpp
#include <iostream>
#include <stack>
#include <string>
#include <vector>
#include <sstream>

int priorityOperation(const std::string &check_string) {
    if (check_string == "*" || check_string == "/") {
        return 2;
    }
    if (check_string == "+" || check_string == "-") {
        return 1;
    } else {
        return 0;
    }
}

void findPostfixNote(std::vector<std::string> *infix_sequence_vector,
                     std::vector<std::string> *postfix_sequence) {
    std::stack<std::string> stack;
    for (size_t index = 0; index < infix_sequence_vector->size(); ++index) {
        if (infix_sequence_vector->operator[](index)[0] >= '0' &&
            infix_sequence_vector->operator[](index)[0] <= '9') {
            postfix_sequence->push_back(infix_sequence_vector->operator[](index));
        } else if (infix_sequence_vector->operator[](index)[0] == '(') {
            stack.push(infix_sequence_vector->operator[](index));
        } else if (infix_sequence_vector->operator[](index)[0] == ')') {
            while (!stack.empty() && stack.top() != "(") {
                postfix_sequence->push_back(stack.top());
                stack.pop();
            }
            stack.pop();
        } else if (infix_sequence_vector->operator[](index)[0] == '+' ||
                   infix_sequence_vector->operator[](index)[0] == '*' ||
                   infix_sequence_vector->operator[](index)[0] == '-' ||
                   infix_sequence_vector->operator[](index)[0] == '/') {
            while (!stack.empty() &&
                   priorityOperation(stack.top()) >=
```

Отправить

Следующая

© 2013–2022 ООО «Яндекс»