

Análisis de Expresiones Aritméticas

Introducción

Las pilas son estructuras de datos tipo LIFO (Last in, First Out), donde el último elemento en ingresar es el primero en salir.

Este comportamiento resulta muy útil en procesos informáticos, una de sus aplicaciones más importantes es el análisis de comportamientos de expresiones aritméticas, utilizando compiladores e intérpretes para evaluar operaciones matemáticas de forma ordenada y segura

Este estudio de caso tiene como objetivo demostrar como una pila permite evaluar expresiones aritméticas de manera eficiente, garantizando el orden de las operaciones y la correcta agrupación de símbolos matemáticos

¿Por qué una pila es adecuada para esta funcionalidad?

El análisis de expresiones requiere procesar operadores y operandos respetando las reglas de precedencia y agrupación, como paréntesis o jerarquías matemáticas

Una pila es ideal porque permite:

- Guardar temporalmente operadores y operandos se analiza la expresión
- Extraerlos en orden correcto para analizar cálculos cuando corresponde
- Controla el balanceo de paréntesis y detecta errores de sintaxis

Seguridad, orden y eficiencia

El uso de pila aporta seguridad y evitar operaciones fuera de orden, orden en la que ejecución correcta de los cálculos y eficiencia en el manejo automático de operaciones anidadas

Facilita:

- Validar operaciones correctamente balanceadas. como $((2 + 3) * 5)$
- Prevenir errores lógicos en el procedimiento de paréntesis o símbolos no válidos
- Optimiza el tiempo de ejecución, ya que las operaciones de inserción (push) y (pop) son de complejidad

Ejemplo de Software

En los compiladores de lenguajes como C, Java o Python, las pilas se usan para evaluar expresiones aritméticas y lógicas.

Cuando un compilador traduce código fuente, convierte las expresiones en notación postfija (también conocida como *Reverse Polish Notation*), que se evalúa con una pila.

Por ejemplo:

$$(3 + (5 * (2 - 1)))$$

1. Los números (3, 5, 2, 1) se apilan temporalmente.
2. Cuando aparece un operador (+, -, *), se desapilan los últimos dos números, se calcula el resultado, y se vuelve a apilar.
3. El resultado final queda en la cima de la pila.

Representación Gráfica

Operación	Acción	Estado de la pila
push(3)	Se apila el número 3	[3]
push(5)	Se apila el número 5	[3, 5]
push(2)	Se apila el número 2	[3, 5, 2]
push(1)	Se apila el número 1	[3, 5, 2, 1]
pop y calcular (2-1)	Resultado: 1	[3, 5, 1]
pop y calcular (5*1)	Resultado: 5	[3, 5]
pop y calcular (3*5)	Resultado: 8	[8]