

Gramática del Guión

$Rexp \rightarrow Rexp \mid Rexp$

$Rexp \rightarrow Rexp \ Rexp$

$Rexp \rightarrow Rexp \ ^*$

$Rexp \rightarrow (\ Rexp \)$

$Rexp \rightarrow letter$

Gramática No Ambigua: Reflejando Asociatividad a Izquierdas y Precedencia de los Operadores

$Rexp1 \rightarrow Rexp1 \mid Rexp2$

$Rexp1 \rightarrow Rexp2$

$Rexp2 \rightarrow Rexp2 \ Rexp3$

$Rexp2 \rightarrow Rexp3$

$Rexp3 \rightarrow Rexp3 \ ^*$

$Rexp3 \rightarrow (\ Rexp1 \)$

$Rexp3 \rightarrow letter$

Gramática sin Recursividad por la Izquierda

$Rexp1 \rightarrow Rexp2 \ Disjunct$

$Disjunct \rightarrow \mid Rexp2 \ Disjunct$

$Disjunct \rightarrow \epsilon$

$Rexp2 \rightarrow Rexp3 \ Concat$

$Concat \rightarrow Rexp3 \ Concat$

$Concat \rightarrow \epsilon$

$Rexp3 \rightarrow (\ Rexp1 \) \ KClosure$

$Rexp3 \rightarrow letter \ KClosure$

$KClosure \rightarrow \ ^* \ KClosure$

$KClosure \rightarrow \epsilon$

Estudio de la Gramática

NO TERMINAL	FIRST	FOLLOW
Rexp1	(letter	\$)
Disjunct	ϵ	\$)
Rexp2	(letter	\$)
Concat	(letter ϵ	\$)
Rexp3	(letter	(letter \$)
KClosure	* ϵ	(letter \$)

PRODUCCIÓN	PREDICTION
Rexp1 \rightarrow Rexp2 Disjunct	(letter
Disjunct \rightarrow Rexp2 Disjunct	
Disjunct $\rightarrow \epsilon$	\$)
Rexp2 \rightarrow Rexp3 Concat	(letter
Concat \rightarrow Rexp3 Concat	(letter
Concat $\rightarrow \epsilon$	\$)
Rexp3 \rightarrow (Rexp1) KClosure	(
Rexp3 \rightarrow letter KClosure	letter
KClosure \rightarrow * KClosure	*
KClosure $\rightarrow \epsilon$	(letter \$)

Definición Dirigida por Sintaxis (DDS) de la Gramática Recursiva

Producción Gramatical	Acción Semántica
$Rexp1 \rightarrow Rexp1_1 Rexp2$	$Rexp1.ptr := mknode(' ', Rexp1_1.ptr, Rexp2.ptr)$
$Rexp1 \rightarrow Rexp2$	$Rexp1.ptr := Rexp2.ptr$
$Rexp2 \rightarrow Rexp2_1 Rexp3$	$Rexp2.ptr := mknode('.', Rexp2_1.ptr, Rexp3.ptr)$
$Rexp2 \rightarrow Rexp3$	$Rexp2.ptr := Rexp3.ptr$
$Rexp3 \rightarrow Rexp3_1 *$	$Rexp3.ptr := mkunode('*', Rexp3_1.ptr)$
$Rexp3 \rightarrow (Rexp1)$	$Rexp3.ptr := Rexp1.ptr$
$Rexp3 \rightarrow letter$	$Rexp3.ptr := mkleaf(letter)$

Esquema de Traducción (ETDS) de la Gramática No Recursiva por la Izquierda

$Rexp1 \rightarrow Rexp2 \{ Disjunct.h := Rexp2.ptr \} Disjunct \{ Rexp1.ptr := Disjunct.s \}$

$Disjunct \rightarrow | Rexp2 \{ Disjunct_1.h := mknode('|', Disjunct.h, Rexp2.ptr) \} Disjunct_1 \{ Disjunct.s := Disjunct_1.s \}$

$Disjunct \rightarrow \epsilon \{ Disjunct.s := Disjunct.h \}$

$Rexp2 \rightarrow Rexp3 \{ Concat.h := Rexp3.ptr \} Concat \{ Rexp2.ptr := Concat.s \}$

$Concat \rightarrow Rexp3 \{ Concat_1.h := mknode('.', Concat.h, Rexp3.ptr) \} Concat_1 \{ Concat.s := Concat_1.s \}$

$Concat \rightarrow \epsilon \{ Concat.s := Concat.h \}$

$Rexp3 \rightarrow (Rexp1) \{ KClosure.h := Rexp1.s \} KClosure \{ Rexp3.s := KClosure.s \}$

$Rexp3 \rightarrow letter \{ KClosure.h := mkleaf(letter) \} KClosure \{ Rexp3.s := KClosure.s \}$

$KClosure \rightarrow * KClosure_1 \{ KClosure.ptr := mkunode('*', KClosure.h) \}$

$KClosure \rightarrow \epsilon$