

# Lenguajes y Sistemas Informáticos para la resolución de problemas complejos



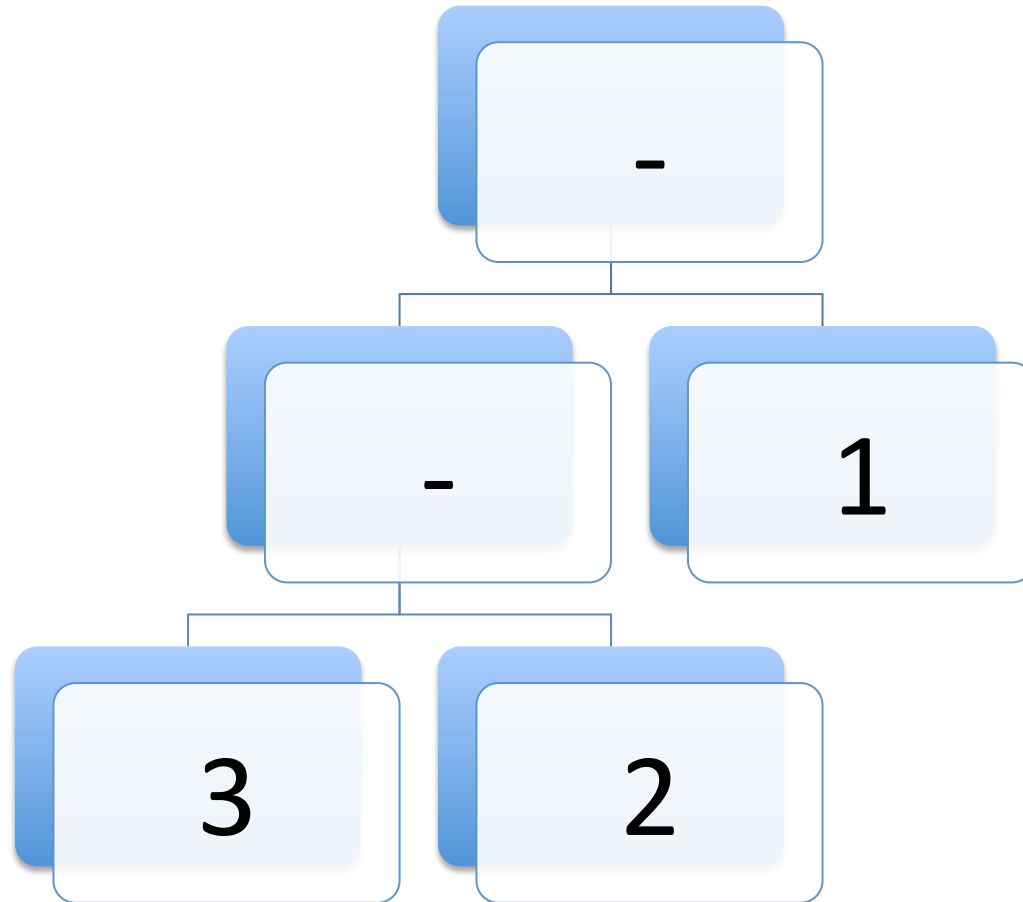
Procesadores de Lenguajes

Casiano Rodríguez León

3 - 2 - 1

# Árbol Sintáctico Abstracto

$(3-2)-1$



# Semántica 3 - 2 - 1

$$0 = 1 - 1$$

-

$$1 = 3 - 2$$

-

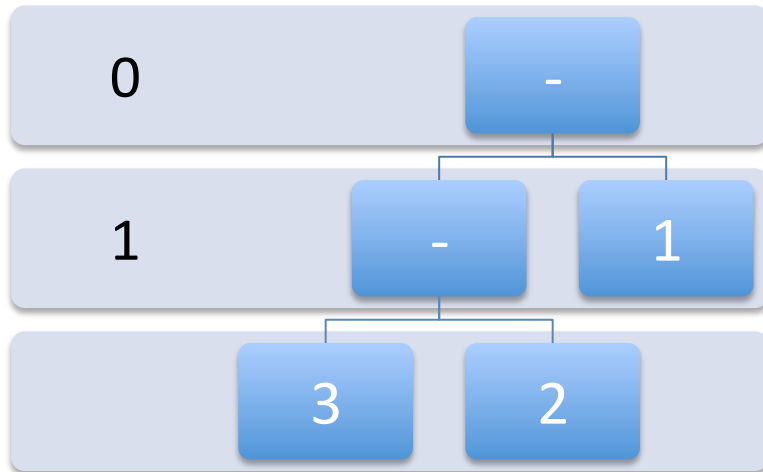
1

3

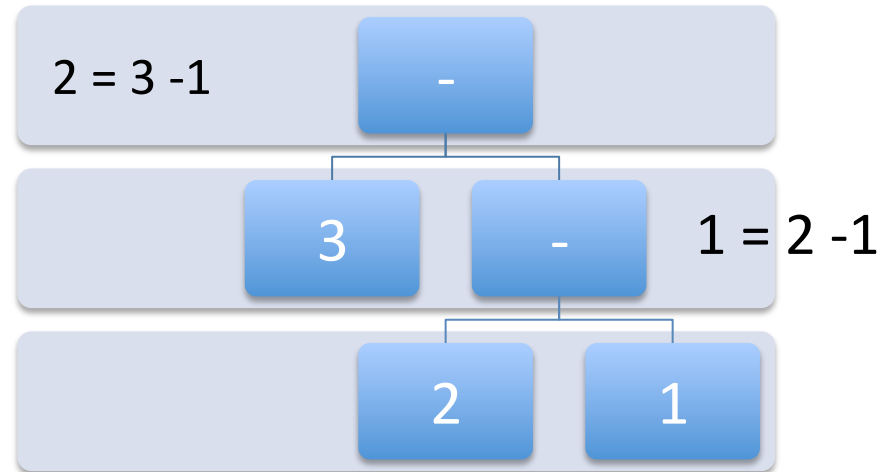
2

# Semántica y Ambigüedad

$(3-2)-1$



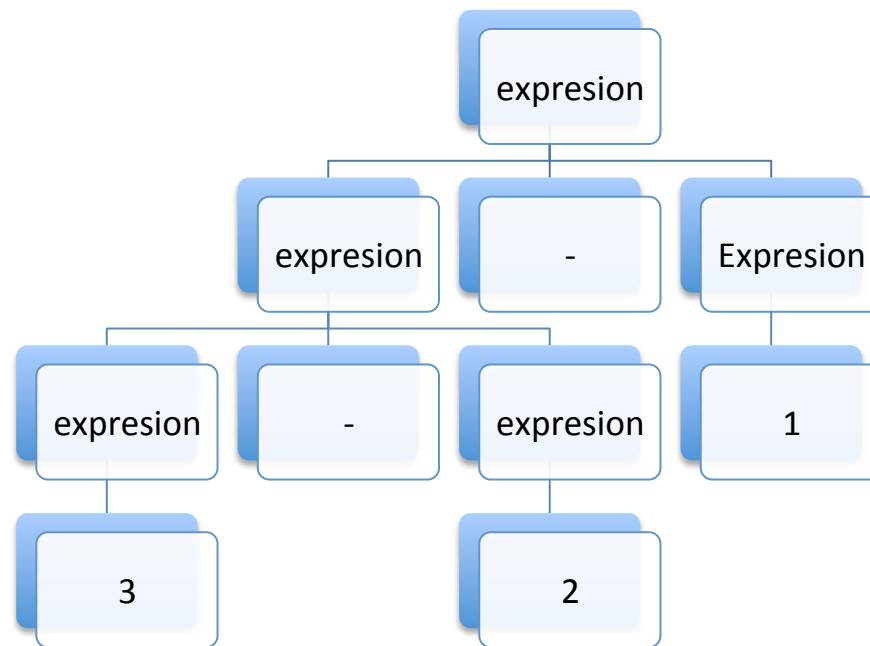
$3-(2-1)$



# Gramática Independiente del Contexto

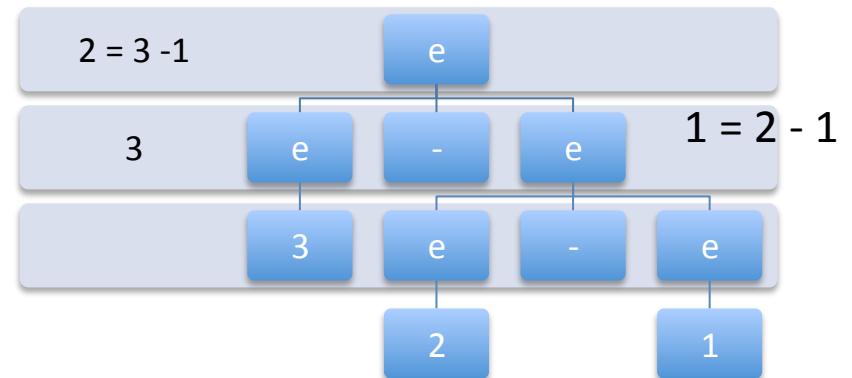
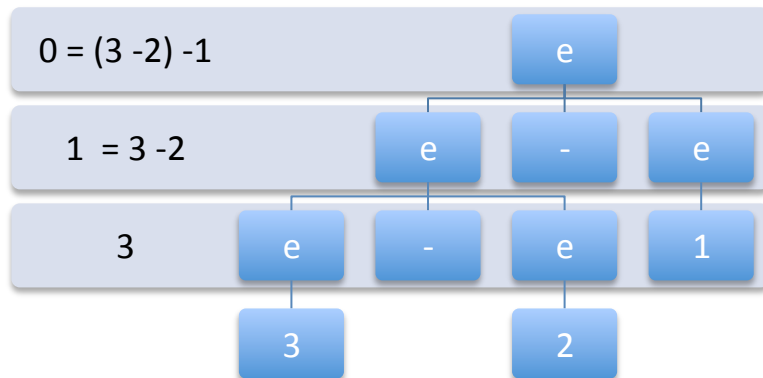
- $\text{expresion} \rightarrow \text{expresion} \text{ '-' expresion}$
- $\text{expresion} \rightarrow \text{NUMERO}$

(3-2)-1



# Gramática Ambigua

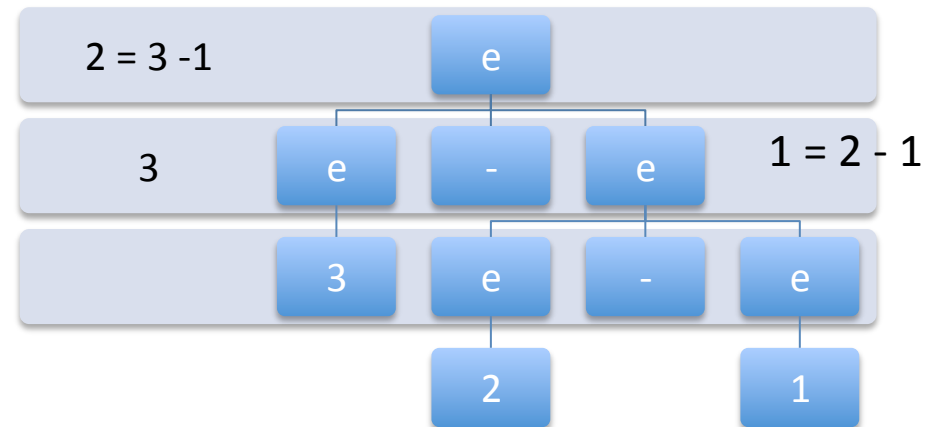
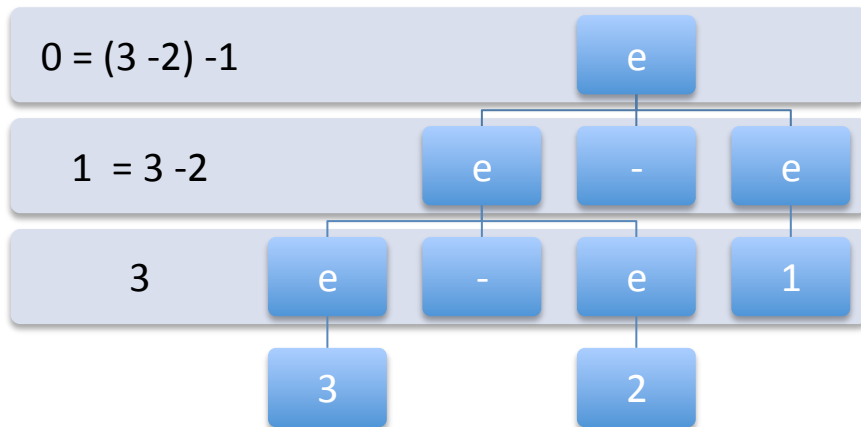
- $\text{expresion} \rightarrow \text{expresion} \text{ '-' } \text{expresion}$
- $\text{expresion} \rightarrow \text{NUMERO}$



# Esquema de Traducción (yacc)

$e \rightarrow e \text{ '-' } e \quad \{ \$\$ = \$1 - \$3; \}$   
 $e \rightarrow \text{NUM} \quad \{ \$\$ = \text{Number}(\$1); \}$

3 - 2 - 1





# Parsing: Construcción del Árbol

$e \rightarrow e \text{ '-' } e \quad \{ \$\$ = \$1 - \$3; \}$

$e \rightarrow \text{NUM} \quad \{ \$\$ = \text{Number}(\$1); \}$

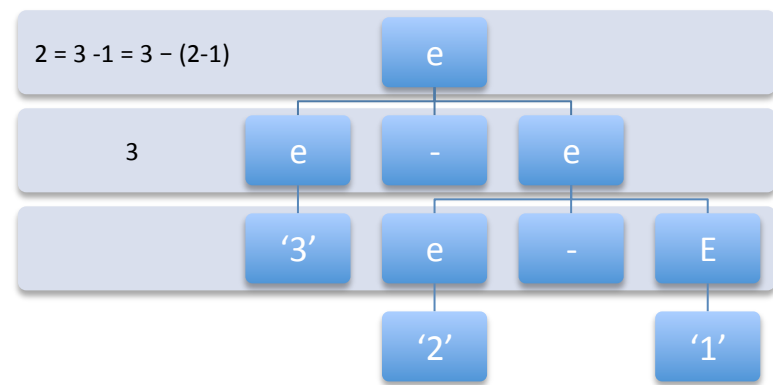
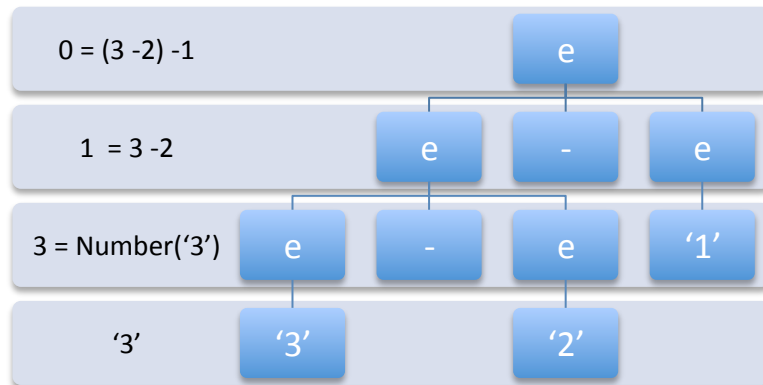
Análisis Sintáctico Ascendente:

$.3 - 2 - 1 \leq e. - 2 - 1 \leq = e -. 2 - 1 \leq = e - 2. - 1 \leq e - e. - 1$

*¿Qué hacer?*

1.  $\leq e. - 1 \leq e -. 1 \leq e - 1. \leq e - e. \leq e.$

2.  $\leq e - e -. 1 \leq e - e - 1. \leq e - e - e. \leq e - e. \leq e.$



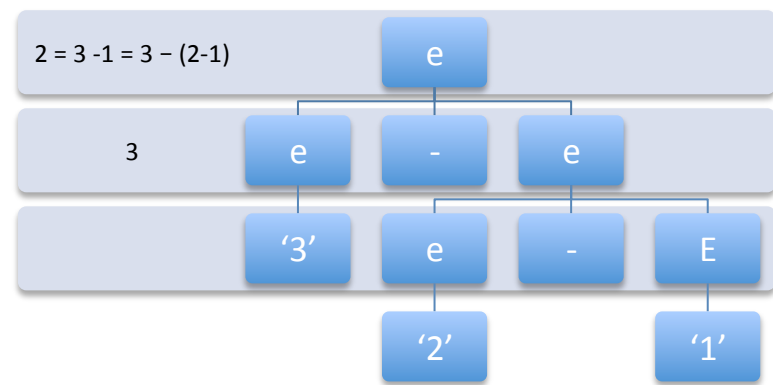
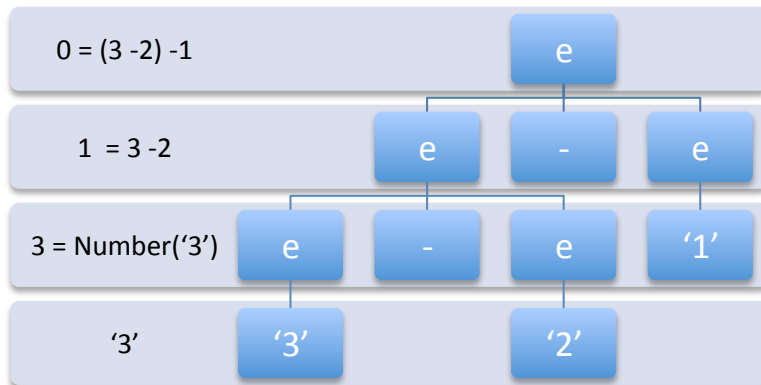
# Conflicto Shift/Reduce

.3 - 2 - 1 <= e. - 2 - 1 <= = e -. 2 - 1 <= = e - 2. - 1 <= e - e. - 1

*¿Qué hacer?*

1. <= e. - 1 <= e - . 1 <= e - 1. <= e - e. <= e.
2. <= e - e - . 1 <= e - e - 1. <= e - e - e. <= e - e. <= e.

El conflicto puede verse como una lucha entre la regla  $e \rightarrow e \text{ '-'}$  e y el terminal/token  $\text{'-'}$



# Un programa Yacc

%left ' \_ ' ← En la lucha entre la regla  $e \rightarrow e \text{ ' - '}$  e y el terminal/token  $\text{' - '}$  debe “ganar” la regla

%%

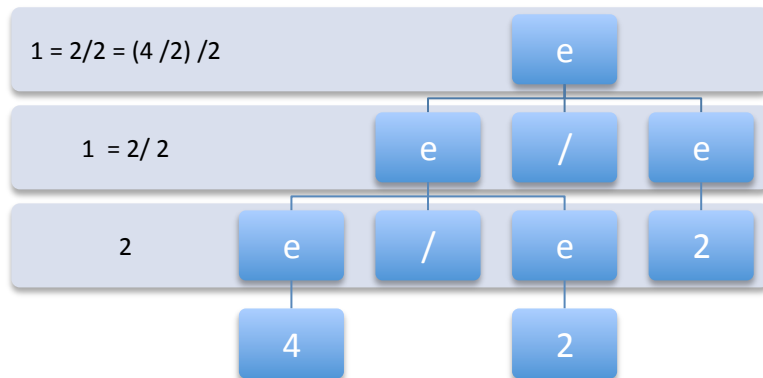
```
s : e      { return $1; }  
;
```

```
e : e ' _ ' e { $$ = $1 - $3; }  
  | NUM      { $$ = Number($1); }  
;
```

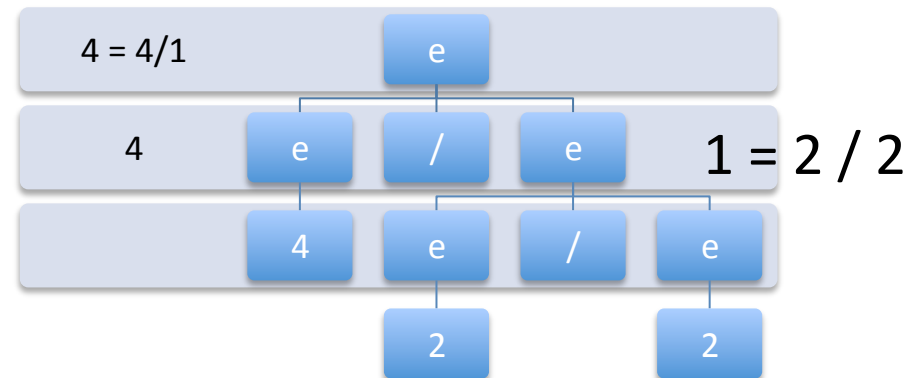
# Ambigüedad: Asociatividad

## 4/2/2

$$(4/2)/2$$



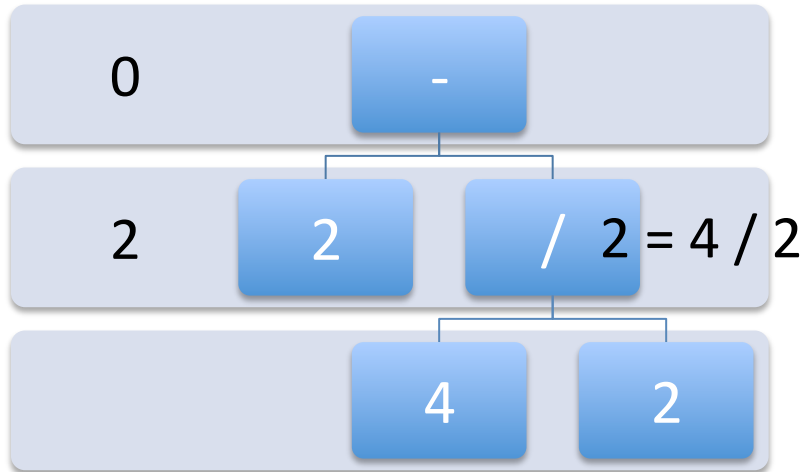
$$4/(2/2)$$



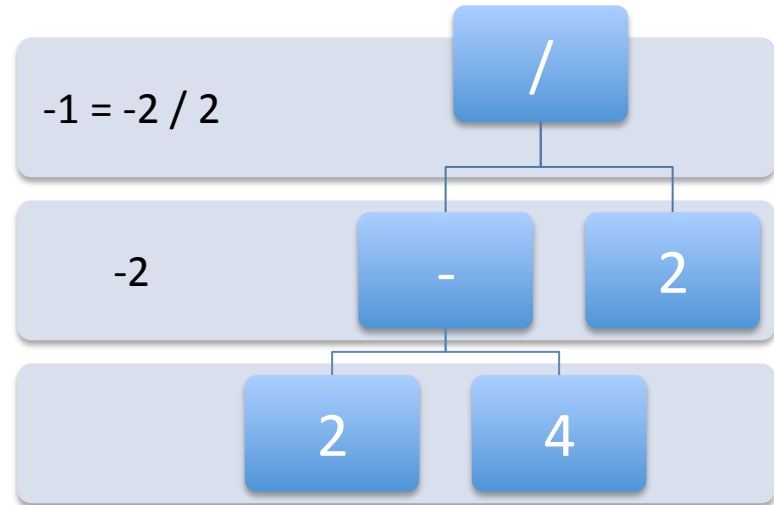
# Ambigüedad: Prioridad

```
e : e ' _ ' e { $$ = $1 - $3; }  
  | e ' / ' e { $$ = $1 / $3; }  
  | NUM { $$ = Number($1); }  
  ;
```

2-(4/2)



(2-4)/2



# Ambigüedad: Prioridad

## 2-4/2

2.-4/2 <= e.-4/2<=e-.4/2<=e-4./2<=e-e./2

*¿Qué hacer?*

1. <= e./2 <= e/. 2 <= e/e.<= e.
2. <= e-e/.2 <= e-e/2.<= e-e/e.<= e-e. <= e.

**El conflicto es entre la regla  $e \rightarrow e \text{ '-'}$  e y el terminal  $\text{'/'}$**

# Ambigüedad: Prioridad



Mas prioridad

%left '-'

%left '/'

%%

e : e '-' e { \$\$ = \$1 - \$3; }

| e '/' e { \$\$ = \$1 / \$3; }

| NUM { \$\$ = Number(\$1); }

;

En la lucha entre reducir por la regla  $e \rightarrow e - e$  y desplazar el terminal '/' debe "ganar" el token