



UNIVERSIDAD AUTONOMA DE TAMAULIPAS

Materia: Programación Base 2

Profesor: Muñoz Quintero Dante Adolfo

Manual de Usuario

Integrantes del Equipo:

- García Azua Jorge Roberto
- Cruz Bonifacio Luis Fernando
- Gonzalez Cavazos Erick Alan

ÍNDICE

1. Introducción	3
2. Requisitos y Ejecución.....	3
2.2. Comandos de Ejecución	3
3. Instrucciones de Uso (Interacción).....	3
4. Funcionalidades Soportadas	4
4.1 Jerarquía de Operaciones.....	4
4.2 Operadores y Constantes.....	5
4.3 Funciones Científicas Detalladas	5

MANUAL DE USO: CALCULADORA CIENTÍFICA

(ANALIZADOR ANTLR)

1. INTRODUCCIÓN

Generar un analizador que pueda reconocer operaciones matemáticas complejas, respetar la jerarquía de operaciones, y resolver las expresiones ingresadas.

2. REQUISITOS Y EJECUCIÓN

Para iniciar la aplicación, es indispensable tener el entorno de Java configurado y la librería ANTLR disponible en el classpath.

Requisitos de Software

- Java Runtime Environment (JRE) o JDK: Versión 8 o superior.
- Librería ANTLR 4: El archivo antlr-4.13.1-complete.jar debe encontrarse en el mismo directorio que el programa principal compilado (Calculadora.class).

Preparación:

- Asegúrese de que el archivo principal del programa (asumido como Calculadora.java o similar) y el runtime de ANTLR (antlr-4.13.1-complete.jar) se encuentren en el mismo directorio.
- Abra la terminal o línea de comandos en el directorio donde se encuentran los archivos.

2.2. Comandos de Ejecución

El programa debe ejecutarse desde la línea de comandos, especificando el classpath que incluye la librería ANTLR y la ubicación de las clases del proyecto.

Sistema Operativo	Comando de Ejecución
Linux/macOS	java -cp antlr-4.13.1-complete.jar:. Calculadora
Windows	java -cp "antlr-4.13.1-complete.jar;." Calculadora

3. INSTRUCCIONES DE USO (INTERACCIÓN)

La calculadora opera en un ciclo continuo de lectura, procesamiento y evaluación de expresiones.

Flujo de Interacción

1. **Inicio:** Al ejecutar el comando, la aplicación muestra el mensaje: Calculadora Científica (escribe 'salir' para terminar).
2. **Entrada:** El sistema solicita la expresión a evaluar con el *prompt* Ingrese expresión:
3. **Procesamiento:** El analizador ANTLR lee la expresión, verifica la sintaxis y, si es correcta, procede a la evaluación.

4. **Salida Estándar (Resultado Exitoso):** Se muestran dos elementos de salida esenciales:
 - o **Árbol Sintáctico (Parse Tree):** Muestra la estructura jerárquica que el analizador utilizó para entender la expresión.
 - o **Resultado:** El valor numérico final de la evaluación.
5. Árbol sintáctico: (prog (expr ...) <EOF>)
6. Resultado: [Valor numérico]
7. **Finalización:** El programa se detiene cuando el usuario ingresa el comando **salir** (no sensible a mayúsculas/minúsculas) en el *prompt* de entrada.

Manejo de Errores

En caso de un error de sintaxis (una expresión mal formada), el sistema detiene el proceso de análisis y reporta la ubicación y el tipo de error, gracias al manejador de errores configurado en Calculadora.java.

Formato de Error: Error: Error en línea [LÍNEA]:[COLUMNA] - [MENSAJE DE ERROR ANTLR]

4. FUNCIONALIDADES SOPORTADAS

El motor de la calculadora está definido por la gramática calc.g4 y su evaluación se implementa en EvalVisitor.java, lo que garantiza la correcta jerarquía y el cálculo preciso.

4.1 Jerarquía de Operaciones

El analizador respeta el orden estándar de precedencia (similar a BODMAS/PEMDAS) para garantizar resultados correctos:

1. **Agrupación** (Paréntesis) ()
2. **Funciones** (Llamadas a sin(x), log(x), etc.)
3. **Potenciación** ^
4. **Negación Unaria** - (Ej. -5)
5. **Multiplicación y División** *, /
6. **Suma y Resta** +, -

4.2 Operadores y Constantes

Tipo	Símbolo/Palabra	Descripción
Aritméticos	+ , - , * , / , ^	Suma, Resta, Multiplicación, División, Potencia.
Constantes	pi	Utiliza el valor de \$\\pi\$ (Math.PI en Java).
	e	Utiliza el valor de \$e\$ (Math.E en Java).
Números	Formato completo	Soporta enteros, decimales y notación científica (Ej. 123, 4.56, 1e-5).

4.3 Funciones Científicas Detalladas

Las siguientes funciones se implementan directamente utilizando los métodos correspondientes de la clase Math de Java (EvalVisitor.java):

Función	Uso	Descripción y Equivalencia en Java
Trigonométricas	sin(x), cos(x), tan(x)	Seno, Coseno, Tangente. Los argumentos deben estar en radianes.
	cot(x), sec(x), csc(x)	Cotangente, Secante, Cosecante (Calculadas a partir de sus inversas).
Logarítmicas	log(x)	Logaritmo base 10 (Math.log10).
	ln(x)	Logaritmo natural base e (Math.log).
	exp(x)	Función exponencial (\$e^x\$).
Misceláneas	sqrt(x)	Raíz cuadrada (Math.sqrt).
	abs(x)	Valor absoluto (Math.abs).
Redondeo	floor(x)	Entero más grande menor o igual a x (Math.floor).
	ceil(x)	Entero más pequeño mayor o igual a x (Math.ceil).
	round(x)	Redondeo al entero más cercano (Math.round convertido a Double).
	trunc(x)	Truncar la parte decimal (Implementado como ceil o floor).
Argumentos Múltiples	min(x, y, ...)	Valor mínimo de la lista de argumentos (Math.min aplicado secuencialmente).
	max(x, y, ...)	Valor máximo de la lista de argumentos (Math.max aplicado secuencialmente).

Nota sobre Argumentos Múltiples: Las funciones min y max aceptan dos o más expresiones separadas por comas. Por ejemplo: max(5, sin(pi), 10)