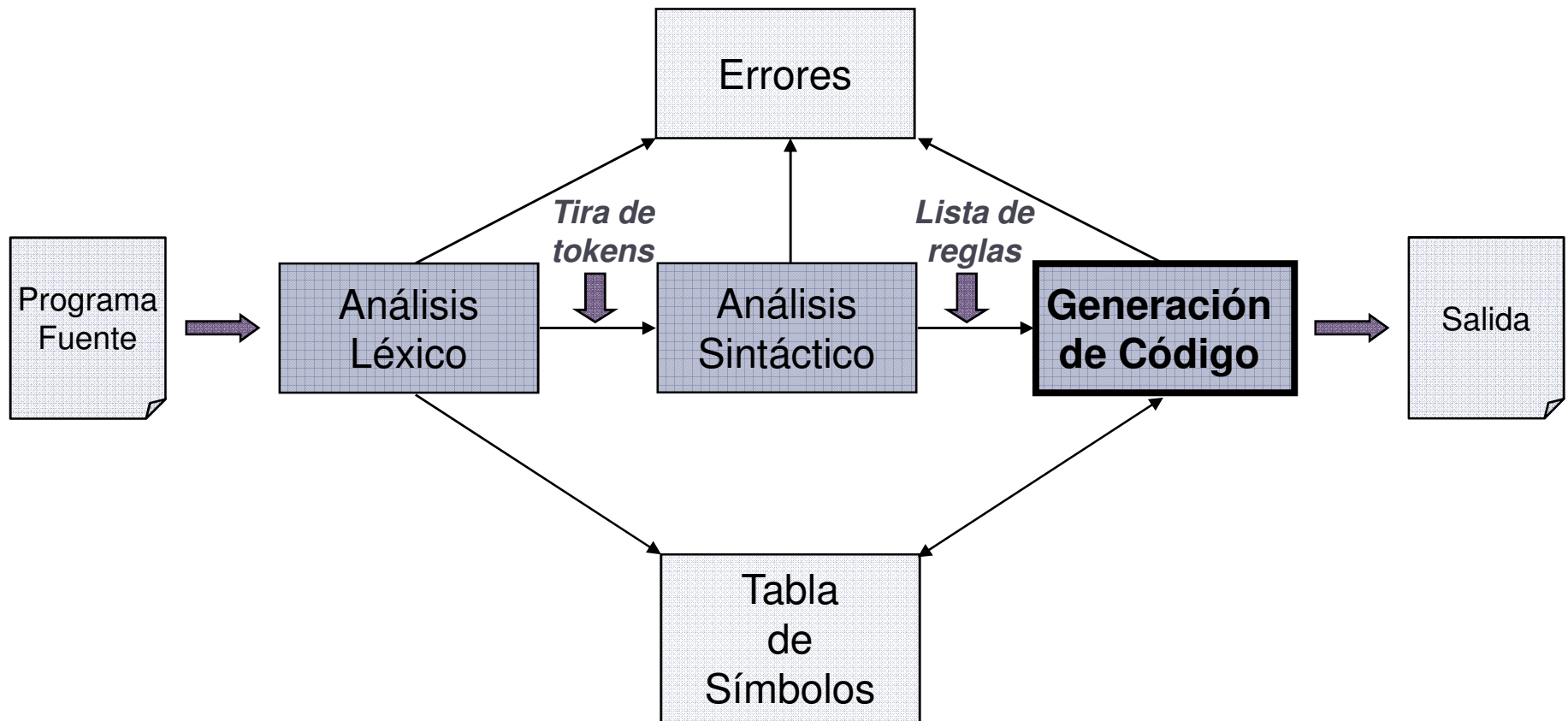


Diseño de Compiladores I

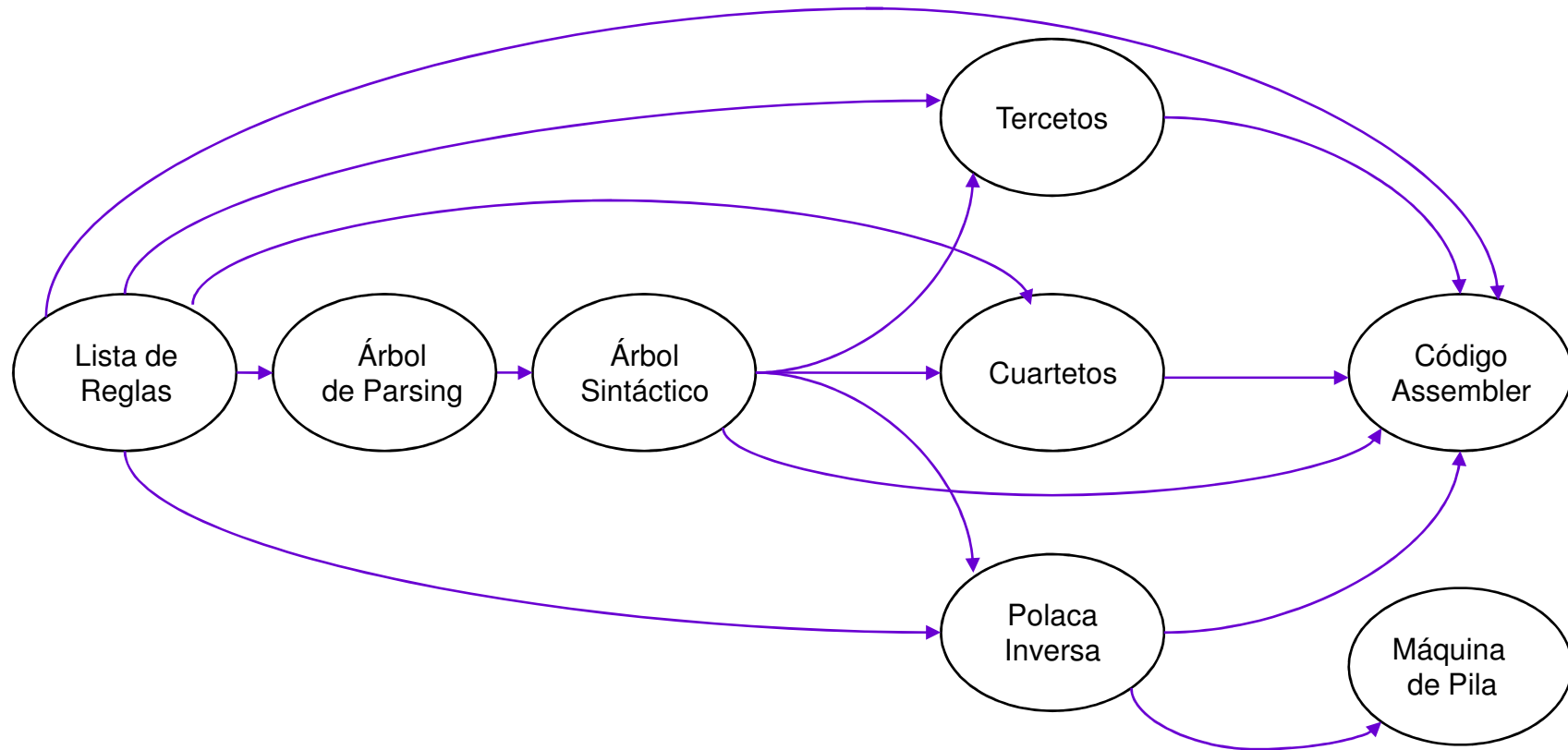
Generación de Código

Fases de la Compilación



► Generación de Código

Generación de Código



► Generación de Código

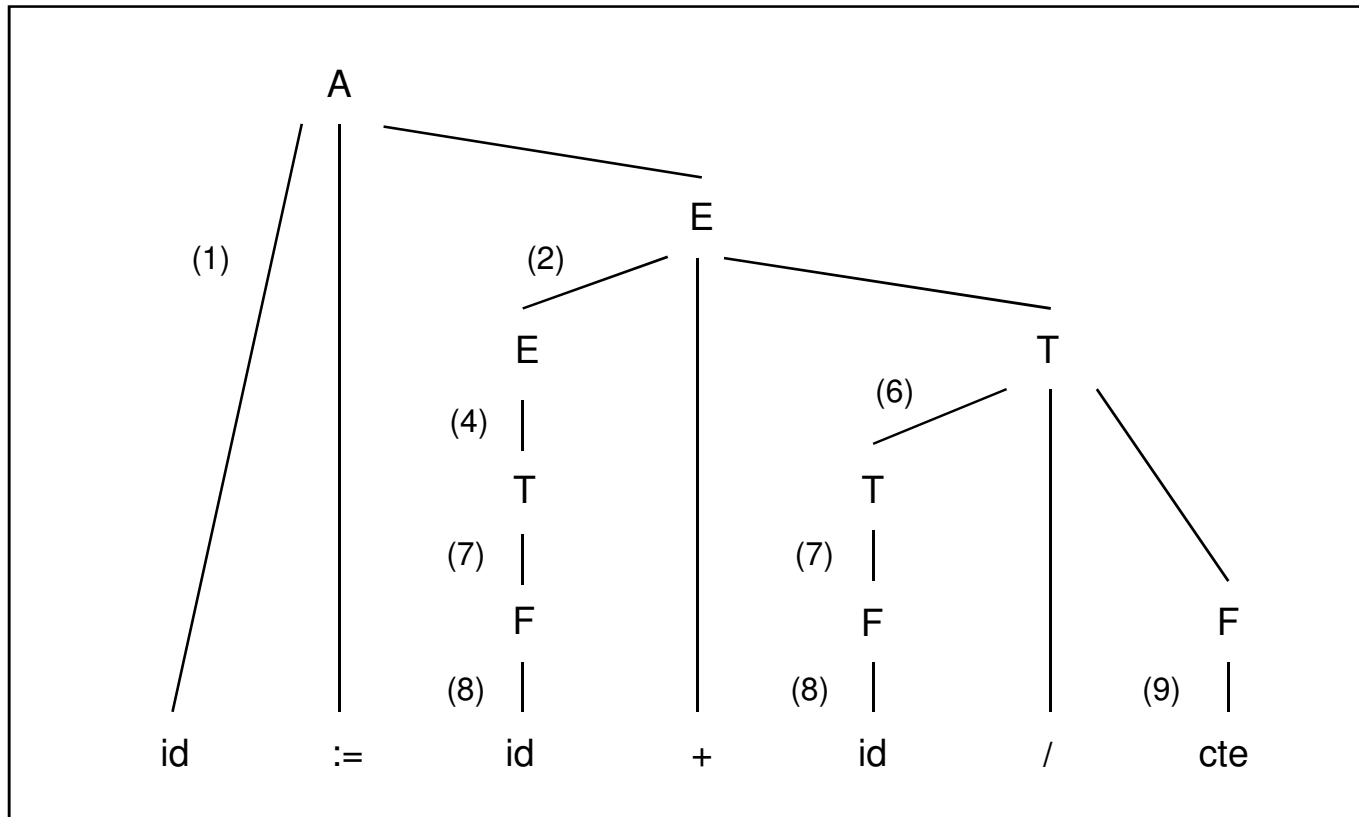
Representaciones: Gramática

- 1) $A \rightarrow \text{id} := E$
- 2) $E \rightarrow E + T$
- 3) $E \rightarrow E - T$
- 4) $E \rightarrow T$
- 5) $T \rightarrow T * F$
- 6) $T \rightarrow T / F$
- 7) $T \rightarrow F$
- 8) $F \rightarrow \text{id}$
- 9) $F \rightarrow \text{cte}$



Representaciones: Árbol de Parsing

Tasa := Base + Recargo / 100.0 → id := id + id / cte

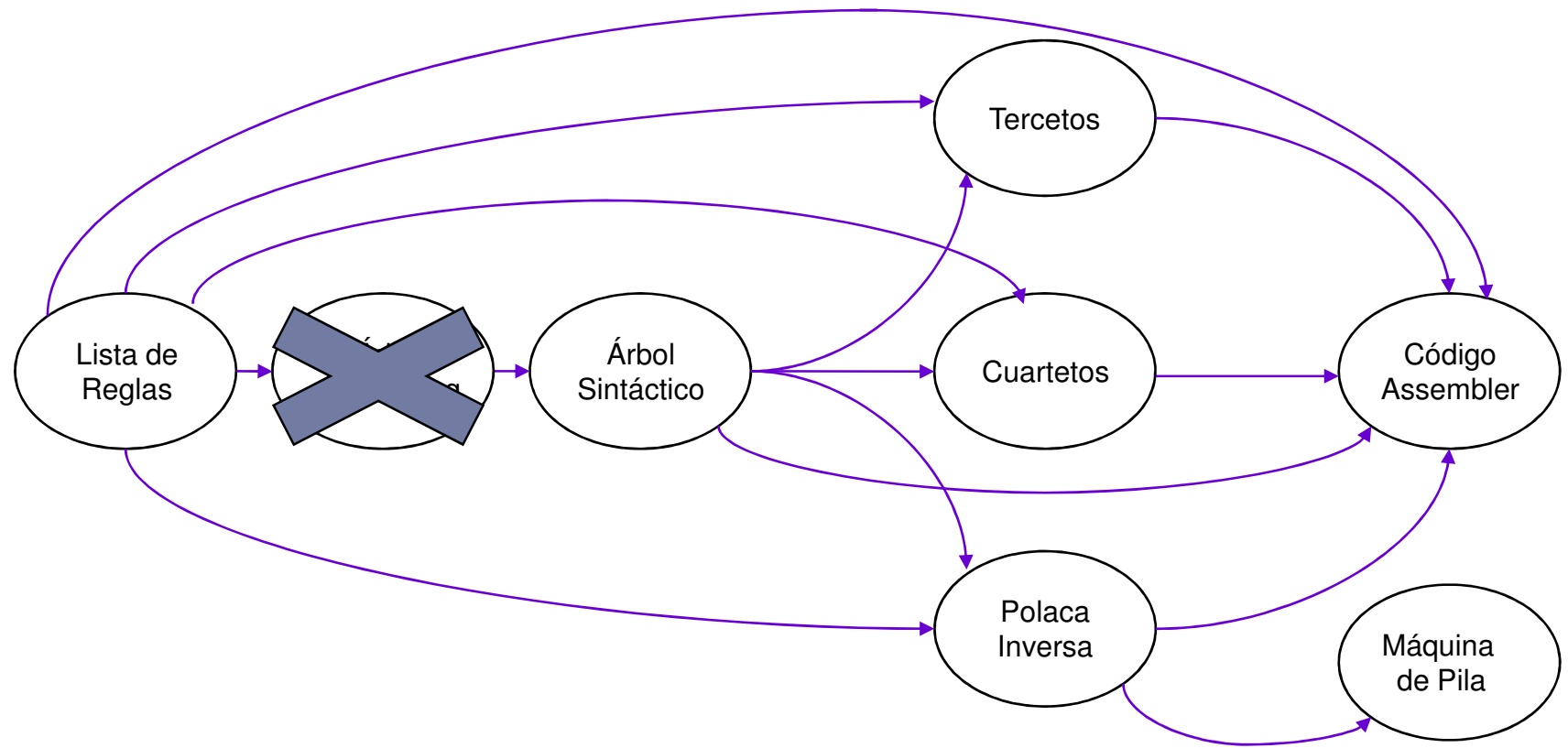


- 1) $A \rightarrow id := E$
- 2) $E \rightarrow E + T$
- 3) $E \rightarrow E - T$
- 4) $E \rightarrow T$
- 5) $T \rightarrow T * F$
- 6) $T \rightarrow T / F$
- 7) $T \rightarrow F$
- 8) $F \rightarrow id$
- 9) $F \rightarrow cte$

Lista de reglas: 8 7 4 8 7 9 6 2 1

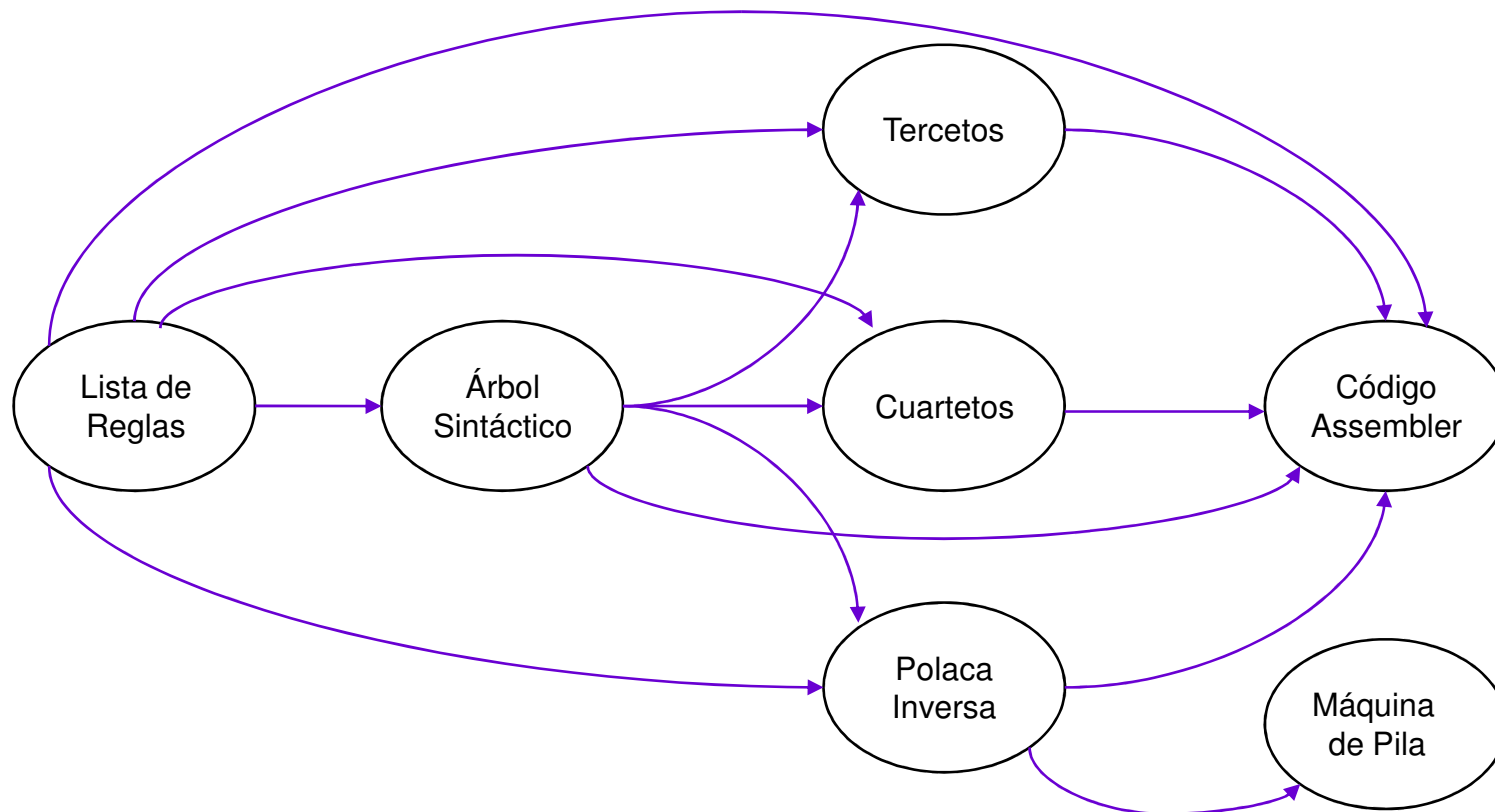


Generación de Código



Generación de Código

Generación de Código



Generación de Código

Representaciones: Árbol Sintáctico

Es una representación comprimida del árbol de parsing.

Tasa := Base + Recargo / 100.0 → ID := ID + ID / CTE

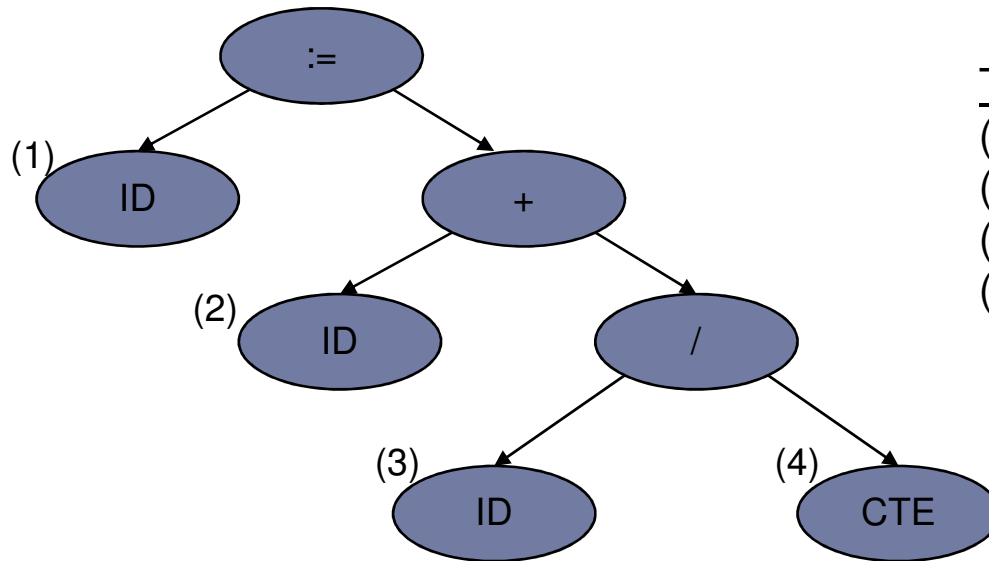


Tabla de Símbolos

(1) Tasa	real
(2) Base	real
(3) Recargo	real
(4) 100.0	real



Representaciones: Tercetos

Es una representación constituida por códigos de 3 direcciones
(operador, operando 1, operando 2)

Tasa := Base + Recargo / 100.0 → ID := ID + ID / CTE

14. ...

15. (/, Recargo, 100.0)

16. (+, Base, [15])

17. (:=, Tasa, [16])

18. ...

Tabla de Símbolos

(1) Tasa	real
(2) Base	real
(3) Recargo	real
(4) 100.0	real

Los operandos son referencias a la Tabla de Símbolos, o a otros tercetos.



Representaciones: Cuartetos

Es una representación constituida por códigos de 4 direcciones
(operador, operando 1, operando 2, resultado)

Tasa := Base + Recargo / 100.0 → ID := ID + ID / CTE

14. ...

15. (/, Recargo, 100.0, aux1)

16. (+, Base, aux1, aux2)

17. (:=, Tasa, aux2, -)

18. ...

Tabla de Símbolos

(1) Tasa	real
(2) Base	real
(3) Recargo	real
(4) 100.0	real

Los operandos son referencias a la Tabla de Símbolos

Representaciones: Polaca Inversa

Para esta representación:

Operadores binarios: $op1 \text{ operador } op2 \rightarrow op1 \ op2 \ \text{operador}$

Operadores unarios: $\text{operador } op1 \rightarrow op1 \ \text{operador}$

Tasa := Base + Recargo / 100.0 \rightarrow ID := ID + ID / CTE

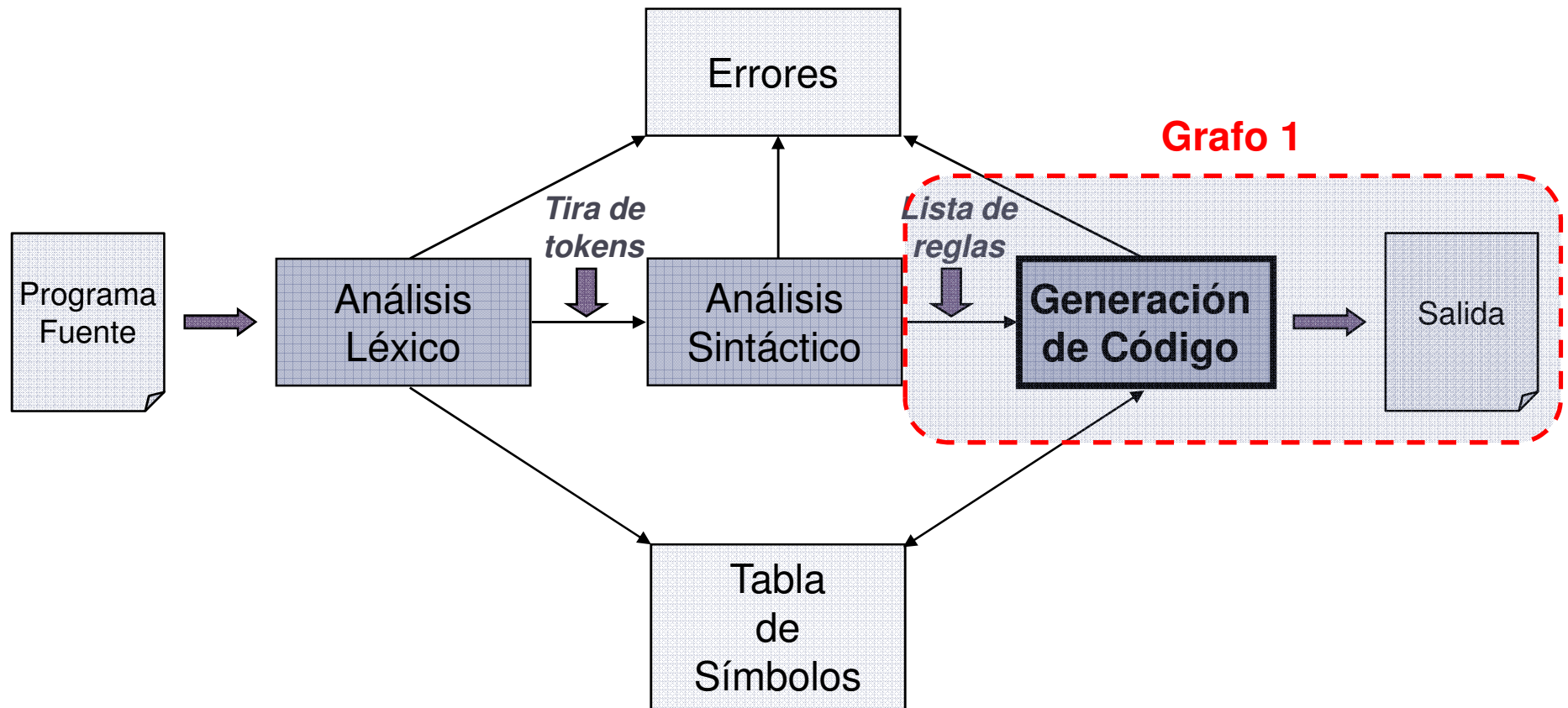
...	Base	Recargo	100.0	/	+	Tasa	:=	...
21	22	23	24	25	26	27	28	29

Los operandos son referencias a
la Tabla de Símbolos

Tabla de Símbolos

(1) Tasa	real
(2) Base	real
(3) Recargo	real
(4) 100.0	real

Fases de la Compilación



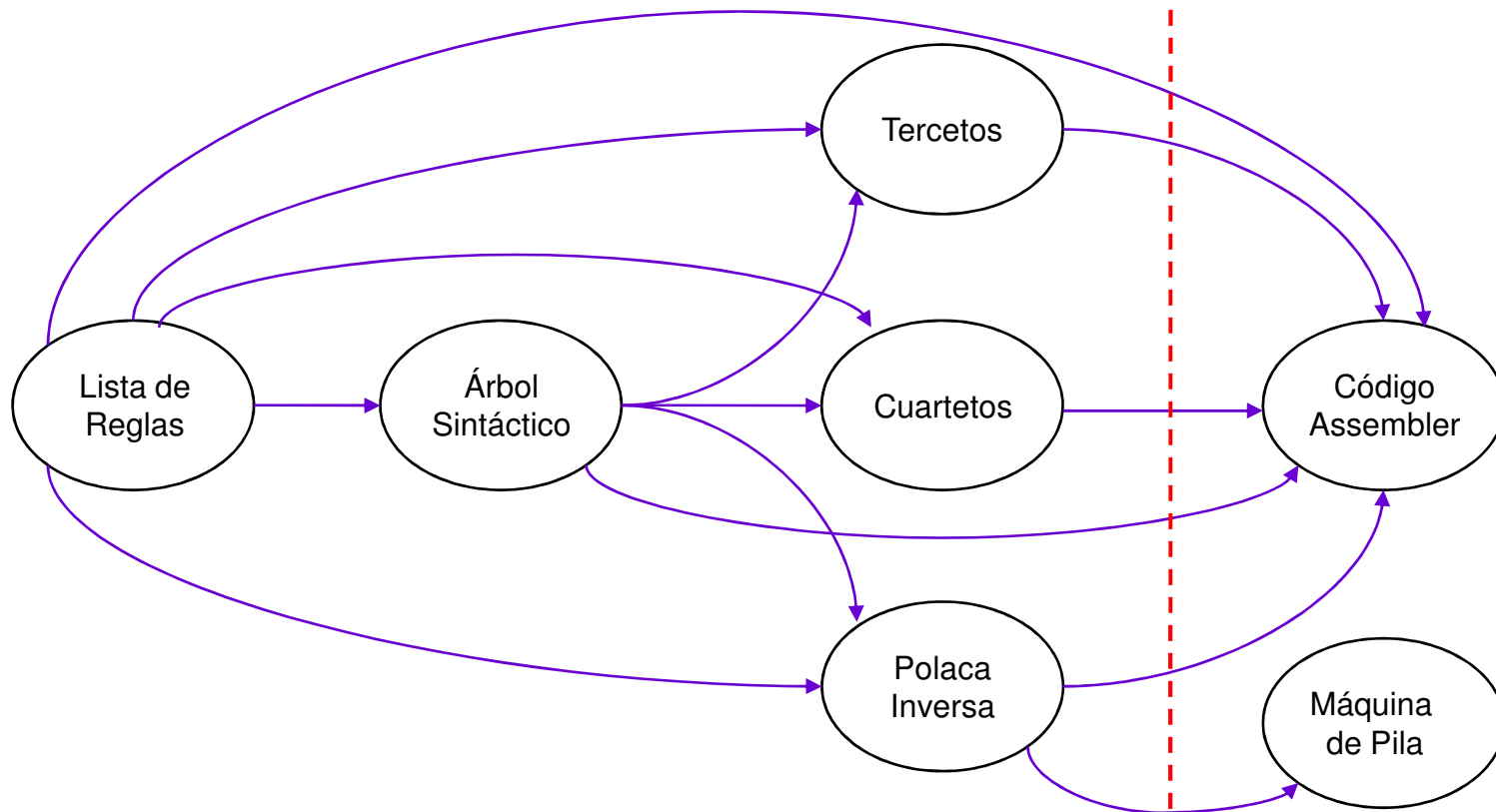
Generación de Código

Grafo 1

Sentencias ejecutables
 $x := x + 2;$

Sólo depende
del lenguaje

Depende del S.O.
y de la máquina



Generación de Código

Representaciones: Código Assembler

Esta representación depende del Sistema Operativo, y la máquina para la cual se genera.

Tasa := Base + Recargo / 100.0 → ID := ID + ID / CTE

...

FLD Base

FLD Recargo

FDIV 100.0

FADD

FST Tasa

...

(Ejemplo de código Assembler utilizando el coprocesador matemático de Pentium)

Máquina de Pila

- ▶ No se traduce → Se interpreta

- ▶ Técnica en desuso.



- ▶ Se retoma:

- ▶ Smalltalk (se interpreta mucho)

- ▶ Java (se compila más y luego se interpreta)

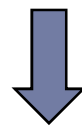


Comparación de caminos

Compilador rápido (el más rápido)



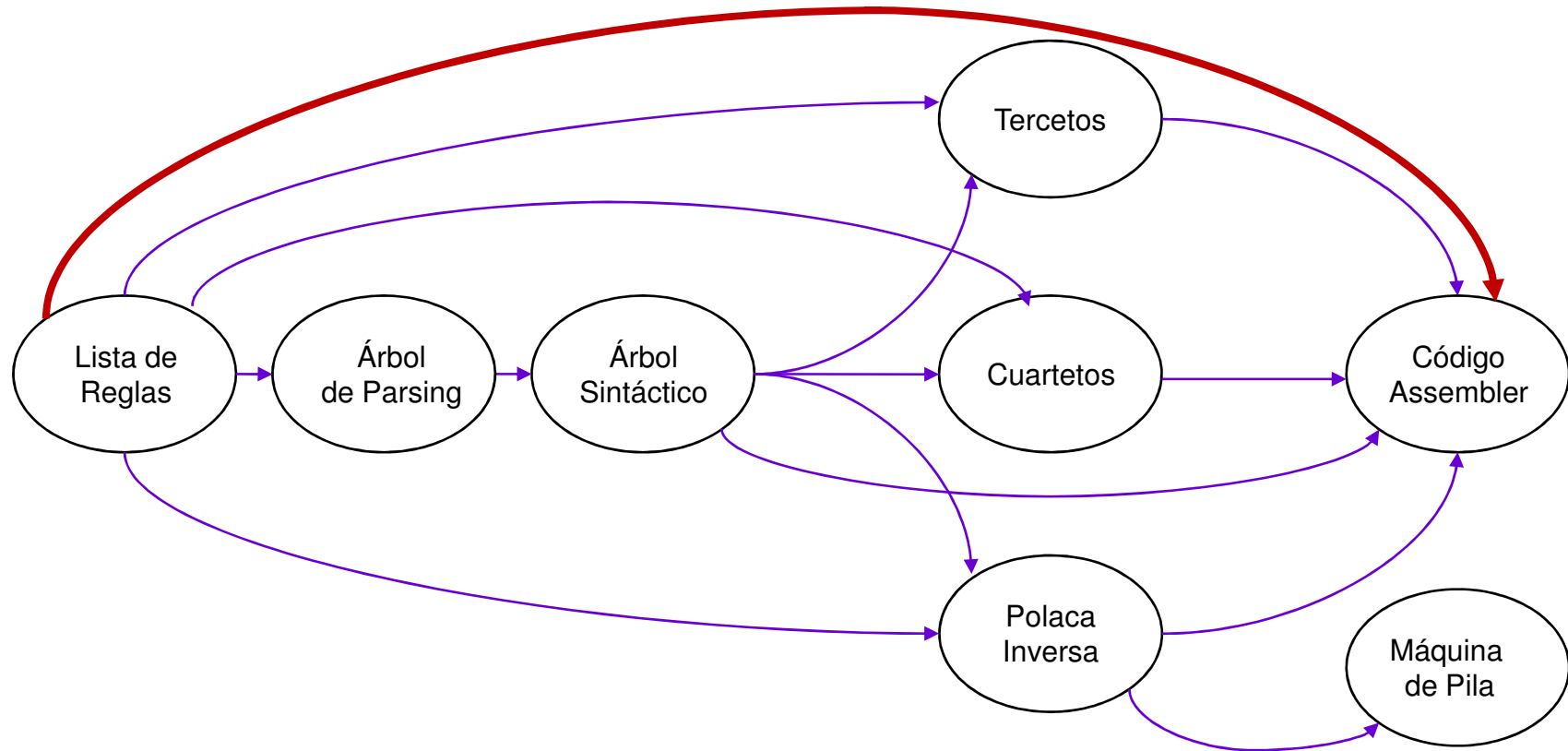
Ejecutable grande y lento



Lista de Reglas → Assembler



Generación de Código



► Generación de Código

Comparación de caminos

Ejecutable rápido



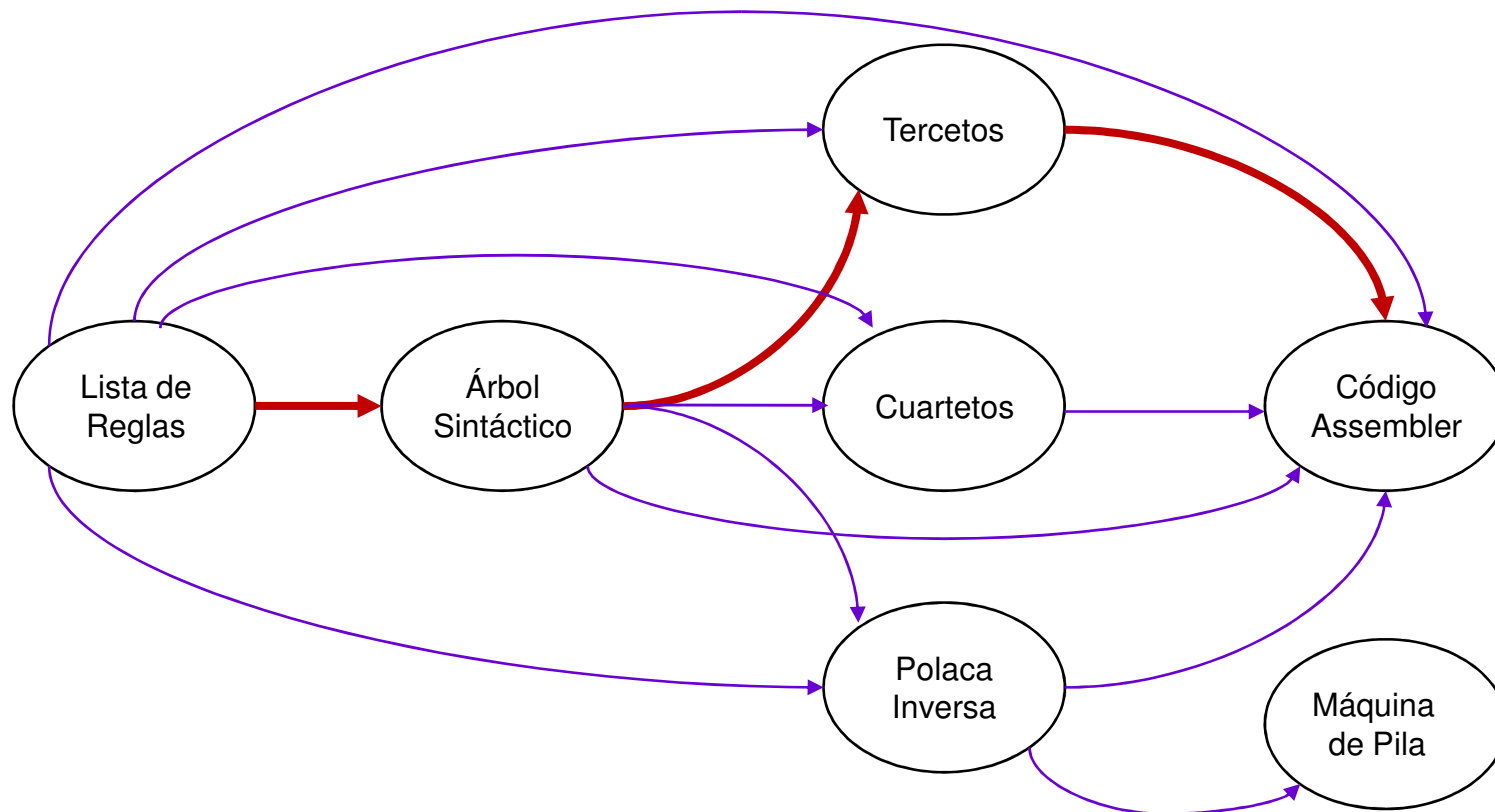
Compilador lento



L.Reglas → Á.Sintáctico → Tercetos → Assembler
(Optimizaciones)



Generación de Código



Comparación de caminos

Desarrollo Rápido



Compilador relativamente rápido



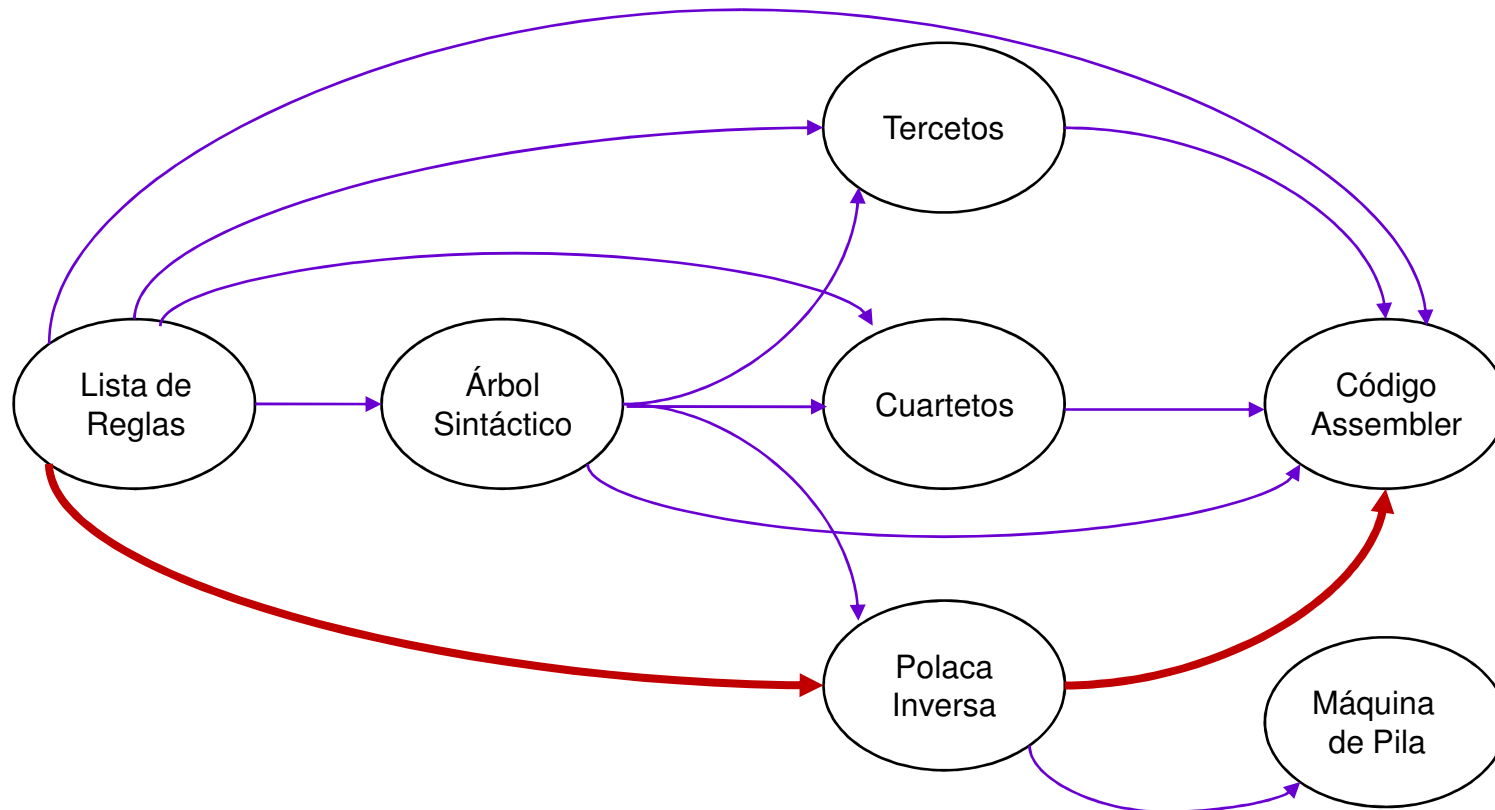
Ejecutable relativamente bueno



Lista de Reglas → Polaca Inversa → Assembler



Generación de Código



Comparación de caminos

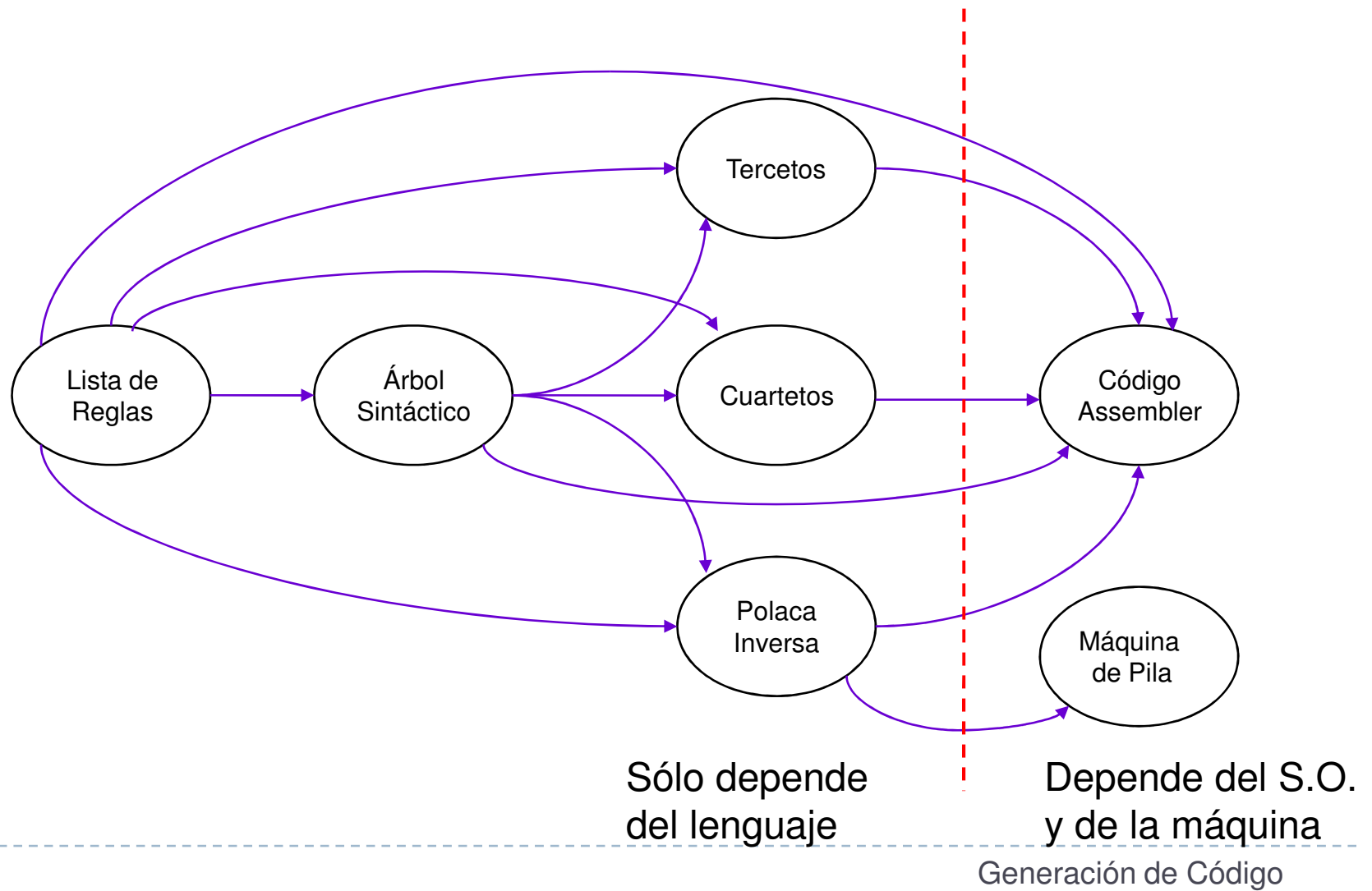
Portabilidad



Representaciones Intermedias

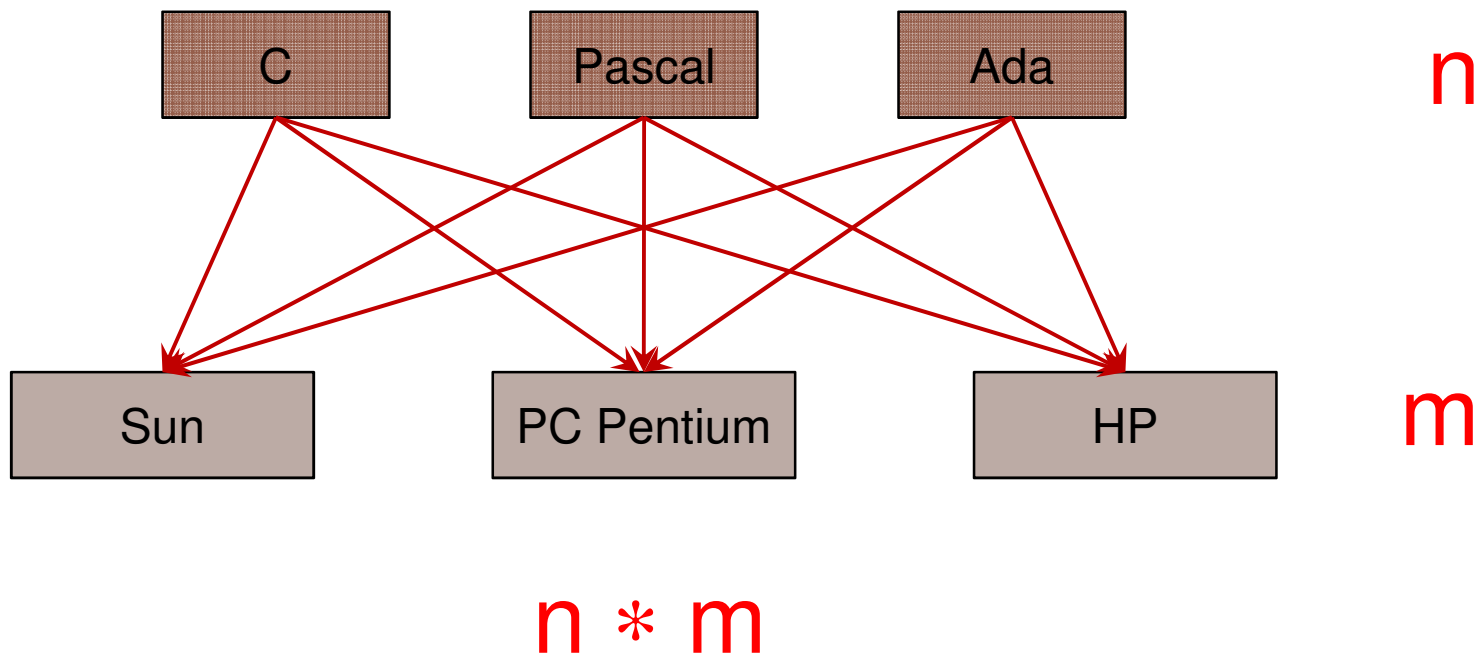


Portabilidad



Portabilidad

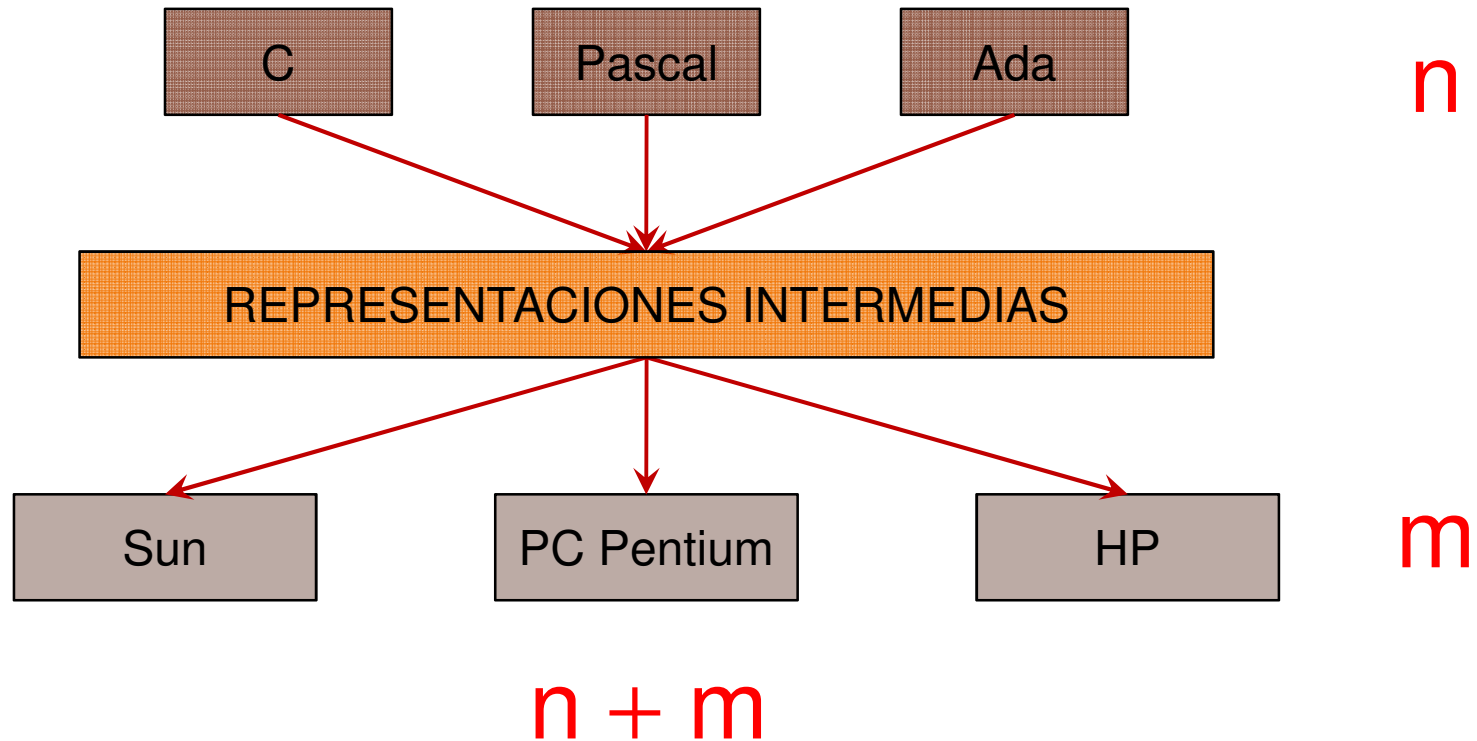
Lista de Reglas \rightarrow Assembler



Generación de Código

Portabilidad

L.Reglas → Representaciones Intermedias → Assembler



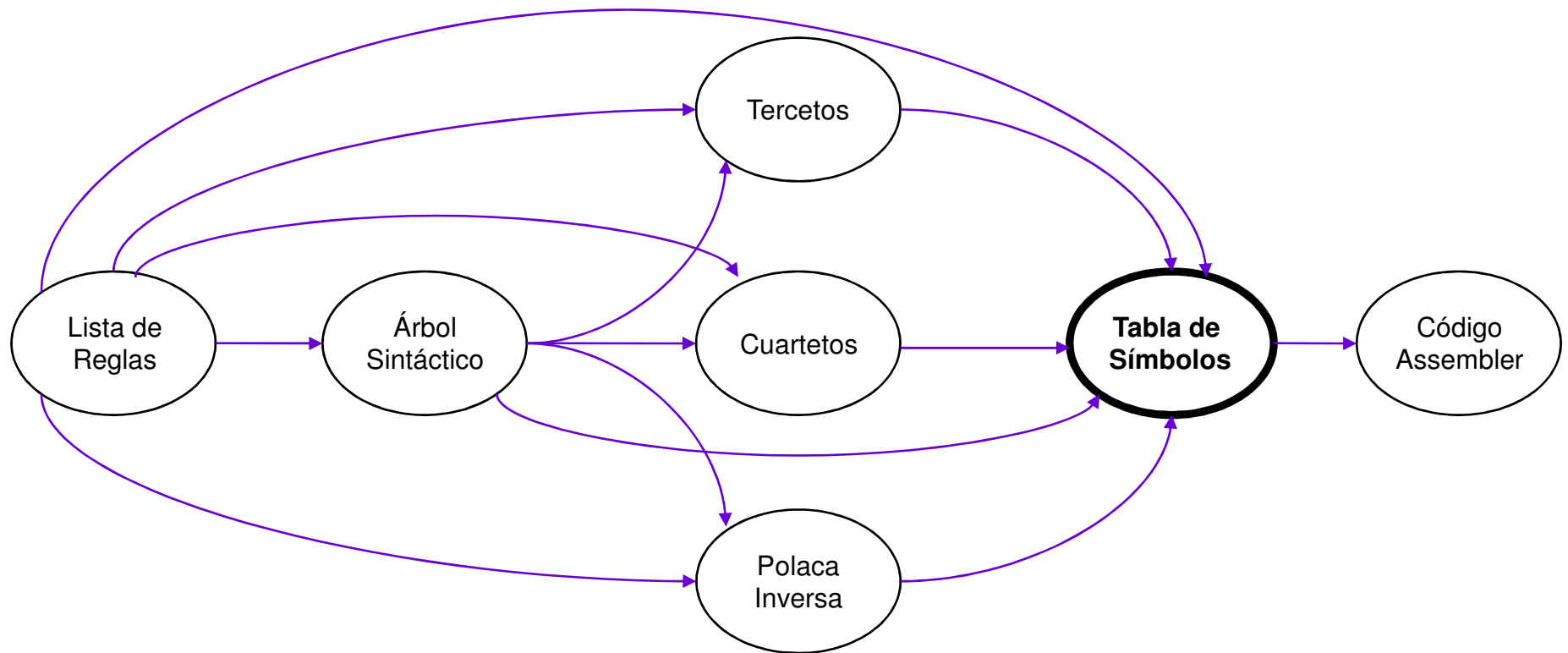
Generación de Código

Generación de Código

Grafo 2: Sentencias declarativas

Sentencias declarativas

int x, y;



► Generación de Código

Comparación de caminos

Grafo 2: Sentencias declarativas

Mejor camino



L. Reglas → Tabla de Símbolos → Assembler
(todo depende del lenguaje)

- ▶ Utilizar representaciones intermedias para sentencias declarativas:
 - ▶ No mejora la portabilidad
 - ▶ Compilador más lento

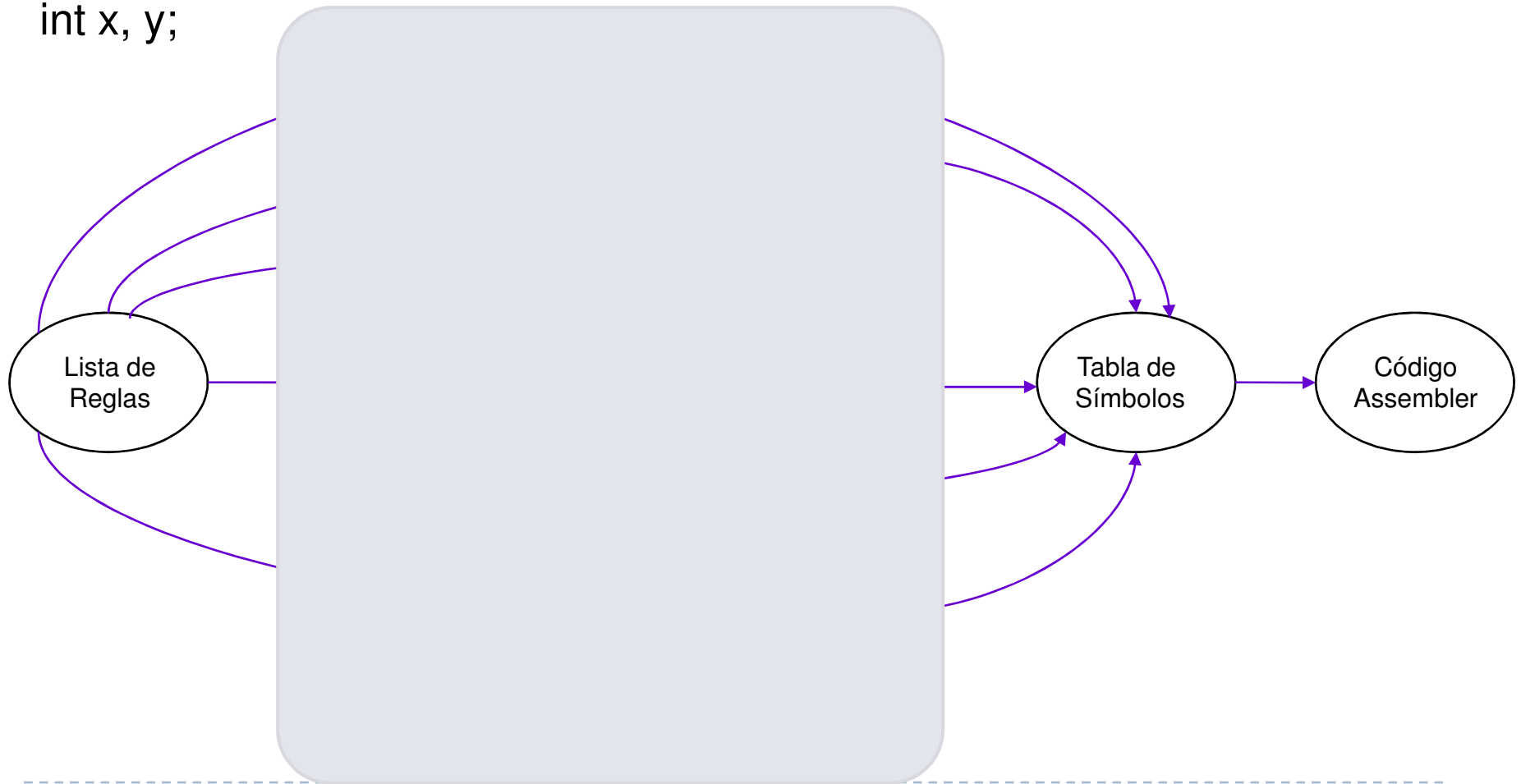


Generación de Código

Grafo 2: Sentencias declarativas

Sentencias declarativas

`int x, y;`



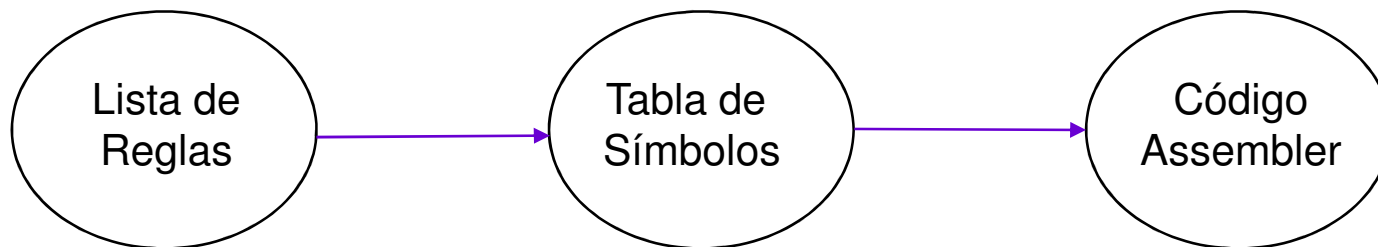
Generación de Código

Generación de Código

Grafo 2: Sentencias declarativas

Sentencias declarativas

```
int x, y;
```

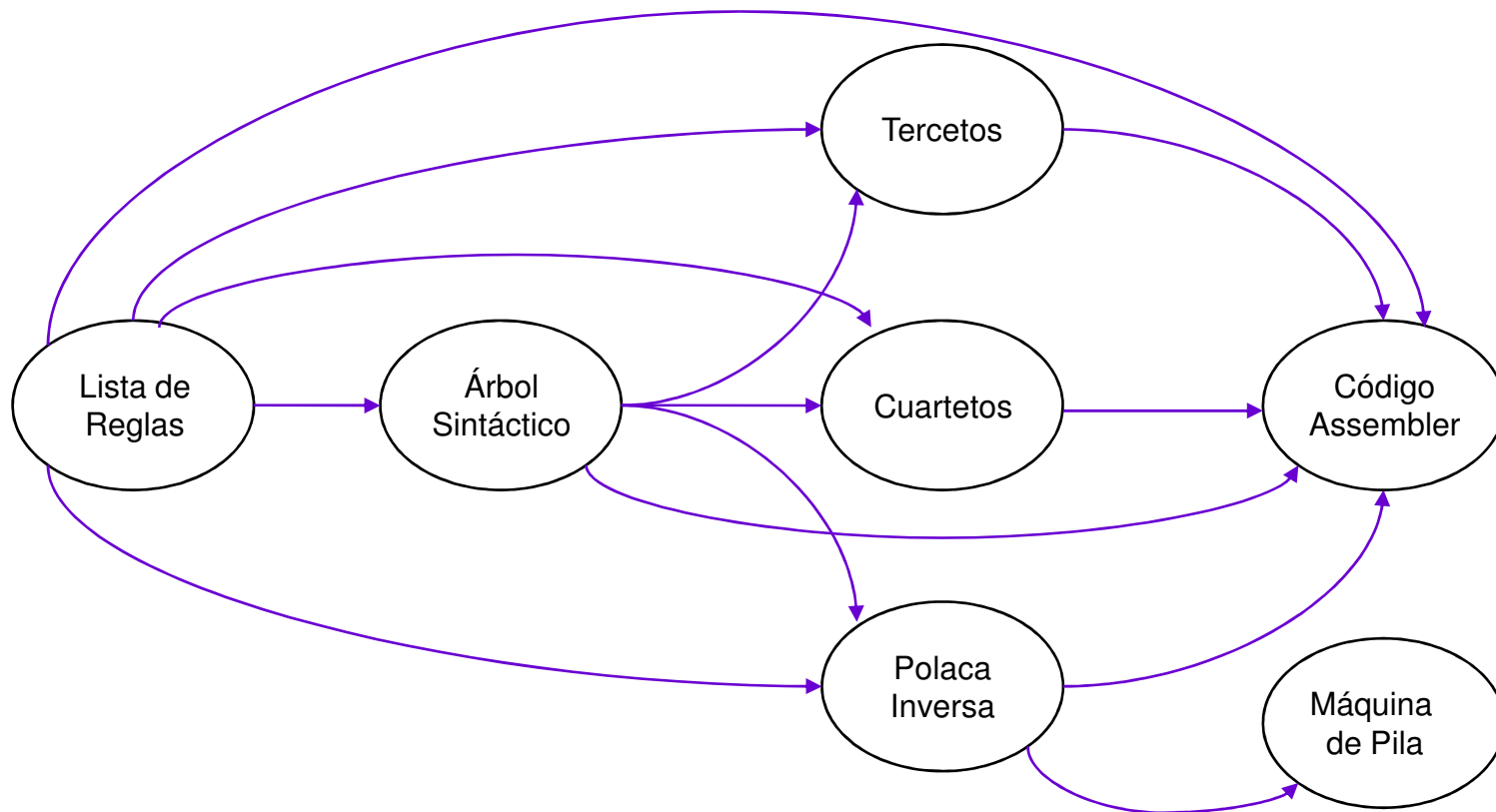


Generación de Código

Grafo 1: Sentencias Ejecutables

Sentencias ejecutables

$x := x + 2;$



► Generación de Código

Generación de código Intermedio

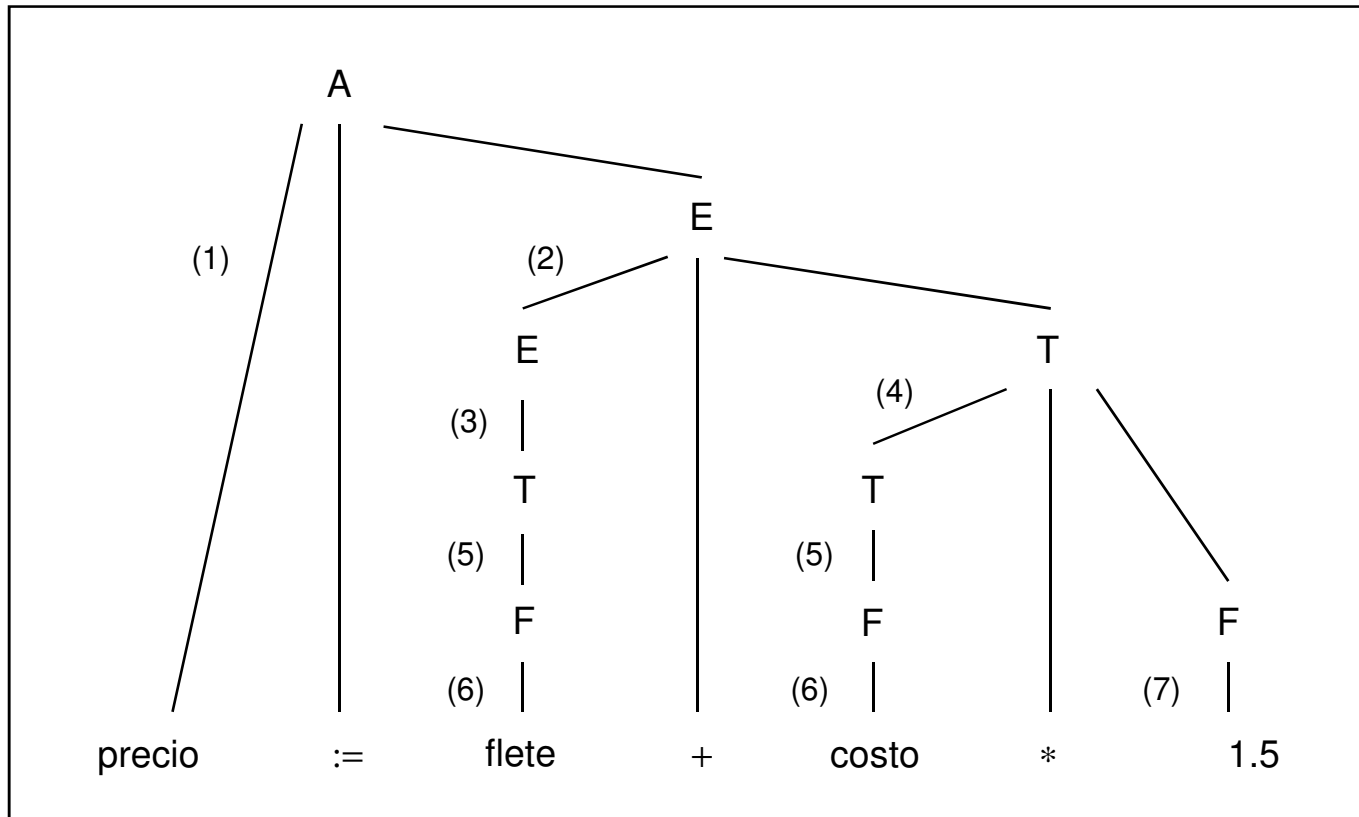
Lista de Reglas → Árbol Sintáctico
Lista de Reglas → Tercetos
Lista de Reglas → Polaca Inversa

Expresiones / Asignaciones

LISTA DE REGLAS → Árbol Sintáctico

Árbol de Parsing

precio := flete + costo * 1.5 → id := id + id * cte



- 1) $A \rightarrow id := E$
- 2) $E \rightarrow E + T$
- 3) $E \rightarrow T$
- 4) $T \rightarrow T * F$
- 5) $T \rightarrow F$
- 6) $F \rightarrow id$
- 7) $F \rightarrow cte$

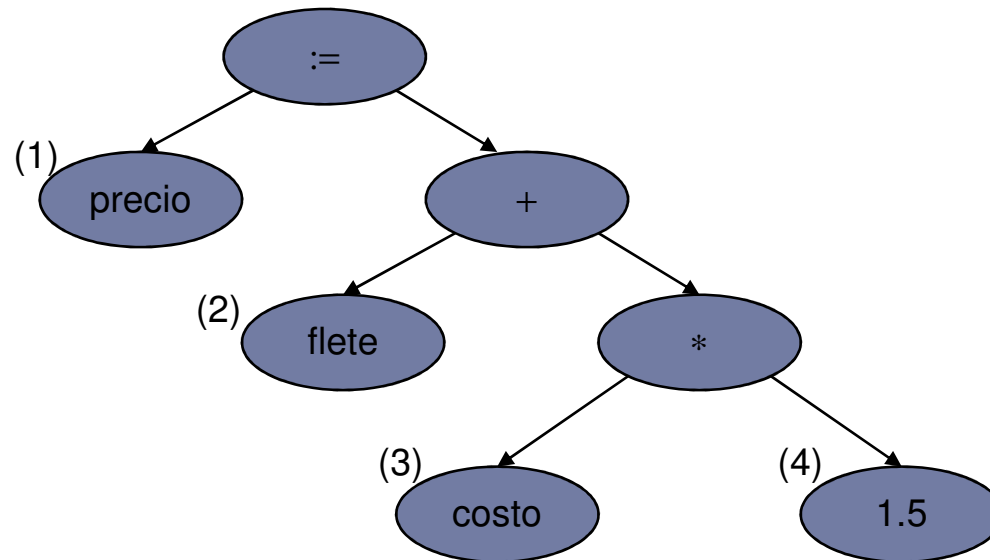
Lista de reglas: 6 5 3 6 5 7 4 2 1



Generación de Código

Árbol Sintáctico

precio := flete + costo * 1.5



Nota: Los nodos (1) (2) (3) y (4) contienen referencias a la Tabla de Símbolos



Lista de Reglas \rightarrow Árbol Sintáctico

Reglas de la Gramática	Acciones Semánticas
1. $A \rightarrow id := E$	$A.ptr = crear_nodo(':='; crear_hoja(id.ptr); E.ptr)$ (*)
2. $E \rightarrow E + T$	$E.ptr = crear_nodo('+; E.ptr; T.ptr)$
3. $E \rightarrow T$	$E.ptr = T.ptr;$
4. $T \rightarrow T * F$	$T.ptr = crear_nodo('*; T.ptr; F.ptr)$
5. $T \rightarrow F$	$T.ptr = F.ptr;$
6. $F \rightarrow id$	$F.ptr = crear_hoja(id.ptr)$ (*)
7. $F \rightarrow cte$	$F.ptr = crear_hoja(cte.ptr)$ (**)

(*) $id.ptr$ = Ref a la entrada del identificador en la Tabla de Símbolos

(**) $cte.ptr$ = Ref a la entrada de la constante en la Tabla de Símbolos



Lista de Reglas \rightarrow Árbol Sintáctico

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

Regla		Acción Semántica		
6. $F \rightarrow id$		F.ptr = crear_hoja(id.ptr) (*)		

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete

A diagram showing a non-terminal symbol 'F' with an arrow pointing to a terminal node 'flete'. The terminal node is represented by a blue oval.



Lista de Reglas \rightarrow Árbol Sintáctico

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

Regla		Acción Semántica		
5. $T \rightarrow F$		T.ptr = F.ptr;		

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete

A diagram showing a node labeled 'flete' in a blue oval. Two arrows point to this node: one from the left labeled 'F' and one from the top labeled 'T'.



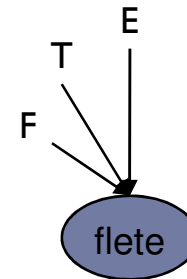
Lista de Reglas \rightarrow Árbol Sintáctico

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

Regla	Acción Semántica
3. $E \rightarrow T$	$E.ptr = T.ptr;$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete



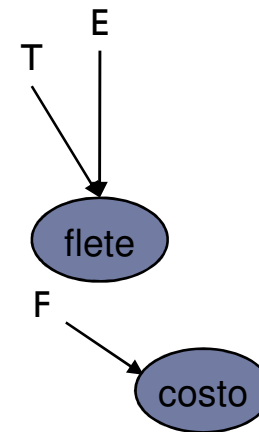
Lista de Reglas → Árbol Sintáctico

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

Regla	Acción Semántica
6. $F \rightarrow id$	$F.ptr = crear_hoja(id.ptr) (*)$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo



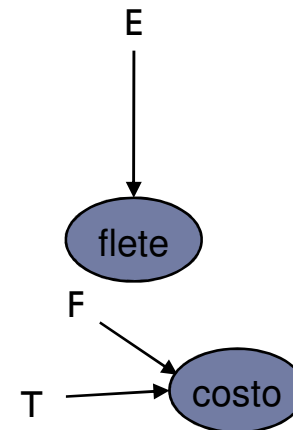
Lista de Reglas \rightarrow Árbol Sintáctico

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

Regla	Acción Semántica
5. $T \rightarrow F$	$T.ptr = F.ptr;$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo
5	-	^flete	^costo	^costo



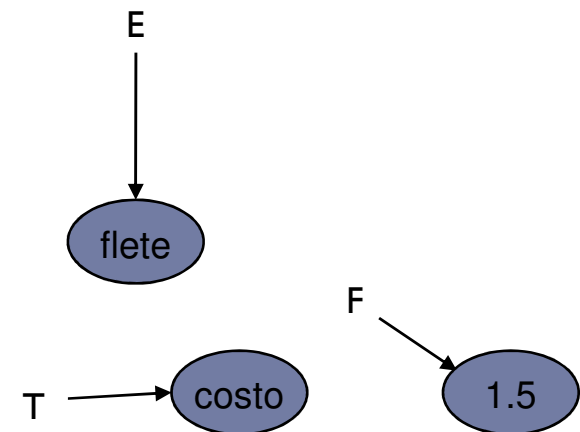
Lista de Reglas → Árbol Sintáctico

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

Regla	Acción Semántica
7. $F \rightarrow cte$	$F.ptr = crear_hoja(cte.ptr) (**)$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo
5	-	^flete	^costo	^costo
7	-	^flete	^costo	^1.5



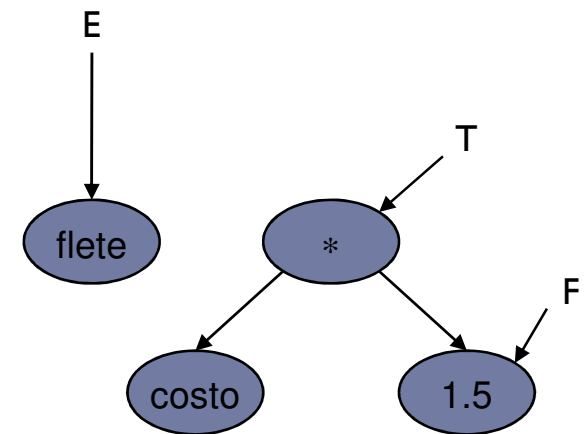
Lista de Reglas \rightarrow Árbol Sintáctico

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

Regla	Acción Semántica
4. $T \rightarrow T * F$	$T.ptr = \text{crear_nodo}(' * ' ; T.ptr ; F.ptr)$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo
5	-	^flete	^costo	^costo
7	-	^flete	^costo	^1.5
4	-	^flete	^*	^1.5



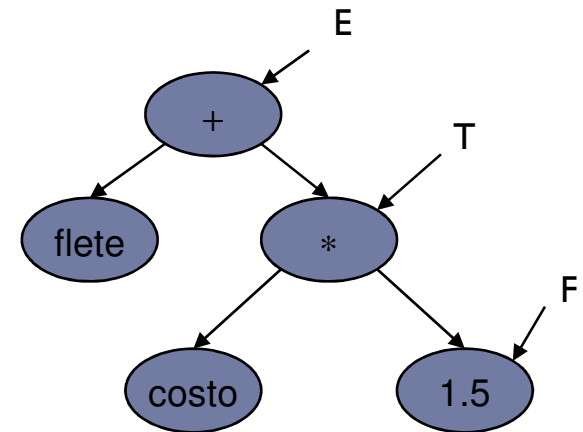
Lista de Reglas \rightarrow Árbol Sintáctico

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

Regla	Acción Semántica
2. $E \rightarrow E + T$	$E.ptr = \text{crear_nodo}('+' ; E.ptr ; T.ptr)$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo
5	-	^flete	^costo	^costo
7	-	^flete	^costo	^1.5
4	-	^flete	^*	^1.5
2	-	^+	^*	^1.5



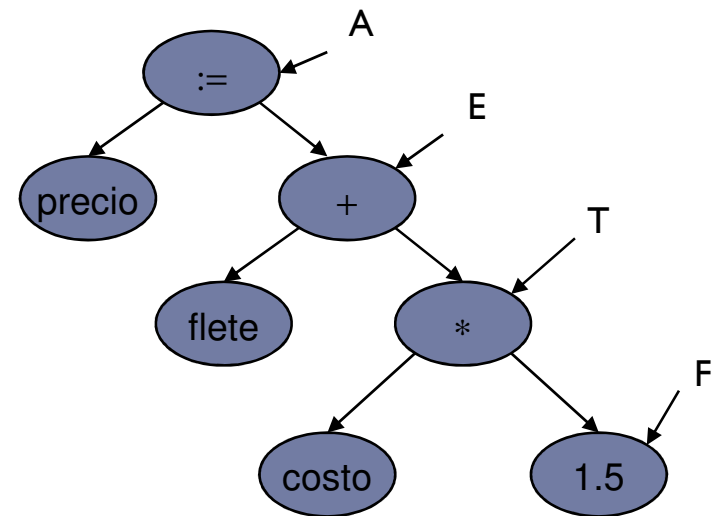
Lista de Reglas \rightarrow Árbol Sintáctico

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

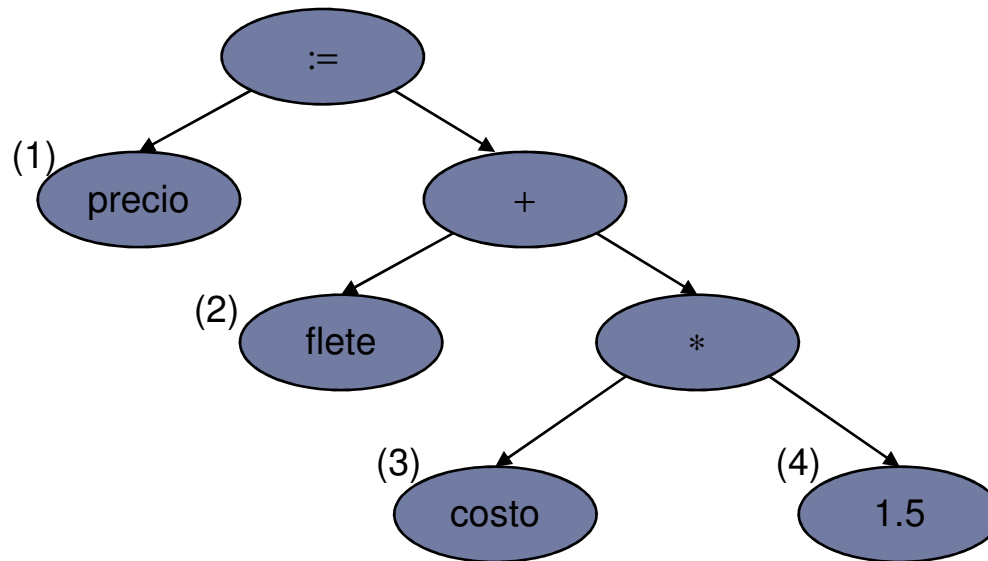
Regla	Acción Semántica
1. $A \rightarrow id := E$	$A.ptr = \text{crear_nodo}(':='; \text{crear_hoja}(id.ptr); E.ptr) (*)$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo
5	-	^flete	^costo	^costo
7	-	^flete	^costo	^1.5
4	-	^flete	^*	^1.5
2	-	^+	^*	^1.5
1	^:=	^+	^*	^1.5



Lista de Reglas \rightarrow Árbol Sintáctico

precio := flete + costo * 1.5



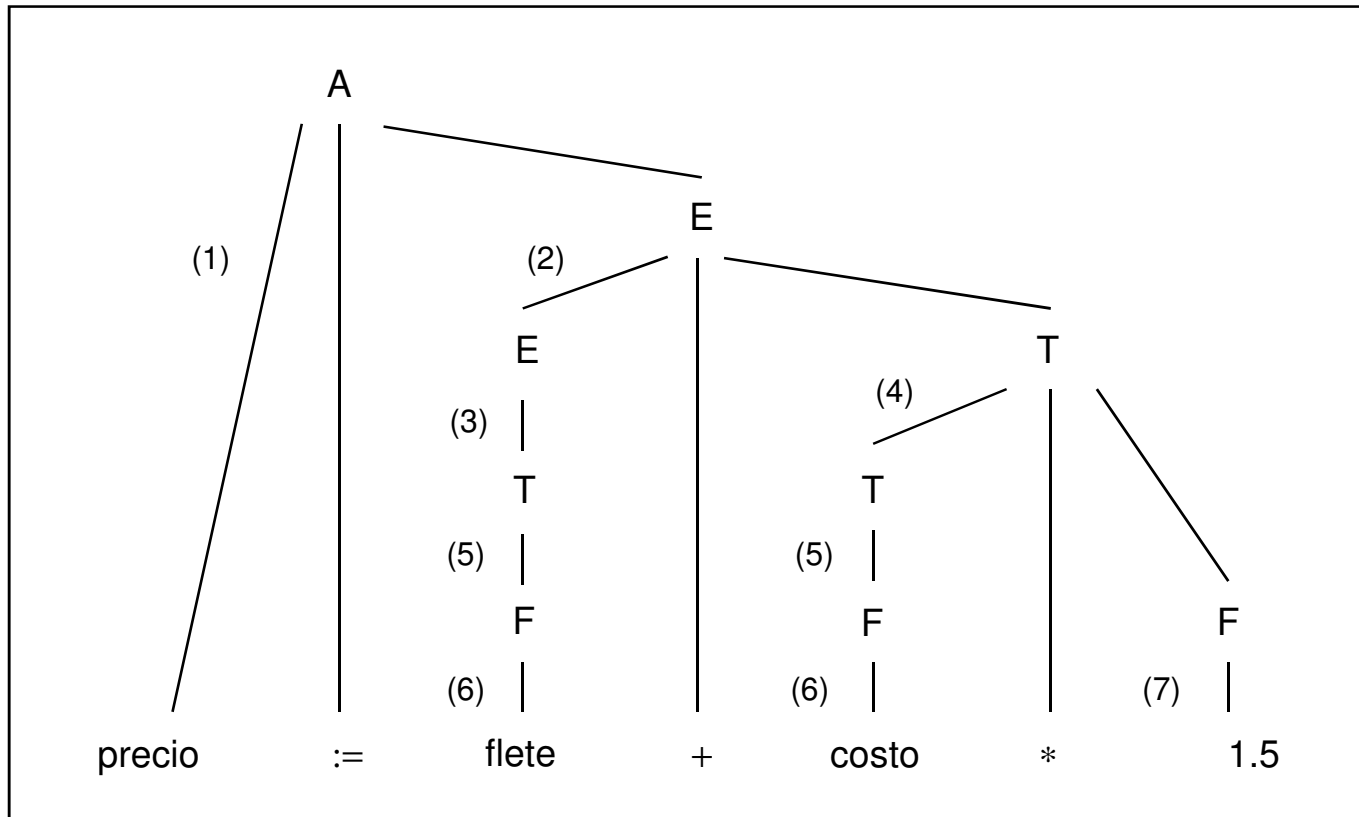
Nota: Los nodos (1) (2) (3) y (4) contienen referencias a la Tabla de Símbolos



LISTA DE REGLAS → Tercetos

Árbol de Parsing

precio := flete + costo * 1.5 → id := id + id * cte



- 1) $A \rightarrow id := E$
- 2) $E \rightarrow E + T$
- 3) $E \rightarrow T$
- 4) $T \rightarrow T * F$
- 5) $T \rightarrow F$
- 6) $F \rightarrow id$
- 7) $F \rightarrow cte$

Lista de reglas: 6 5 3 6 5 7 4 2 1



Generación de Código

Tercetos

precio := flete + costo * 1.5

14. ...
15. (* , costo , 1.5)
16. (+ , flete , [15])
17. (:= , precio , [16])
18. ...

Los operandos *costo*, *flete*, *precio* y *1.5* son referencias a la Tabla de Símbolos
Los operandos [15] y [16] son referencias a los correspondientes tercetos

Lista de Reglas \rightarrow Tercetos

Reglas de la Gramática	Acciones Semánticas
1. $A \rightarrow id := E$	$A.ptr = crear_terceto(':='; id.ptr ; E.ptr) (*)$
2. $E \rightarrow E + T$	$E.ptr = crear_terceto('+'; E.ptr ; T.ptr)$
3. $E \rightarrow T$	$E.ptr = T.ptr;$
4. $T \rightarrow T * F$	$T.ptr = crear_terceto('*'; T.ptr ; F.ptr)$
5. $T \rightarrow F$	$T.ptr = F.ptr;$
6. $F \rightarrow id$	$F.ptr = id.ptr (*)$
7. $F \rightarrow cte$	$F.ptr = cte.ptr (**)$

(*) $id.ptr$ = Ref a la entrada del identificador en la Tabla de Símbolos

(**) $cte.ptr$ = Ref a la entrada de la constante en la Tabla de Símbolos



Lista de Reglas \rightarrow Tercetos

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6** 5 3 6 5 7 4 2 1

Regla	Acción Semántica
6. $F \rightarrow id$	$F.ptr = id.ptr \quad (*)$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete



Lista de Reglas \rightarrow Tercetos

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
5. $T \rightarrow F$	$T.ptr = F.ptr;$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete



Lista de Reglas \rightarrow Tercetos

Ejemplo: `precio := flete + costo * 1.5`

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
3. $E \rightarrow T$	$E.ptr = T.ptr;$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete



Lista de Reglas \rightarrow Tercetos

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
6. $F \rightarrow id$	$F.ptr = id.ptr$ (*)

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo



Lista de Reglas \rightarrow Tercetos

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
5. $T \rightarrow F$	$T.ptr = F.ptr;$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo
5	-	^flete	^costo	^costo



Lista de Reglas \rightarrow Tercetos

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
7. $F \rightarrow \text{cte}$	$F.\text{ptr} = \text{cte}.\text{ptr} (**)$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo
5	-	^flete	^costo	^costo
7	-	^flete	^costo	^1.5



Lista de Reglas \rightarrow Tercetos

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

Regla	Acción Semántica
4. $T \rightarrow T * F$	T.ptr = crear_terceto('*' ; T.ptr ; F.ptr)

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo
5	-	^flete	^costo	^costo
7	-	^flete	^costo	^1.5
4	-	^flete	^[15]	^1.5

```

14. ...
15. ( * , costo , 1.5 )
    
```



Lista de Reglas \rightarrow Tercetos

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
2. $E \rightarrow E + T$	$E.ptr = \text{crear_terceto}('+' ; E.ptr ; T.ptr)$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	$\wedge \text{flete}$
5	-	-	$\wedge \text{flete}$	$\wedge \text{flete}$
3	-	$\wedge \text{flete}$	$\wedge \text{flete}$	$\wedge \text{flete}$
6	-	$\wedge \text{flete}$	$\wedge \text{flete}$	$\wedge \text{costo}$
5	-	$\wedge \text{flete}$	$\wedge \text{costo}$	$\wedge \text{costo}$
7	-	$\wedge \text{flete}$	$\wedge \text{costo}$	$\wedge 1.5$
4	-	$\wedge \text{flete}$	$\wedge [15]$	$\wedge 1.5$
2	-	$\wedge [16]$	$\wedge [15]$	$\wedge 1.5$

```

14. ...
15. ( * , costo , 1.5 )
16. ( + , flete , [15] )
    
```

F



Lista de Reglas \rightarrow Tercetos

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
I. $A \rightarrow id := E$	$A.ptr = \text{crear_terceto}(':='; id.ptr ; E.ptr) (*)$

Regla	A.ptr	E.ptr	T.ptr	F.ptr
6	-	-	-	^flete
5	-	-	^flete	^flete
3	-	^flete	^flete	^flete
6	-	^flete	^flete	^costo
5	-	^flete	^costo	^costo
7	-	^flete	^costo	^1.5
4	-	^flete	^[15]	^1.5
2	-	^[16]	^[15]	^1.5
1	^[17]	^[16]	^[15]	^1.5

```

14. ...
15. ( * , costo , 1.5 )
16. ( + , flete , [15] )
17. ( := , precio , [16] )
18. ...
    
```



Lista de Reglas \rightarrow Tercetos

precio := flete + costo * 1.5

14. ...
15. (* , costo , 1.5)
16. (+ , flete , [15])
17. (:= , precio , [16])
18. ...

Los operandos *costo*, *flete*, *precio* y *1.5* son referencias a la Tabla de Símbolos

Lista de Reglas \rightarrow Tercetos

$z := a * b * 2.5 + c * 5 + d ;$

- 20. ...
- 21. (* , a , b)
- 22. (* , [21] , 2.5)
- 23. (* , c , 5)
- 24. (+ , [22] , [23])
- 25. (+ , [24] , d)
- 26. (:= , z , [25])
- 27. ...

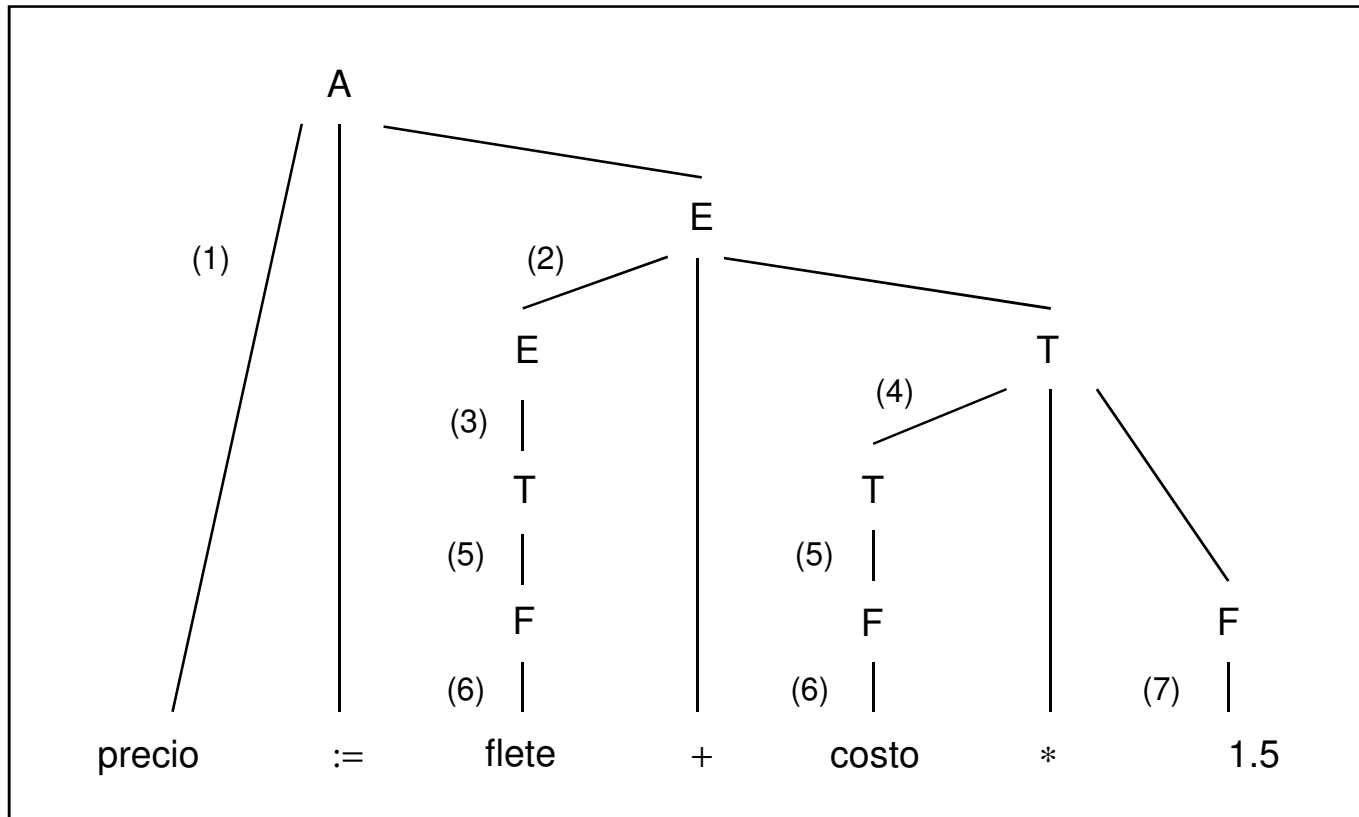
Los operandos $a, b, c, d, z, 2.5$ y 5 son referencias a la Tabla de Símbolos



LISTA DE REGLAS → Polaca Inversa

Árbol de Parsing

precio := flete + costo * 1.5 → id := id + id * cte



- 1) $A \rightarrow id := E$
- 2) $E \rightarrow E + T$
- 3) $E \rightarrow T$
- 4) $T \rightarrow T * F$
- 5) $T \rightarrow F$
- 6) $F \rightarrow id$
- 7) $F \rightarrow cte$

Lista de reglas: 6 5 3 6 5 7 4 2 1



Generación de Código

Polaca Inversa

precio := flete + costo * 1.5

...	flete	costo	1.5	*	+	precio	:=	...
21	22	23	24	25	26	27	28	29

Los operandos *costo*, *flete*, *precio* y *1.5* son referencias a la Tabla de Símbolos



Lista de Reglas \rightarrow Polaca Inversa

Reglas de la Gramática	Acciones Semánticas
1. $A \rightarrow id := E$	Agregar(id.ptr);Agregar(':= ') (*)
2. $E \rightarrow E + T$	Agregar('+')
3. $E \rightarrow T$	-
4. $T \rightarrow T * F$	Agregar('*')
5. $T \rightarrow F$	-
6. $F \rightarrow id$	Agregar(id.ptr) (*)
7. $F \rightarrow cte$	Agregar(cte.ptr) (**)

(*) id.ptr = Ref a la entrada del identificador en la Tabla de Símbolos

(**) cte.ptr = Ref a la entrada de la constante en la Tabla de Símbolos



Lista de Reglas \rightarrow Polaca Inversa

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: 6 5 3 6 5 7 4 2 1

Regla	Acción Semántica
6. $F \rightarrow id$	Agregar(id.ptr) (*)

...	flete
21	22



Lista de Reglas \rightarrow Polaca Inversa

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
5. $T \rightarrow F$	-

...	flete
21	22



Lista de Reglas \rightarrow Polaca Inversa

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
3. $E \rightarrow T$	-

...	flete
21	22



Lista de Reglas \rightarrow Polaca Inversa

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
6. $F \rightarrow id$	Agregar(id.ptr) (*)

...	flete	costo
21	22	23



Lista de Reglas \rightarrow Polaca Inversa

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
5. $T \rightarrow F$	-

...	flete	costo
21	22	23



Lista de Reglas \rightarrow Polaca Inversa

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
7. $F \rightarrow \text{cte}$	Agregar(cte.ptr) (**)

...	flete	costo	1.5
21	22	23	24



Lista de Reglas \rightarrow Polaca Inversa

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
4. $T \rightarrow T * F$	Agregar('*')

...	flete	costo	1.5	*
21	22	23	24	25



Lista de Reglas \rightarrow Polaca Inversa

Ejemplo: $\text{precio} := \text{flete} + \text{costo} * 1.5$

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
2. $E \rightarrow E + T$	Agregar('+')

...	flete	costo	1.5	*	+
21	22	23	24	25	26



Lista de Reglas \rightarrow Polaca Inversa

Ejemplo: precio := flete + costo * 1.5

Lista de reglas: **6 5 3 6 5 7 4 2 1**

Regla	Acción Semántica
I. $A \rightarrow id := E$	Agregar(id.ptr);Agregar(':= ') (*)

...	flete	costo	1.5	*	+	precio	:=
21	22	23	24	25	26	27	28



Lista de Reglas → Polaca Inversa

precio := flete + costo * 1.5

...	flete	costo	1.5	*	+	precio	:=	...
21	22	23	24	25	26	27	28	29

Los operandos *costo*, *flete*, *precio* y *1.5* son referencias a la Tabla de Símbolos



Lista de Reglas → Polaca Inversa

$z := a * b * 2.5 + c * 5 + d ;$

...	a	b	*	2.5	*	c	5	*	+	d	+	z	:=	...
11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Los operandos a , b , c , d , z , 2.5 y 5 son referencias a la Tabla de Símbolos

