

Programación con Memoria Dinámica

Evaluador de expresiones matemáticas

I. Objetivo de la actividad

La evaluación extra-aúllica del parcial, en la materia de Programación con Memoria Dinámica (PMD), tiene por objetivo medir su capacidad de resolución de problemas computacionales, incluyendo algunos o todos los temas que se han abordado en el curso; además, se pretende evidenciar su aprendizaje en programación utilizando el lenguaje C, así como sus habilidades de investigación y capacidad del trabajo en equipo.

II. Descripción del problema

En esta actividad deberás crear un programa en C que, permita al usuario escribir una expresión matemática. El programa deberá validar si la expresión fue escrita de forma correcta, evaluarla e imprimir el resultado en pantalla. En caso de que exista un error en la expresión, su programa deberá mostrar un mensaje de error.

Previo a que inicies con el diseño de la solución y posterior implementación, conviene conocer tres de las principales notaciones que nos permiten representar y evaluar expresiones matemáticas.

2.1 Notación infija, prefija y postfija

Una expresión matemática simple se puede ver como una secuencia de operandos y operadores. Los operandos indican las operaciones a realizar sobre los operandos, por ejemplo: $2+3$. Para esta actividad, únicamente consideraremos el uso de los operadores: $+$, $-$, $/$, $*$, y operandos numéricos enteros (representados en el sistema decimal). Los operadores que utilizaremos son binarios ya necesitan dos operandos para evaluar la operación. Adicionalmente, se podrán utilizar paréntesis para la notación infija.

Una expresión matemática puede ser escrita mediante el uso de diversas notaciones, que se diferencian por el orden en que se escriben los operandos y los operadores. Las más relevantes son:

- **Notación infija:** El orden es primer operando, operador, segundo operando.
- **Notación prefija:** El orden es operador, primer operando, segundo operando.
- **Notación postfija:** El orden es primer operando, segundo operando, operador.

Normalmente los seres humanos utilizamos la notación infija, por ejemplo: $2 + 3$. Sin embargo, la notación infija tiene el siguiente problema: si una expresión tiene más de un operador puede existir ambigüedad sobre cuál es el orden en el que se deben realizar las operaciones. Por ejemplo, la expresión $8/4/2$ se puede interpretar como $(8/4)/2$ o bien como $8/(4/2)$. Las otras notaciones no sufren este problema.

Para resolver estas ambigüedades, se añaden reglas denominadas orden de precedencia de operadores. Cuando dos operadores compiten por el mismo operando como en el ejemplo anterior, el primer y el segundo operador de división se disputan el operando 4, gana el operador que tiene mayor precedencia, o si dos operadores tienen la misma precedencia, se efectúa primero la operación situada más a la izquierda.

Las reglas de precedencia habituales son que los operadores “/” y “*” tienen igual precedencia y ganan al resto de operadores. Mientras que los operadores “+” y “-” tienen igual precedencia.

Así, la expresión $8/4/2$ se evalúa como $(8/4)/2$, y $2+3*4$ se evalúa como $2+(3*4)$. Si deseamos cambiar el orden de evaluación, se pueden agrupar partes de una expresión utilizando paréntesis. En el resto de las notaciones no es necesario utilizar paréntesis ya que siempre podemos conocer el orden exacto de evaluación sin que exista ambigüedad.

Por ejemplo, si deseamos representar las expresiones $(2+(3*4))$ y $((2+3)*4)$ en las notaciones mencionadas, el resultado sería:

Notación	$(2 + (3*4))$	$((2+3)*4)$
Prefija	+ 2 * 3 4	* + 2 3 4
Infija	2 + 3 * 4	(2 + 3) * 4
Postfija	2 3 4 * +	2 3 + 4 *

2.2 Requisitos funcionales

Para el diseño y posterior implementación de la solución, es necesario que consideres los siguientes requerimientos funcionales:

- Tu programa deberá esperar que el usuario ingrese una expresión aritmética utilizando la notación infija (como una cadena de caracteres).
- Su programa debe verificar la posible existencia de los siguientes errores:
 - Se encontró un símbolo no válido. Ejemplo del error: $21+b$.
 - Error en el balance de paréntesis. Ejemplo del error: $(21+4$
 - Se esperaba un operando. Ejemplo del error: $(2 +)/4$
 - Se esperaba un operador. Ejemplo del error: $21 \ 4$
- Si la expresión escrita por el usuario es correcta, el programa deberá mostrar la expresión que escribió el usuario, pero en notación postfija.
- Su programa deberá imprimir en pantalla el resultado de evaluar la expresión matemática que escribió el usuario.

Ejemplo de entrada de datos

$$2 + 3 * 5 / 2 - 1$$

Ejemplo de salida

Posfijo: 2 3 5 * 2 / + 1 -

Evaluación: 8.5

Ejemplo de entrada de datos

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

Ejemplo de salida

Posfijo: 2 3 + 5 2 - * 1 /

Evaluación: 15

2.3 Requisitos para el desarrollo

Para la implementación del programa es necesario que utilices los contenedores que has implementado en el curso, los operadores que se considerarán serán solo: +, -, *, / y los paréntesis. Los operandos válidos son números enteros representados en el sistema numérico decimal.

3. Evaluación

Los productos que debe entregar en CANVAS son:

- Código fuente de sus contenedores y solución del problema (*.c y *.h)
- Reporte de la actividad (consultar plantilla)

Los criterios ha considerar para evaluar su actividad son los siguientes:

Requerimientos funcionales (80%)

Funcionalidades	10	5	0
Conversión de infijo a post fijo	Funciona de manera correcta	Tiene errores de sintaxis	Tiene errores de lógica o no fue implementada
Evaluación de la expresión matemática	Funciona de manera correcta	Tiene errores de sintaxis	Tiene errores de lógica o no fue implementada
Identifica símbolos no válidos	Funciona de manera correcta	Tiene errores de sintaxis	Tiene errores de lógica o no fue implementada
Identifica la falta de un operador	Funciona de manera correcta	Tiene errores de sintaxis	Tiene errores de lógica o no fue implementada
Identifica la falta de un operando	Funciona de manera correcta	Tiene errores de sintaxis	Tiene errores de lógica o no fue implementada

Nota: para tener derecho a calificación, es requisito indispensable utilizar contenedores genéricos (Stack y/o Queue).

Reporte (20%)

Criterios	10	5	0
Portada	Incluye los elementos necesarios para identificar la asignatura, el trabajo realizado, la institución educativa, los integrantes del equipo y fecha. Adicionalmente, tiene una buena presentación.	Incluye los elementos necesarios para identificar la asignatura, el trabajo realizado, la institución educativa, los integrantes del equipo y fecha. Sin embargo, tiene una mala presentación.	No se incluye o no cumple con los requisitos para una calificación mayor
Introducción	Brinda un resumen de su trabajo que permite conocer al lector de manera breve cuál es el contenido de su trabajo: problema, propuesta de solución y resultados.	Brinda un resumen de su trabajo, pero no permite conocer al lector de manera breve cuál es el contenido de su trabajo: problema, propuesta de solución y resultados.	No se incluye o no cumple con los requisitos para una calificación mayor
Análisis del problema	Brinda evidencia del análisis realizado sobre el problema: incluye redacción y diagramas.	Brinda evidencia del análisis realizado sobre el problema: incluye redacción, pero no incluye diagramas.	No se incluye o no cumple con los requisitos para una calificación mayor
Diseño de la solución	Brinda evidencia del diseño de la solución que se realizó en base aun análisis previo del problema: incluye redacción y diagramas.	Brinda evidencia del diseño de la solución que se realizó en base aun análisis previo del problema: incluye redacción y, pero no incluye diagramas.	No se incluye o no cumple con los requisitos para una calificación mayor
Pruebas	Muestra evidencia de las pruebas que fueron realizadas para validar cada requerimiento funcional.	Muestra evidencia de las pruebas que fueron realizadas, pero no son suficientes para validar cada requerimiento funcional.	No se incluye
Conclusiones	Brinda evidencia de una reflexión y discusión realizada entre los autores del trabajo respecto a: problemas, aprendizajes, resultados y oportunidades de mejora.	Brinda evidencia de una reflexión y discusión realizada entre los autores del trabajo, sin embargo, no aborda alguno de los siguientes aspectos: problemas, aprendizajes, resultados o oportunidades de mejora.	No se incluye