

Laboratorio 04: Analizador Descendente Recursivo

Profesor: Rommel Quintanilla

Semestre: 2021-II, Unidad: 01

Entregable: Código fuente en C++.**Problema: 1**

Implementar un analizador sintáctico utilizando el método descendente recursivo para la siguiente gramática:

$\text{exp} \rightarrow \text{exp opsuma term} \mid \text{term}$	$\text{exp} \rightarrow \text{term} \{\text{opsuma exp}\}$
$\text{opsuma} \rightarrow + \mid -$	$\text{term} \rightarrow \text{factor} \{\text{opmult term}\}$
$\text{term} \rightarrow \text{term opmult factor} \mid \text{factor}$	$\text{factor} \rightarrow (\text{exp}) \mid \text{numero}$
$\text{opmult} \rightarrow *$	$\text{opsuma} \rightarrow + \mid -$
$\text{factor} \rightarrow (\text{exp}) \mid \text{numero}$	$\text{opmult} \rightarrow *$

Nota: El no terminal *numero* puede ser considerado únicamente para valores enteros (con signo).

(a) Incluir la posibilidad de que el analizador sintáctico informe los errores encontrados, indicando la posición del error, el token recibido y el esperado según sea el caso.

(b) El analizador debe retornar el árbol de análisis sintáctico.

(c) Permitir que el analizador pueda retornar la evaluación de una cadena válida. Ejemplo:

Entrada: 34 - 3 * 42

Salida: -92

Entrada: (34 - 3) * 42

Salida: 1302

Entrada: 34 - 3 - 42

Salida: -11

34 - 3 * 42

term opsuma exp

exp -> term {opsuma exp}

exp-> factor {opmult term} {- exp}

exp-> numero {opmult term} - term {opsuma exp}

exp -> numero {opmult term} - factor{opmult term} {opsuma exp}

exp-> numero {opmult term} - numero{opmult term} {opsuma exp}

exp-> numero - numero * term

exp-> numero - numero *factor

exp -> numero - numero * numero