

Documentación del Código: Resolución de Expresiones Matemáticas en Notación Postfija

Este documento proporciona una explicación detallada del código proporcionado para resolver expresiones matemáticas en notación postfija utilizando C++. A continuación, se describirán los componentes principales del código, su funcionamiento y su estructura general.

Introducción

El código tiene como objetivo principal recibir una expresión matemática en notación infija y convertirla en notación postfija para su posterior evaluación. La notación postfija, también conocida como notación polaca inversa, es una forma de notación matemática en la que los operadores siguen a los operandos. Este código consta de las siguientes partes principales:

1. Lectura de la expresión matemática en notación infija.
2. Conversión de la expresión infija a postfija.
3. Evaluación de la expresión postfija.

Componentes Principales del Código

A continuación, se describen los componentes principales del código y su funcionalidad:

1. Lectura de la Expresión

El código comienza solicitando al usuario que ingrese una expresión matemática en notación infija. La expresión ingresada se almacena en un arreglo de caracteres llamado ``exp``.

2. Conversión de la Expresión Infija a Postfija

La conversión de la expresión infija a postfija se realiza mediante el uso de una pila (``Pila pc1``) y un arreglo de caracteres llamado ``postfija``. El código recorre cada carácter de la expresión infija y toma las siguientes acciones:

- Si se encuentra un carácter 'x', se solicita al usuario que ingrese un valor para 'x' hasta que se ingrese un dígito válido.
- Si se encuentra un paréntesis izquierdo '(', se coloca en la pila.
- Si se encuentra un número, se agrega al arreglo `postfija`. Nota: se ha comentado un bloque de código alternativo que parece tener un error y que se supone que debería leer números completos.
- Si se encuentra un operador (+, -, *, /), se compara su prioridad con los operadores en la pila. Se siguen sacando operadores de la pila y se agregan a `postfija` hasta que la prioridad en la pila sea menor o igual a la del operador actual. Luego, el operador actual se coloca en la pila.
- Si se encuentra un paréntesis derecho ')', se sacan operadores de la pila y se agregan a `postfija` hasta que se encuentra un paréntesis izquierdo '('.

Finalmente, se verifican los elementos restantes en la pila y se agregan a `postfija`.

3. Evaluación de la Expresión Postfija

La evaluación de la expresión postfija se realiza utilizando otra pila (`Pila2 pi1`). El código recorre cada carácter en la expresión postfija y realiza las siguientes acciones:

- Si se encuentra un número, se agrega a la pila `pi1`.
- Si se encuentra un operador (+, -, *, /), se sacan dos elementos de la pila `pi1`, se realiza la operación correspondiente y se coloca el resultado nuevamente en la pila.

Finalmente, el código muestra el resultado de la expresión postfija, que es el valor final de la evaluación, utilizando `pi1.datoTope()`.

Conclusiones

El código proporcionado es capaz de tomar una expresión matemática en notación infija, convertirla en notación postfija y evaluarla para obtener el resultado. Además, permite al usuario ingresar valores para la variable 'x' en la expresión