



**Tecnológico de Costa Rica**

**San José**

**Compiladores e Intérpretes**

Apuntes de la clase del día miércoles 17 de mayo del 2017

**Apuntador:**

Kevin Lobo Chinchilla (2015088135)

**Profesor:**

Dr. Francisco Torres Rojas

I Semestre 2017

<b>Quiz 10</b>	<b>2</b>
<b>Dudas del Proyecto</b>	<b>2</b>
<b>Análisis Sintáctico</b>	<b>2</b>
Follow	2
Cálculo de Follow()	2
Argumento de FOLLOW()	3
FOLLOW(X)	3
Observaciones de FOLLOW()	4
Ejemplo de FOLLOW()	4
Ejemplo de FIRST() Y FOLLOW()	10
Predict	11
Cálculo de PREDICT()	11
Argumento de PREDICT()	12
PREDICT(X)	12
Ejemplo de PREDICT()	12
Ejemplo de PREDICT() - 2	13
Construcción de Tabla de Parsing	14
Ejemplo de Construcción de Tabla $LL(1)$ de Parsing	14
Ejemplo de Construcción de Tabla $LL(1)$ de Parsing - 2	15
<b>Recordatorios</b>	<b>16</b>

## Quiz 10

1. Calcule el FIRST de todos los No terminales de las siguientes 3 gramáticas.

$$\begin{aligned} E &\rightarrow i E' \\ E &\rightarrow + i E' \mid \epsilon \end{aligned}$$

$$A \rightarrow a A a \mid \epsilon$$

$$\begin{aligned} D &\rightarrow T V \\ T &\rightarrow \text{int} \mid \text{float} \\ V &\rightarrow \text{id} , V \mid \text{id} \end{aligned}$$

## Dudas del Proyecto

- Para el include<> hay que replicar “lo que haga gcc”.
- El preproceso ya no es opcional, siempre corre.
- El prettyprint, como dice la especificación, es opcional y se ejecuta con un comando en la consola.

## Análisis Sintáctico

### Follow

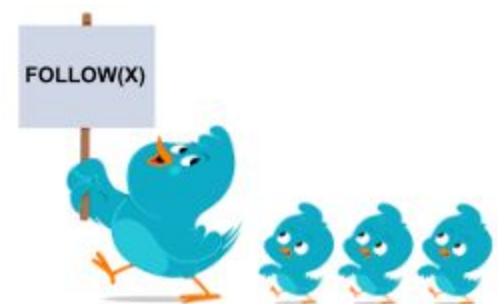
#### Cálculo de Follow()

- Sea G una ~~GFG (Glorious Free Grammar, Grammar Free Grammar)~~ CFG.
- La función **FOLLOW(X)** regresa un **conjunto de terminales** y posiblemente \$ que indican todas las “continuaciones” factibles del No terminal **X** bajo la gramática G.

\$ representa el final de una hilera.

Comparación entre first y follow:

- Ambos tienen como salidas conjuntos.
- En first, hay posibilidad de  $\epsilon$  en el conjunto de salida.
- En follow, hay posibilidad de \$ en el conjunto de salida.
- En follow, nunca puede aparecer un  $\epsilon$  en el conjunto de salida.



## Argumento de FOLLOW()

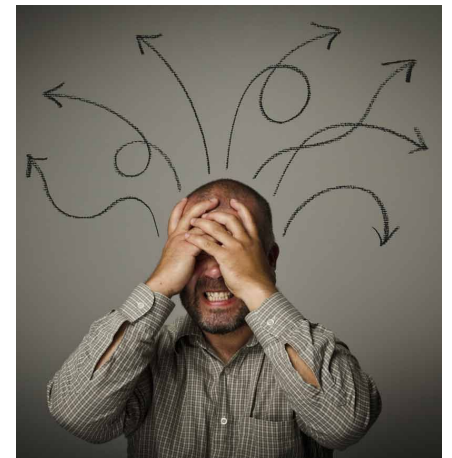
- **FOLLOW(X)**.
- El argumento **X** sólo puede ser un No terminal.
- El cálculo de **FOLLOW** es iterativo hasta que no hayan más cambios.

Nunca  $\epsilon$



## FOLLOW(X)

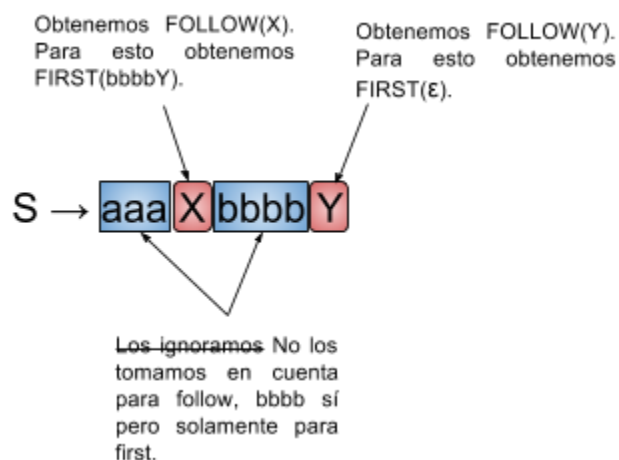
- **X** es un no terminal.
- Reglas:
  1. Si **X** es el símbolo inicial, agregue \$ a FOLLOW(**X**).
  2. Si hay una regla **A**  $\rightarrow \alpha$  **X**  $\beta$ , agregue FIRST(**X**) - { $\epsilon$ } a FOLLOW(**X**).
  3. Si hay una regla **A**  $\rightarrow \alpha$  **X**  $\beta$ , y  $\epsilon$  pertenece a FIRST( $\beta$ ), agregue FOLLOW(**A**) a FOLLOW(**X**).



$\beta$  puede ser cualquier cosa: Otro no terminal, una hilera, un terminal, incluso  $\epsilon$ .  
Cuándo es  $\epsilon$  simplemente no hay nada después del no terminal al cual le estamos aplicando FOLLOW().

¿Cómo va a servir?

- Usaremos una tabla de follow.
- Necesitamos el first de varios elementos, pero antes de follow ya hemos calculado first.

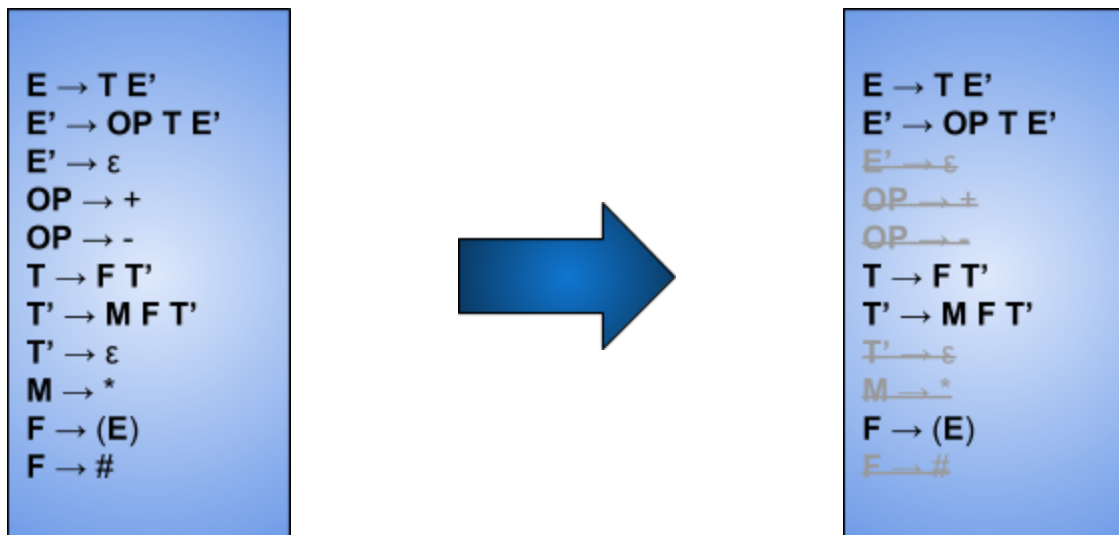


## Observaciones de FOLLOW()

- Sólo se calcula para No terminales.
- $\epsilon$  nunca es parte de FOLLOW(), pero \$ sí puede estar.
- Primero hay que calcular FIRST() de todos los No terminales.
- Se ignoran las reglas de la gramática que únicamente tengan no terminales o  $\epsilon$  en su lado derecho, las demás se procesan repetidamente hasta que no hayan cambios.

## Ejemplo de FOLLOW()

*Como en follow son innecesarias las reglas con terminales al lado derecho, estas se omiten.*



Debemos tomar en cuenta el First() de esta gramática:

No Terminal	FIRST()
E	{(, #}
E'	{+, -, ε}
OP	{+, -}
T	{(, #}
T'	{*, ε}
M	{*}
F	{(, #}

$E \rightarrow T E'$
$E' \rightarrow O P T E'$
$E' \rightarrow \epsilon$
$OP \rightarrow +$
$OP \rightarrow -$
$T \rightarrow F T'$
$T' \rightarrow M F T'$
$T' \rightarrow \epsilon$
$M \rightarrow *$
$F \rightarrow (E)$
$F \rightarrow \#$

FOLLOW()			
No Terminal	Inicio	Pasada 1	Pasada 2
E	{ \$ }		
E'	$\emptyset$		
OP	$\emptyset$		
T	$\emptyset$		
T'	$\emptyset$		
M	$\emptyset$		
F	$\emptyset$		

Siempre el inicio de la regla inicial será { \$ }.

Recordemos que lo que hacemos es “sacarle el jugo” a cada regla, por lo que en realidad no hacemos FOLLOW() del lado izquierdo de la regla, si no de los No terminales que aparezcan al lado derecho. Hacemos el Follow() regla por regla.

### **Pasada 1:**

#### **1. $E \rightarrow T E'$ :**

- a. Vemos el lado derecho, hacemos Follow(T). Para esto ocupamos First(E') el cual es  $\{+, -, \epsilon\}$ . Agregamos el First(E') al Follow(T) pero sin  $\epsilon$ . Ahora bien, como First(E') contiene a  $\epsilon$ , significa que podría no seguir nada a T, entonces debemos echar al saco de Follow(T) lo que tenga el saco de Follow(E), por lo que Follow(T) nos queda  $\{+, -, \$\}$ .
- b. Seguimos en el lado derecho y hacemos Follow(E'), si hubieran no terminales entre T y E' los omitimos. Al hacer Follow(E') debemos ver el First( $\epsilon$ ), agregamos al Follow(E') el First( $\epsilon$ ) menos  $\epsilon$ , o sea no agregamos nada al saco de Follow(E'). Como después de E' hay  $\epsilon$  entonces debemos agregar lo que tenga el saco de Follow(E) a Follow(E') que nos queda  $\{\$\}$ .

*En resumen, si el First() de lo que le sigue al No terminal del que estamos obteniendo Follow() contiene a  $\epsilon$ , debemos agregar el Follow() de la regla actual, al Follow() del No terminal. Así mismo, si lo que le sigue al No terminal del que estamos obteniendo Follow() es  $\epsilon$  (o sea nada), hacemos lo mismo.*

#### **2. $E' \rightarrow OP T E'$ :**

- a. Hacemos Follow(OP). Ocupamos el First(TE'), el cual es  $\{(\, \#\}$ . Lo agregamos al Follow(OP). Como no contiene a  $\epsilon$  terminamos con el Follow(OP) el cual nos queda  $\{(\, \#\}$ .
- b. Hacemos Follow(T). Ocupamos el First(E'), el cual es  $\{+, -, \epsilon\}$ . Lo agregamos al Follow(T), pero Follow(T) ya contiene a  $\{+, -\}$ . Observemos que First(E') contiene a  $\epsilon$ , por lo que debemos agregar al Follow(T) el contenido de Follow(E') que sería  $\{\$\}$ . También lo contenía, entonces el Follow(T) nos queda igual  $\{+, -, \$\}$ .
- c. Hacemos Follow(E'), como no le sigue nada ( $\epsilon$  para ser exactos) debemos echar al Follow(E') el Follow(E'), pero son el mismo entonces queda igual  $\{\$\}$ .

*Se omiten las reglas 3, 4, 5 dado que solo tienen terminales en su lado derecho.*

**6.  $T \rightarrow F T'$ :**

- Hacemos Follow(F). Ocupamos First( $T'$ ) que es  $\{*, \epsilon\}$ . Se lo agregamos al Follow(F) sin  $\epsilon$ . Como el First( $T'$ ) contiene a  $\epsilon$ , entonces debemos agregar al Follow(F) el contenido del Follow(T). Por lo que el Follow(F) nos queda  $\{*, +, -, \$\}$ .
- Hacemos Follow( $T'$ ), como le sigue  $\epsilon$ , agregamos al Follow( $T'$ ) el contenido del Follow(T), por lo que Follow( $T'$ ) nos queda  $\{+, -, \$\}$ .

**7.  $T \rightarrow M F T'$ :**

- Follow(M), necesitamos First( $FT'$ ) el cual es  $\{ (, \# \}$ . Se lo agregamos al Follow(M), como no contiene a  $\epsilon$  el Follow(M) queda  $\{ (, \# \}$ .
- Follow(F), necesitamos First( $T'$ ) el cual es  $\{*, \epsilon\}$ . Se lo agregamos al Follow(F) pero sin  $\epsilon$ , y Follow(F) nos queda igual porque en la regla anterior ya le habíamos agregado  $*$ . Luego como  $T'$  puede ser  $\epsilon$ , debemos agregar al Follow(F) el contenido del Follow(T). Por lo que el Follow(F) no cambia (vea el procedimiento de la regla 6, el punto a) y nos queda  $\{*, +, -, \$\}$ .
- Follow( $T'$ ), como después del no terminal  $T'$  sigue  $\epsilon$ , debemos echar al saco de Follow( $T'$ ), lo que haya en el saco de Follow(T), esto es lo mismo que hicimos en 6.b, por lo que Follow( $T'$ ) no cambia.

*Omitimos las reglas 8 y 9.*

**10.  $F \rightarrow (E)$ :**

- Omitimos el ( y hacemos Follow(E). Para esto ocupamos First( ( ) el cual es  $\{ ( \}$ , lo agregamos al Follow(E). Como el First( ( ) claramente no contiene a  $\epsilon$ , entonces Follow(E) nos queda  $\{ (, \$ \}$ .

*Omitimos la regla 11.*

Terminamos la **primera** pasada :).





$E \rightarrow T E'$
$E' \rightarrow \epsilon$
$E' \rightarrow OP T E'$
$OP \rightarrow +$
$OP \rightarrow -$
$T \rightarrow F T'$
$T' \rightarrow M F T'$
$T' \rightarrow \epsilon$
$M \rightarrow *$
$F \rightarrow (E)$
$E \rightarrow \#$

FOLLOW()			
No Terminal	Inicio	Pasada 1	Pasada 2
E	{ $\$$ }	{ $\$, \)$ }	
E'	$\emptyset$	{ $\$$ }	
OP	$\emptyset$	{ $(, \#$ }	
T	$\emptyset$	{ $+, -, \$$ }	
T'	$\emptyset$	{ $+, -, \$$ }	
M	$\emptyset$	{ $(, \#$ }	
F	$\emptyset$	{ $*, +, -, \$$ }	

### Pasada 2:

#### 1. $E \rightarrow T E'$ :

- Hacemos Follow(T). Para esto ocupamos First(E'). Agregamos el First(E') al Follow(T) pero sin  $\epsilon$ , ya lo contenía. Ahora bien, como First(E') contiene a  $\epsilon$  debemos echar al saco de Follow(T) lo que tenga el saco de Follow(E). Follow(T) no cambia.
- Hacemos Follow(E'). Como después de E' hay  $\epsilon$  entonces debemos agregar lo que tenga el saco de Follow(E) a Follow(E') que nos queda { $\$, \)$ }.

#### 2. $E' \rightarrow OP T E'$ :

- Hacemos Follow(OP). Ocupamos el First(TE'), el cual es {(, #}. Lo agregamos al Follow(OP), pero ya lo tenía, entonces Follow(OP) no cambia.
- Hacemos Follow(T). Ocupamos el First(E'), el cual es {+, -,  $\epsilon$ }. Lo agregamos al Follow(T), pero Follow(T) ya lo contiene. Observemos que First(E') contiene a  $\epsilon$ , por lo que debemos agregar al Follow(T) el contenido de Follow(E') que sería { $\$, \)$ }. Entonces el Follow(T) nos queda {+, -,  $\$, \)$ }.

- c. Hacemos  $\text{Follow}(E')$ , como no le sigue nada ( $\epsilon$  para ser exactos) debemos echar al  $\text{Follow}(E')$  el  $\text{Follow}(E')$ , pero son el mismo entonces queda igual  $\{\$, \}$ .

*Se omiten las reglas 3, 4, 5 dado que solo tienen terminales en su lado derecho.*

#### 6. $T \rightarrow F T'$ :

- a. Hacemos  $\text{Follow}(F)$ . Ocupamos  $\text{First}(T')$  que es  $\{*, \epsilon\}$ . Se lo agregamos al  $\text{Follow}(F)$  sin  $\epsilon$ . Como el  $\text{First}(T')$  contiene a  $\epsilon$ , entonces debemos agregar al  $\text{Follow}(F)$  el contenido del  $\text{Follow}(T)$ . Por lo que el  $\text{Follow}(F)$  nos queda  $\{*, +, -, \$, \}$ .
- b. Hacemos  $\text{Follow}(T')$ , como le sigue  $\epsilon$ , agregamos al  $\text{Follow}(T')$  el contenido del  $\text{Follow}(T)$ , por lo que  $\text{Follow}(T')$  nos queda  $\{+, -, \$, \}$ .

#### 7. $T \rightarrow M F T'$ :

- a.  $\text{Follow}(M)$ , necesitamos  $\text{First}(FT')$  el cual es  $\{ (, \# \}$ . Se lo agregamos al  $\text{Follow}(M)$ , ya lo tenía entonces el  $\text{Follow}(M)$  no cambia.
- b.  $\text{Follow}(F)$ , necesitamos  $\text{First}(T')$  el cual es  $\{*, \epsilon\}$ .  $\text{Follow}(F)$  ya contiene  $*$ . Luego como  $T'$  puede ser  $\epsilon$ , debemos agregar al  $\text{Follow}(F)$  el contenido del  $\text{Follow}(T)$ . Por lo que el  $\text{Follow}(F)$  no cambia.
- c.  $\text{Follow}(T')$ , como después del no terminal  $T'$  sigue  $\epsilon$ , debemos echar al saco de  $\text{Follow}(T')$ , lo que haya en el saco de  $\text{Follow}(T)$ , esto es lo mismo que hicimos en 6.b, por lo que  $\text{Follow}(T')$  no cambia.

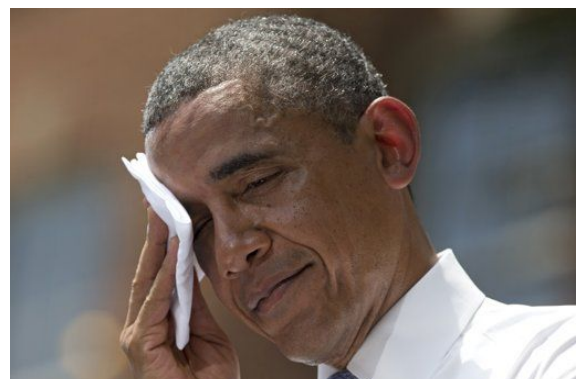
*Omitimos las reglas 8 y 9.*

#### 10. $F \rightarrow (E)$ :

- a. Omitimos el  $($  y hacemos  $\text{Follow}(E)$ . Para esto ocupamos  $\text{First}( ( )$  el cual es  $\{ ( \}$ , lo agregamos al  $\text{Follow}(E)$ . Ya lo contenía, por lo que el  $\text{Follow}(E)$  no cambia.

*Omitimos la regla 11.*

Terminamos la **segunda** pasada y  
terminamos :D.



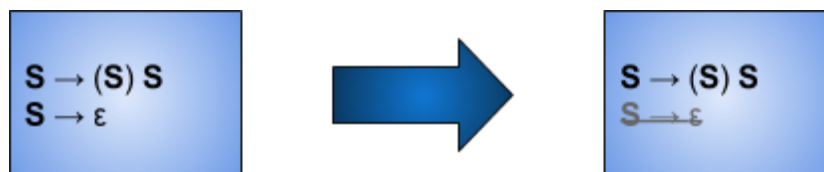
## Resultado

$E \rightarrow T E'$   
 $E' \rightarrow OP T E'$   
 $E' \rightarrow \epsilon$   
 $OP \rightarrow +$   
 $OP \rightarrow -$   
 $T \rightarrow F T'$   
 $T' \rightarrow M F T'$   
 $T' \rightarrow \epsilon$   
 $M \rightarrow *$   
 $F \rightarrow (E)$   
 $F \rightarrow \#$

FOLLOW()			
No Terminal	Inicio	Pasada 1	Pasada 2
E	{ $\$$ }	{ $\$, )$ }	{ $\$, )$ }
E'	$\emptyset$	{ $\$$ }	{ $\$, )$ }
OP	$\emptyset$	{ $(, \#$ }	{ $(, \#$ }
T	$\emptyset$	{ $+, -, \$$ }	{ $+, -, \$, )$ }
T'	$\emptyset$	{ $+, -, \$$ }	{ $+, -, \$, )$ }
M	$\emptyset$	{ $(, \#$ }	{ $(, \#$ }
F	$\emptyset$	{ $*, +, -, \$$ }	{ $*, +, -, \$, )$ }

Puede hacerse una tercera pasada para verificar que los Follow() no van a cambiar.

## Ejemplo de FIRST() Y FOLLOW()



FIRST()		
No Terminal	Inicio	Pasada 1
S	$\emptyset$	{ $\epsilon, ($ }

FOLLOW()		
No Terminal	Inicio	Pasada 1
S	{ $\$$ }	{ $\$, )$ }

*En McDonald's de Europa dan combos  
con cerveza, aquí no :(.*



*Torres y Varguitas: Hace un tiempo, ambos profesores fueron a una conferencia de Intel. Varguitas eligió habitación acompañada y nuestro señor Torres individual, pero le tocó a Varguitas como compañero de habitación. Un día, como Varguitas es raro, Torres llegó hasta... borracho y él estaba planchando ropa, y le dijo que si quería que le planchara la ropa, pero Torres con costos y andaba un calzoncillo para todos los días de la conferencia.*



## Predict

### Cálculo de PREDICT()

- Sea  $G$  una ~~GFG~~(Glorious Free Grammar, Grammar Free Grammar) CFG.
- La función **PREDICT**( $X$ ) regresa un **conjunto de terminales** y posiblemente  $\$$  que predicen cuando hay que usar la regla  $X$ .



## Argumento de PREDICT()

- **PREDICT(X)**.
- El argumento **X** sólo puede ser una **regla de la gramática**.
- Antes se deben calcular los FIRST() y FOLLOW().



## PREDICT(X)

- **X** es una regla de la forma **A**  $\rightarrow$   $\alpha$ .

if (FIRST( $\alpha$ ) incluye a  $\epsilon$ )

$$\text{PREDICT}(\mathbf{X}) = (\text{FIRST}(\alpha) - \{\epsilon\}) \cup \text{FOLLOW}(\mathbf{A})$$

else

$$\text{PREDICT}(\mathbf{X}) = \text{FIRST}(\alpha)$$



## Ejemplo de PREDICT()

1. **S**  $\rightarrow$  (**S**) **S**
2. **S**  $\rightarrow \epsilon$

No Terminal	First()	Follow()
S	{ $\epsilon$ , ( }	{\$, ) }

Regla	FIRST( $\alpha$ )	FOLLOW(A)	PREDICT()
<b>S</b> $\rightarrow$ ( <b>S</b> ) <b>S</b>	{ ( }	{\$, ) }	{ ( }
<b>S</b> $\rightarrow \epsilon$	{ $\epsilon$ }	{\$, ) }	{\$, ) }

**PREDICT(1):**  $\text{First}((S) S) = \{ ( \}$ . No contiene a  $\epsilon$ , entonces  $\text{PREDICT}(1) = \{ ( \}$ .

**PREDICT(2):**  $\text{First}(\epsilon) = \{\epsilon\}$ . Contiene a  $\epsilon$ , entonces  $\text{PREDICT}(2) = (\text{FIRST}(\epsilon) - \{\epsilon\}) \cup \text{FOLLOW}(S) = \{\$, )\}$ .

Ejemplo de  $\text{PREDICT}()$  - 2

1. $E \rightarrow T E'$
2. $E' \rightarrow \text{OP } T E'$
3. $E' \rightarrow \epsilon$
4. $\text{OP} \rightarrow +$
5. $\text{OP} \rightarrow -$
6. $T \rightarrow F T'$
7. $T' \rightarrow M F T'$
8. $T' \rightarrow \epsilon$
9. $M \rightarrow *$
10. $F \rightarrow (E)$
11. $F \rightarrow \#$

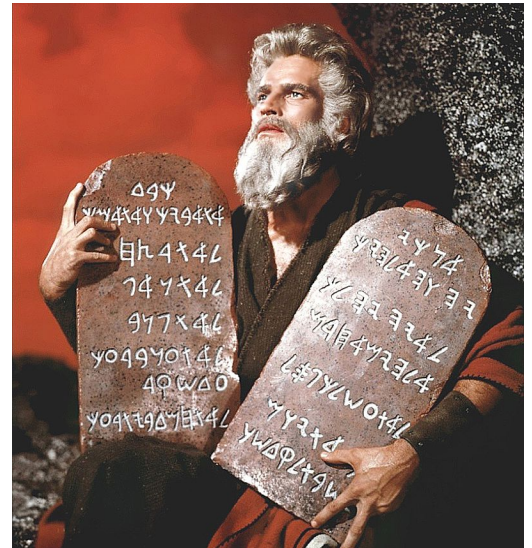
No Terminal	FIRST()	FOLLOW()
E	{(, #}	{\$, )}
E'	{+, -, $\epsilon$ }	{\$, )}
OP	{+, -}	{(, #}
T	{(, #}	{+, -, \$, )}
T'	{*, $\epsilon$ }	{+, -, \$, )}
M	{*}	{(, #}
F	{(, #}	{+, -, \$, )}

Regla	FIRST( $\alpha$ )	PREDICT()
1. $E \rightarrow T E'$	{(, #}	{(, #}
2. $E' \rightarrow \text{OP } T E'$	{+, -}	{+, -}
3. $E' \rightarrow \epsilon$	{ $\epsilon$ }	{\$, )}
4. $\text{OP} \rightarrow +$	{+}	{+}
5. $\text{OP} \rightarrow -$	{-}	{-}
6. $T \rightarrow F T'$	{(, #}	{(, #}
7. $T' \rightarrow M F T'$	{*}	{*}
8. $T' \rightarrow \epsilon$	{ $\epsilon$ }	{+, -, \$, )}
9. $M \rightarrow *$	{*}	{*}
10. $F \rightarrow (E)$	{ ( }	{ ( }
11. $F \rightarrow \#$	{#}	{#}

*Somos monstruos peludos en preprocesar gramática, FIRST, FOLLOW y PREDICT.*

## Construcción de Tabla de Parsing

- Tabla **M**:
  - Una fila por cada No Terminal y \$.
  - Una columna por cada terminal y \$.
- Se calculan todos los FIRST() y los FOLLOW().
- Se calcula PREDICT() de cada regla.
- Para toda regla  $A \rightarrow \alpha$ :
  - Para cada elemento **a** en  $PREDICT(A \rightarrow \alpha)$  agregue regla  $A \rightarrow \alpha$  en  $M[A][a]$



## Ejemplo de Construcción de Tabla LL(1) de Parsing

Regla	PREDICT()
1. $S \rightarrow (S) S$	{ ( }
2. $S \rightarrow \epsilon$	{ \$, ) }

Tabla de Parsing

No Terminal	(	)	\$
S	1	2	2

En cada casilla se agrega la regla donde el lado izquierdo sea el no terminal de la fila, y en donde en el PREDICT() aparezca el no terminal de la columna.

## Ejemplo de Construcción de Tabla LL(1) de Parsing - 2

Regla	PREDICT()
1. $E \rightarrow T E'$	{(, #}
2. $E' \rightarrow OP T E'$	{+, -}
3. $E' \rightarrow \epsilon$	{\$, )}
4. $OP \rightarrow +$	{+}
5. $OP \rightarrow -$	{-}
6. $T \rightarrow F T'$	{(, #}
7. $T' \rightarrow M F T'$	{*}
8. $T' \rightarrow \epsilon$	{+, -, \$, )}
9. $M \rightarrow *$	{*}
10. $F \rightarrow (E)$	{ ( }
11. $F \rightarrow \#$	{#}

Tabla de Parsing

No Terminal	(	#	)	+	-	*	\$
E	1	1					
E'			3	2	2		3
OP				4	5		
T	6	6					
T'			8	8	8	7	8
M						9	
F	10	11					

PD: Los de la Latina y la Ulacit no tienen idea de cómo hacer una tabla de parsing.

PD2: El mae de bioinformática de la UCR está ~~mama...~~ muy perdido, el pobrecito venía de una privada :(.



## Recordatorios

- Último quiz el miércoles 24 de mayo!
- Entrega de la tarea de la gramática de Micro el miércoles 24 de mayo.
- Entrega de 3er proyecto el miércoles 31 de mayo.
- Segundo parcial el viernes 2 de junio.
- Examen final el viernes 9 de junio.
- La nota de quices y tareas sólo puede subir, entonces a ponerle!
- El profe nos va a dar donas!

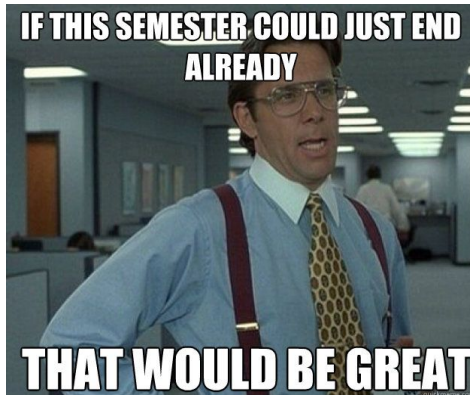
**BRACE YOURSELVES**

**END OF SEMESTER IS  
COMING**



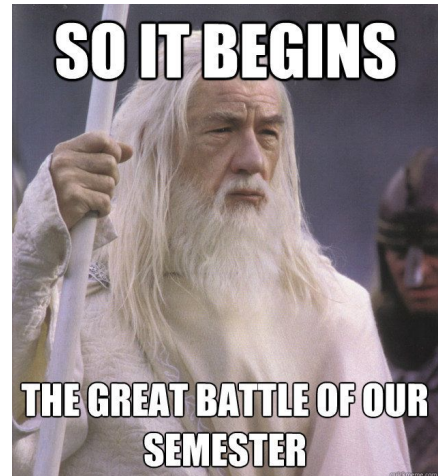
**IF THIS SEMESTER COULD JUST END  
ALREADY**

**THAT WOULD BE GREAT**



**SO IT BEGINS**

**THE GREAT BATTLE OF OUR  
SEMESTER**



**what the end of the semester  
feels like**

