

## Introducción a la compilación

### Tipos de traducción:

Traducir un programa fuente en un idioma ejecutable en otro idioma y producir los resultados de ese programa.

TRAN

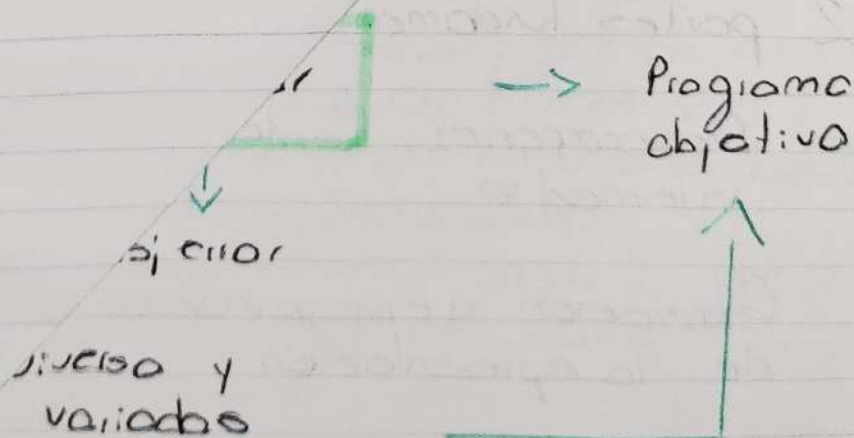
Programa fuente y producir los resultados de ese programa.

Unidad

# 1

(M)

compiladores traducen (fuente) a otro (destino).



## Introducción a la compilación

### Mecanismos de traducción:

- **Compilación:** Traducir un programa fuente en un idioma o un programa ejecutable en otro idioma y producir resultados mientras se ejecuta el nuevo programa.

Ej: C, C++, FORTRAN

- **Interpretación:** Leer un programa fuente y producir los resultados mientras se entiende ese programa.

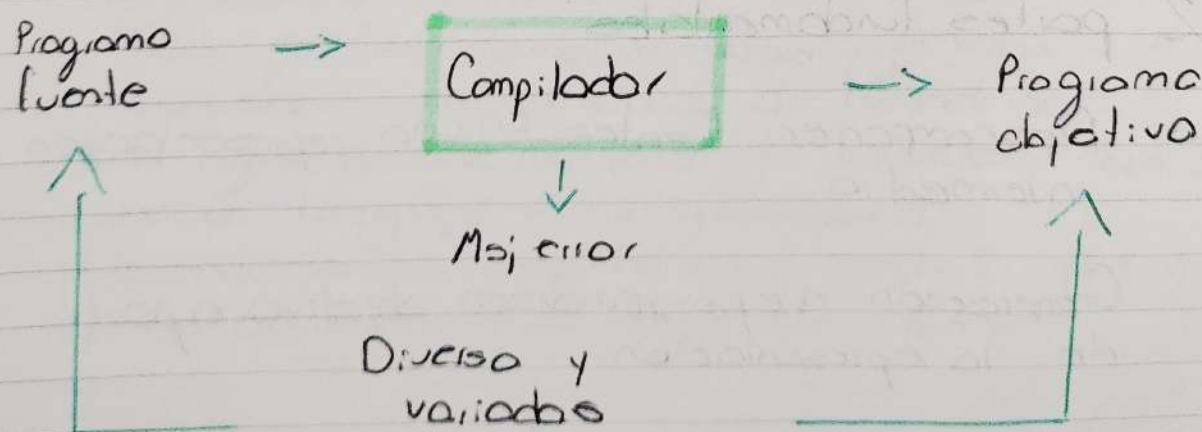
Ej: BASIC, LISP

- **Case Study: JAVA**

1ero: Traducir a bytecode de java

2do: ejecutar por interpretación (JVM)

**Propósito del compilador:** Los compiladores traducen un programa escrito en un lenguaje (fuente) a otro (destino)

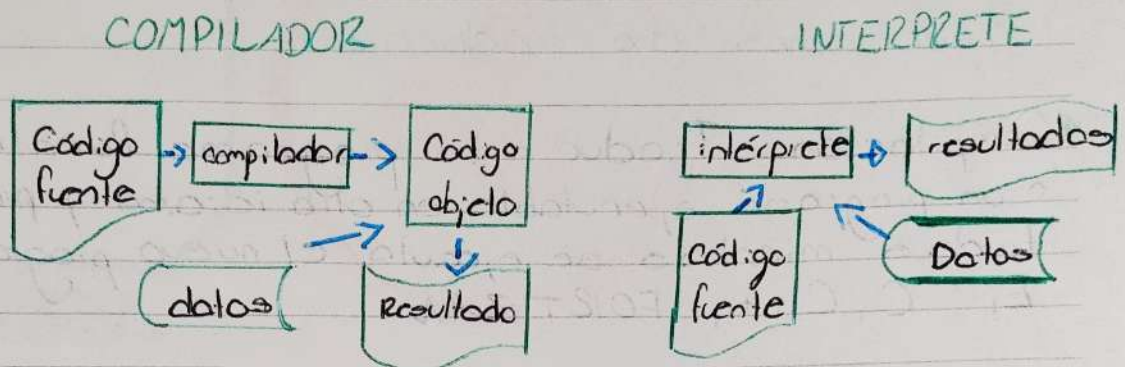




Gonzalez Morales Victor

## Comparación de compilador / intérprete.

### Visión General



### Ventajas

Ejecución rápida del programa; explota las características de la arquitectura.

Fácil de depurar; flexible para modificar; independiente de la máquina.

### Desventajas

Pre-procesamiento del programa; Complejidad.

Sobrecarga en ejecución; Sobrecarga de espacio.

## EL MODELO

Las 2 partes fundamentales

**Análisis:** Descomponer fuentes en una representación intermedia

**Síntesis:** Generación de programación objetiva a partir de la representación

Centrarse en el **módulo**

## notas importantes:

Hoy: hay muchas herramientas de software que usan el modelo de análisis-síntesis:

Editores dirigidos a estructura / sintaxis: Sintaxis:  
forzar la introducción del código "sintácticamente" correcto.

Impresoras estéticas: Versión estandarizadas para la estructura del programa. Es decir, espacios en blanco, sangría.

Verificadores estáticos: Una compilación "rápida" para detectar errores rudimentarios.

Interpretes: Ejecución "en tiempo real" del código  
"línea a línea"

La compilación no se limita a las aplicaciones de lenguaje de programación.

- Formateadores de texto

LATEX y TROFF son lenguajes cuyos comandos dan formato al texto.

- Compiladores de sílice:

Textos gráficos: Toman entradas y generan el diseño <sup>del</sup> circuito.

- Procesadores de consultas de bases de datos:

Los lenguajes de consulta de bases de datos también son un lenguaje de programación.

La entrada se compila en un conjunto de operaciones para acceder a la base de datos.



Gonzalez Morales Victor

22-08-24

La tarea de análisis para la compilación  
(del programa fuente)

Tres fases:

**Análisis léxico / léxico:**

Escaneo de I-a-D para identificar tokens y Secuencia de caracteres que tienen un significado colectivo.

**Análisis jerárquico:** Agrupación de tokens en una categoría significativa.

**Análisis semántico:** Comprobación para garantizar la coherencia de las componentes

FASE 1 - Análisis Léxico

23-08-24

Análisis más fácil - Identificar tokens que son los bloques de construcción básicos

**ej:**

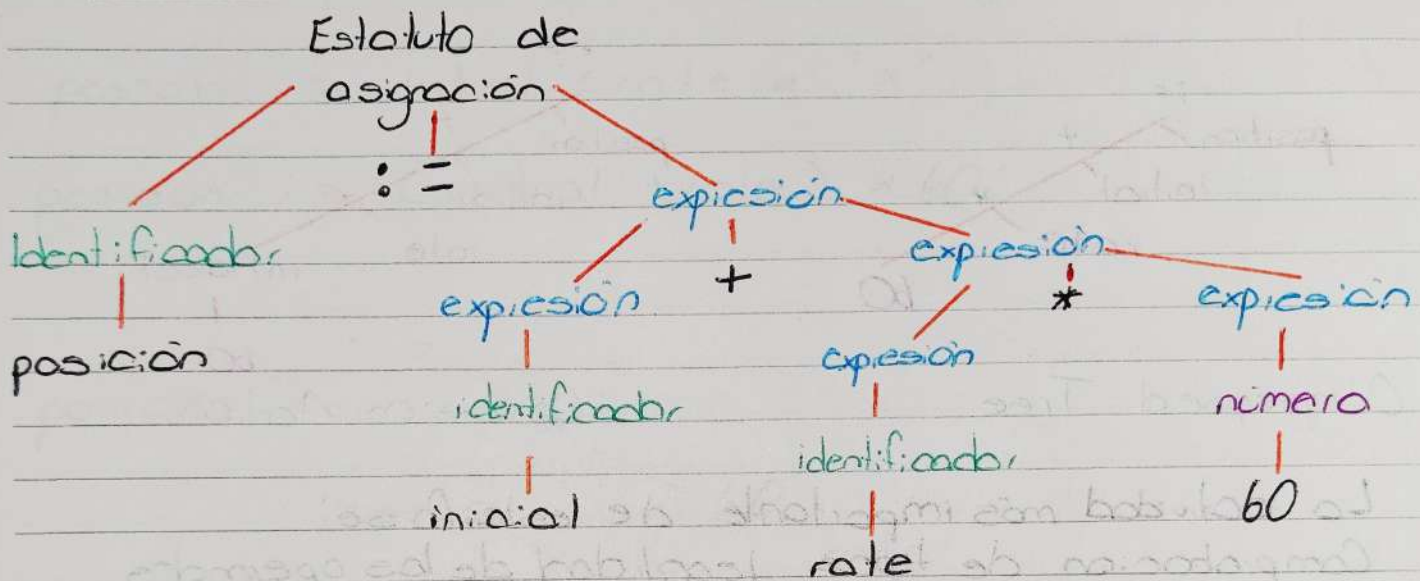
posición := inicial + rate \* 60 f



No se escanean espacios en blanco, saltos de línea, etc.

## Fase 2: Análisis jerárquico alias Parsing o Análisis de Sintáctico

Para el ejemplo anterior, tendríamos Árbol sintáctico:



Los nodos de árbol se construyen usando una gramática para el lenguaje

### ¿Qué es una gramática?

Es un conjunto de registros reglas que gobiernan las interdependencias y la estructura entre los tokens.

estituto

es

estituto de asignación, o while, o estatuto if, o...

asignación

es

identificador := expresión;

expresión

es

(expresión), o expresión + expresión, o expresión \* expresión, o número, o identificador, o...

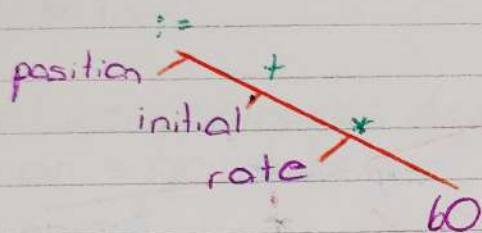


Gonzalez Morales Victor

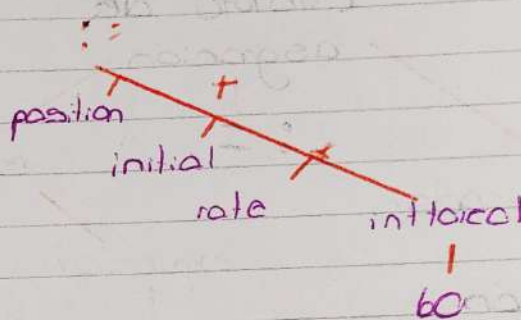
## Fase 3: Análisis Semántico

Encuentra errores semánticos más complicados y para la generación de código

El árbol de análisis se aumenta con acciones semánticas.



Compressed Tree



Conversion Action

La actividad más importante de esta fase:  
Comprobación de tipos, legalidad de los operandos

¿Por qué hemos dividido el análisis de esta manera?

Análisis léxico: escanea la entrada, sus acciones lineales no son recursivas.

Identificar solo "palabras" individuales que son los tokens del lenguaje

La recursividad es necesaria para identificar la estructura de una expresión, como se indica en el árbol sintáctico. Verificar que las palabras estén correctamente ensambladas en oraciones.

Gonzalez Morales Victor

22-Ago

expresion

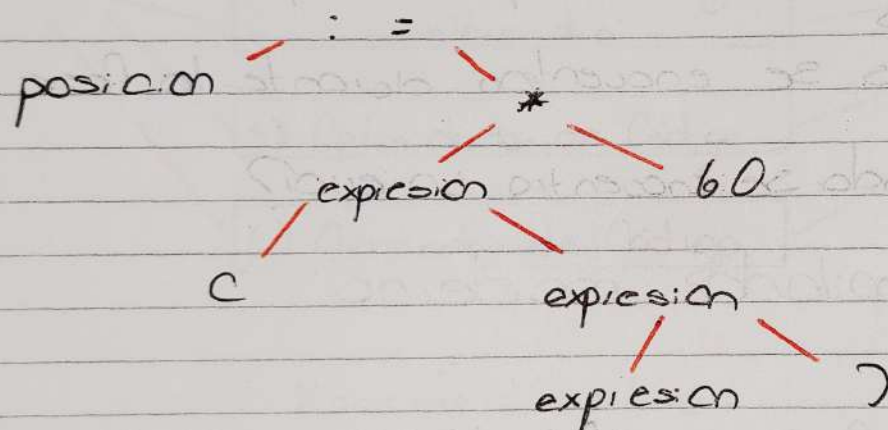
número

-208E06

initial \* posicion + 'x' rate :=

posicion := initial + rate \* 'x';

posicion := (initial + rate) \* 60;



28-Ago

Fases / actividades

de apoyo para el análisis.

Creación / mantenimiento de tabla de símbolos

- Contiene información (almacenamiento, tipo, ámbito, argumentos) de cada token "significativo" normalmente identificadores
- Estructura de datos creada / inicializada durante el proceso análisis léxico
- Utilizado / actualizado durante el análisis y la síntesis posteriores



Gonzalez Morales Victor

28/10/8

## Tablo de Símbolos

Name	Type	Def/Use/Def	other
avg	real:d	D	...
if	Keyword		...
num	int:d	D	...
sum	real:d	D	...
then	Keyword		...

## Controlador de errores

- Detección de diferentes errores que corresponden a todas las fases.
- ¿Qué tipo de errores se encuentran durante la fase de análisis?
- ¿Qué sucede cuando se encuentra un error?

## Fases de un compilador moderno

Programa fuente

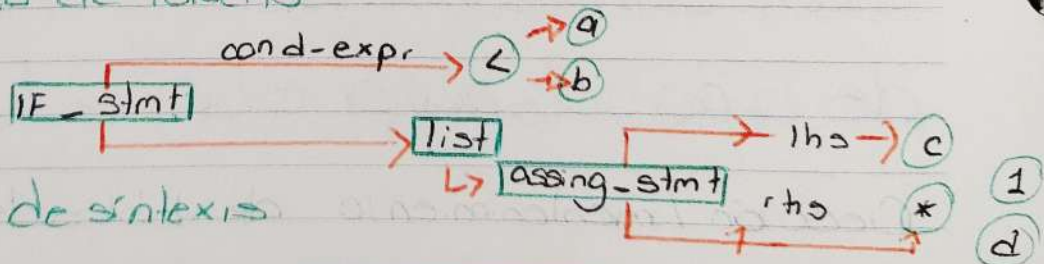
if (a < b) THEN c = 1 \* d;

Analizador Léxico

Secuencia de Tokens

IF ( (ID "a" ) < (ID "b" ) THEN (ID "c" ) = (CONST "1" ) \* (ID "d" )

Analizador Sintáctico



Árbol de sintaxis

Analizador Semántico

Código de 3 direcciones

GE, a, b, L1  
MULT, 1, d, c  
L1:

Optimizador de Código

Código de 3 direcciones Optimizado

GE, a, b, L1  
MOV, d, c,  
L1

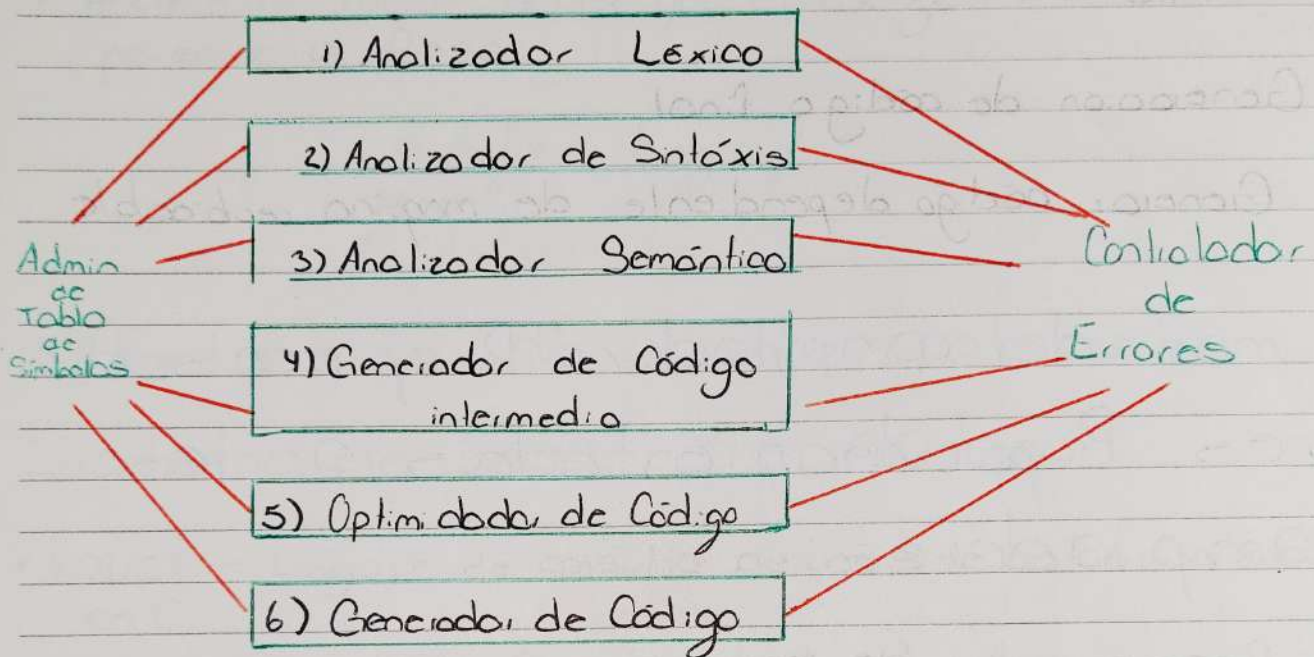
Generación de código

Código ensamblador

loadi R1, a  
cmpi R1, b  
jge L1  
loadi R1, d  
storei R1, c, L1

# Las muchas fases de un compilador

Programa fuente



Programa Objetivo

1, 2, 3: Analisis - nuestro enfoque      4, 5, 6: Síntesis

## La tarea de síntesis

29-08-24

## para la compilación

\* Generación de código intermedio

- Versión del código independiente de la arquitectura para una máquina abstracta
- Fácil de producir y hacer la generación de código final independiente de la máquina



Gonzalez Moroles Victor

## \* Optimización de código

28/10/8

- Encuentra formas más eficientes de ejecutar código
- Reemplazar código con instrucciones más óptimas
- 2 enfoques: Lenguaje de alto nivel y optimización de "micillo"

## \* Generación de código final

- Generar código dependiente de máquina recibida

Primeros del compilador: Preprocesadores.  
Proporcionan entradas a los compiladores

### 1. Procesamiento de macros:

```
# define en C: Sustituye el texto antes de compilar  
# define x 3  
# define y A*B+C  
# define z getchar()
```

### 2. Inclusión de archivos

#include en C - Trae otro archivo antes de compilar

defs.h		main.c
//////		#include "defs.h" → { ////
//////	→	.....
//////		.....
		.....

### 3) Preprocesadores racionales

- Aumentan lenguajes "antiguos" con construcciones modernas
- Agregan macros para if-then, while, etc.
- #define Puede hacer que el código C sea más parecido a Pascal

```
#define begin {
#define end }
#define then
```

### 4) Extensión de idiomas para un Sistema de bases de datos

- EQUOL - Lenguaje de consulta de base de datos incrustado en C.

```
## Retrieve (DN = Department, Dnum) where
## Department.Dname = 'research'
```

- se procesa en:

```
ingres - system ("Retr...research", ..., -);
```

- Una llamada a procedimiento en un lenguaje de programación

### Ensambladores

- Código de ensamblaje: Se utilizan nombres para las instrucciones y también se utilizan nombres para las direcciones de memoria

```
MOV a, R1
ADD #2, R1
MOV R1, b
```



Gonzalez Morales Viator

• Ensamble de pasos: 29108

- Primer paso: todos los identificadores se asignan a direcciones de memoria (desplazamiento) por ejemplo, sustituyo 0 por a, y 4 por b
- Segundo paso: Producir código máquina reubicable.

001 01 00 00000000\*  
0011 01 10 00000010  
0010 01 00 0000100\*



0001 01 00 00001111  
0010 01 10 00000010  
0010 01 00 00010011

\* bit de reubicación

## Ensambladores

Todavía no conocemos la dirección, por lo que el código fuente debe leerse dos veces.

Código fuente  
(Ensamblador:)

Primero  
Pasado

Segundo  
Pasado

Dirección  
relativa

Instrucción  
mnemónica

Dirección  
relativa

Instrucción  
mnemónica

John Start 0					
Using	* 15				
Load	1, Five	0	L1, -(0, 15)	0	L1, 16 (0, 15)
Add	1, Four	4	A1, -(0, 15)	4	A1, 12 (0, 15)
Sto	1, Temp	8	ST1, -(0, 15)	8	ST, 20 (0, 15)
Four	DC F' 4'	12	4	12	4
Five	DC F' 5'	16	5	16	5
Temp	DS 1F	20	-	20	-
End					

## Cargadores y Editores de Enlaces

- Cargador: toma código máquina reubicable, altera las direcciones y colocar las instrucciones alteradas en la memoria.
- Editor de enlaces: toma muchos programas (reubicables) de código máquina (con referencias cruzadas) y producir un solo archivo.
  - Necesita realizar un seguimiento de la correspondencia entre los nombres de las variables y las direcciones correspondientes en cada fragmento de código.

## La agrupación de fases

Front End ; Analisis + Generación de Código intermedio vs

Back end: Generación de Código + Optimización

Número de pasadas:

Una pasada: requiere lectura / escritura archivos intermedios

Menos pasadas: más eficiente

Sin embargo: menos pasadas requieren una administración de memoria más sofisticada y una interacción de fase del compilador

Equilibrio...



# Sistema de procesamiento de lenguaje

Programa fuente

1 Pre-procesador

2 Compilador

3 Ensamblador

4 Código máquina  
reubicable

5 Cargadores y  
editores de enlace

Ejecutable

Biblioteca  
archivos  
objetos  
reubicables

## Herramientas de construcción de compiladores

Generadores de analizadores: Producen analizadores de sintaxis

Generadores de escaner: Producen analizadores léxicos

Motores de traducción dirigidos por sintaxis: Generan código intermedio

Generadores automáticos de código: Generan código real

Motores de flujo de datos: Apoyan Optimización

## Herramientas

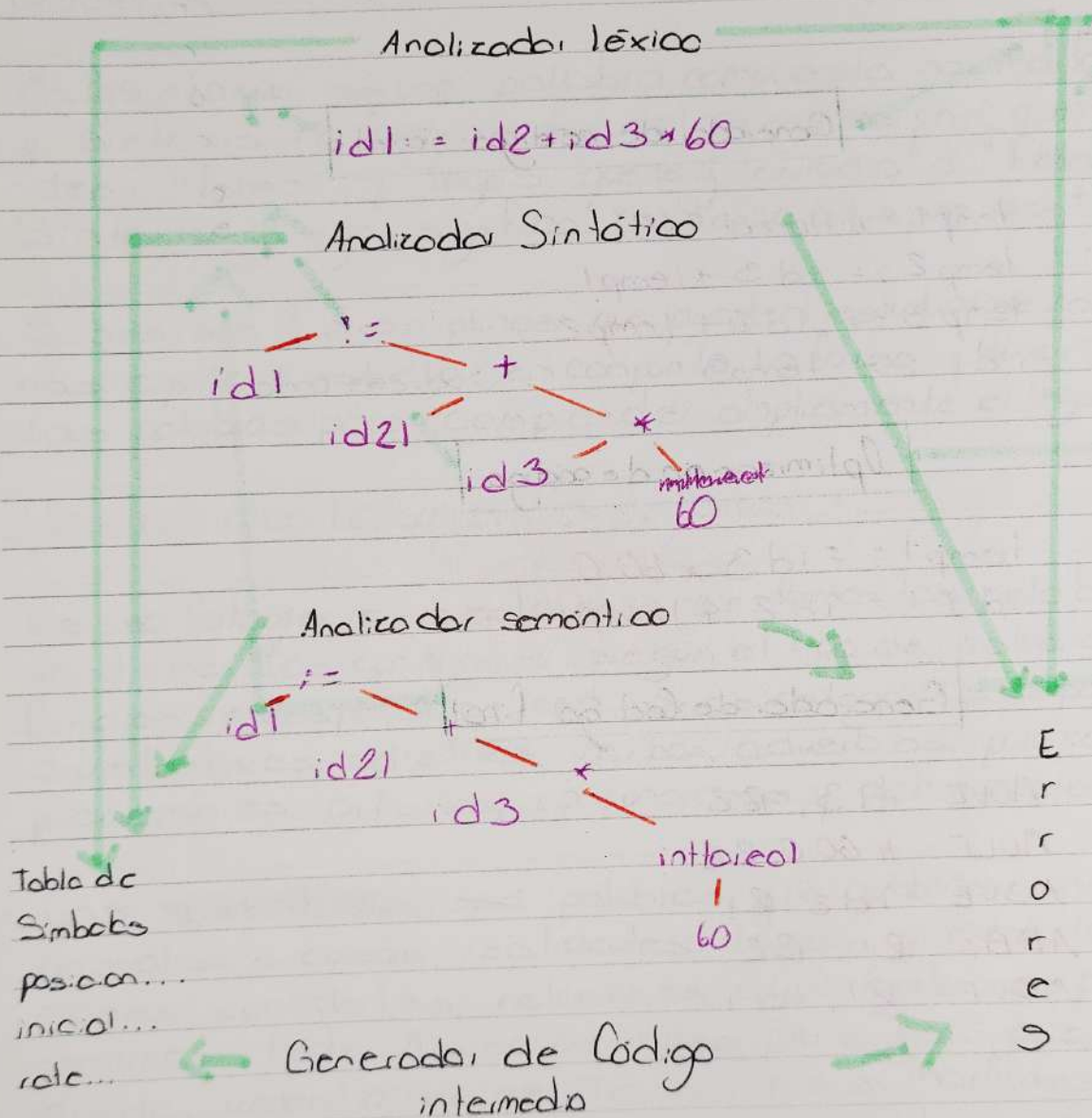
- Existen herramientas para ayudar en el desarrollo de algunas etapas del compilador
- Lex(Flex) - Generador de analizador léxico
- Yacc (Bison) - Generador de analizador sintáctico

Gonzalez Morales Victor

29/08

## Revisión de todo el proceso

$posicion := inicial + rate * 60$





# Gonzalez Morales Victor

## Revisión de todo el proceso

29/08

Tabla de símbolos

posición...

inicial...

final...

Generador de códigos intermedio

$temp1 := inttoreal(60)$   
 $temp2 := id3 * temp1$   
 $temp3 := id2 + temp2$   
 $id1 := temp3$

Address code

Optimización de código

$temp1 := id3 * 60.0$   
 $id1 := id2 + temp1$

Generador de Código final

MOVE id3, R2  
MULF #60.0, R2  
MOVE id2, R1  
ADDF R1, R2  
MOVF R1, id1

## Morfosintaxis 02-Sep

Es la parte de la gramática que estudia la formación de las palabras y la relación entre ellas para formar oraciones.

Morfosintaxis es una palabra compuesta por morfología y sintaxis. Morfología viene del griego morphé, que quiere decir "forma" y logos, que sig "estudio" o "tratado". Sintaxis, griego, significa "orden" o "disposición".

Sí, bien son 2 disciplinas que pueden estudiarse por separado, nos ayuda a estudiar, en conjunto la forma y función de las palabras para comprender ampliamente el lenguaje.

### Partes de la morfosintaxis

La morfología nos cuenta como se forman las palabras de un idioma, las categoriza según el tipo de palabras y su función. En español se ven varias categorías de palabras (sustantivos, adjetivos, verbos, adverbios, preposiciones, pronombres, artículos, conjunciones y determinantes).

- Las **sustantivos** son palabras que nombran a personas, animales o cosas, realidades físicas, y mentales (emociones, cualidades, relaciones, etc.). Pueden ser propias y comunes: Pedro, Asunción, mesa, perro, gato, paz, nación, miedo, valentía, vejez, María... Tienen morfemas de género y número para concordar con otras palabras.
- Los **adjetivos** son palabras que describen a los sustantivos: rápido, corto, mexicana, medroso, veloz, etc. y tienen morfemas de género y número que deben concordar con el sustantivo.



Gonzalez Marales Victor 021 Sep

Los **verbos** son palabras que expresan acciones y tienen marcos de número, persona, tiempo, modo y aspecto: camino, corriendo, cantó, tuviese, pensamos, etc.

Los **adverbios** son palabras invariables (que no tienen marcos de género o número) que designan diversas circunstancias: de lugar (aquí, allá), de tiempo (hoy, ayer, mañana), de modo (así), de cantidad (más, menos), de afirmación (sí, seguramente), de negación (no, jamás), de duda (tal vez, quizá, puede que), en fin.

Los **artículos** son palabras sin significado propio que determinan la presencia de un sustantivo, y concuerdan con él en género y número: el, la, los, las, un, una, unos, una.

Los **determinantes**, también llamados adyuntivos determinantes o determinativos, concuerdan en número y género, y la precisión o determinan: este, nuestra, vuestra, primera.

Los **pronombres** son las palabras que sustituyen ocasionalmente al sustantivo, y se reconocen por el contexto: ella, él, tú, vosotros, vosotras, estas, aquellas, me, se, mí.

Los **preposiciones** son palabras invariables que unen o relacionan una palabra con otras: a, ante, con, contra, de, desde, en, por, entre, bajo, para, sin, sobre, tras, durante, aunque, pero, más, etc.

Los **conjunciones** son también palabras invariables que conectan o unen palabras, sintagmas u oraciones: y, o, ni, e, que.



021 Sep

Gonzalez Morales Victor

## ¿Qué es la sintaxis?

Se encarga del ordenamiento de las palabras en una oración, y de cómo juntas pueden formar sintagmas. Sin, a su vez, tienen un sentido concreto y una función específica de la oración.

En español hay un orden por defecto (las palabras no se juntan u ordenan arbitrariamente) y es SVP (Sujeto, Verbo, predicado) = Pedro (sujeto), come (verbo), pescado (predicado). El sujeto, o sintagma nominal, es el conjunto de palabras que indica quién ejecuta la acción. Tiene un núcleo, generalmente un sustantivo o pronombre.

El verbo es la palabra que indica la acción dentro de la oración, y debe tener concordancia con el sujeto. El predicado es el sintagma verbal, y es toda la que se dice del sujeto. El núcleo siempre es el verbo.

Con la sintaxis se aprende a reconocer en las oraciones el sujeto y el predicado, los verbos y los complementos gramaticales.

## Ejemplos de análisis morfo sintáctico.

### Ejemplo #1: Los niños juegan con sus primas.

De esta oración podemos decir que es bimembre, tiene sujeto (los niños) y predicado (juegan con sus primas). El núcleo del sujeto es niños (nombre común, femenino, plural) y el núcleo del predicado es juegan (verbo intransitivo, presente de indicativo, 3ra persona del plural).



González Morales Victor 02-Sep

También podemos advertir que en esta oración hay 2 sintagmas nominales: los niños y con sus primas.

El sustantivo del primer sintagma nominal, niños, está precedido por un determinante, el artículo, Los. y el 2do sintagma nominal está precedido por un adjetivo posesivo "sus". Ambas determinantes concuerdan en número y género con los respectivos sustantivos que acompañan (niños y primas) sustantivos femeninos, plural).

Además, el sintagma verbal (juegan con sus primas), está unido al primer sintagma nominal por una preposición con. Esta convierte a sus primas en una preposicional y complemento circunstancial de compañía, que responde al ¿quién?. Es una oración simple, predicativa, activa, intransitiva, declarativa, afirmativa.

### Ejemplo #2: Comí una hamburguesa

(Yo) es el sujeto elíptico o tácito, es decir, no aparece expresamente. Comí una hamburguesa, sintagma verbal. Núcleo del sintagma: Comí (verbo transitivo, pasado simple de indicativo, 1er persona del singular).

El sintagma verbal también tiene un sintagma nominal, una hamburguesa. Está precedido por un artículo determinante, una (concorda en número y género con el sustantivo, femenino, singular, hamburguesa).

A su vez, una hamburguesa constituye un complemento directo porque dice qué fue lo que comió. El verbo comer es transitivo, y necesita ser completado por un constituyente sintáctico, que explique qué comí: 1 hamburguesa. Sobre el complemento directo recae la acción del verbo.

Es una oración simple, predicativa, transitiva, declarativa



02/Sep

Gonzalez Morales  
Victor

### Ejemplo # 3: Aún es temprano

A advertimos que no hay sujeto, pues este es impersonal, lo que significa que toda la oración es un predicado, es decir, un sintagma verbal. Está conformado por 2 sintagmas adverbiales: aún temprano y ambos son adverbios de tiempo. Sintácticamente, cumplen la función de complementos circunstanciales de tiempo.

"Es" es el verbo (verbo ser, presente de indicativo, 3era persona singular) es el núcleo del sintagma verbal. Es una oración simple, impersonal, declarativa, afirmativa.

### Ejemplo # 4: Mi hijo se parece a su padre

Oración bimembre, con sujeto y predicado. Mi hijo es el sujeto, y es el sintagma nominal del que hijo es el núcleo. Se parece a su padre es el predicado o sintagma verbal, el núcleo es se parece (acompañado de la forma impersonal se, en presente, de indicativo, 3era persona del singular, que concuerda con mi hijo).

### Ejemplo # 5: El Barça jugará mañana contra el Real Madrid

Oración bimembre: Sujeto, el Barça, predicado, jugará mañana contra el Real Madrid. Núcleo del sintagma nominal, Barça; Núcleo del sintagma verbal: jugará (verbo intransitivo, futuro simple del indicativo, 3era persona singular, que concuerda con Barça).

Contra una preposición, que enlaza y convierte Real Madrid en un sintagma preposicional de término y complemento circunstancial de modo (oposición).

KYMA



Gonzalez Moroles Victor

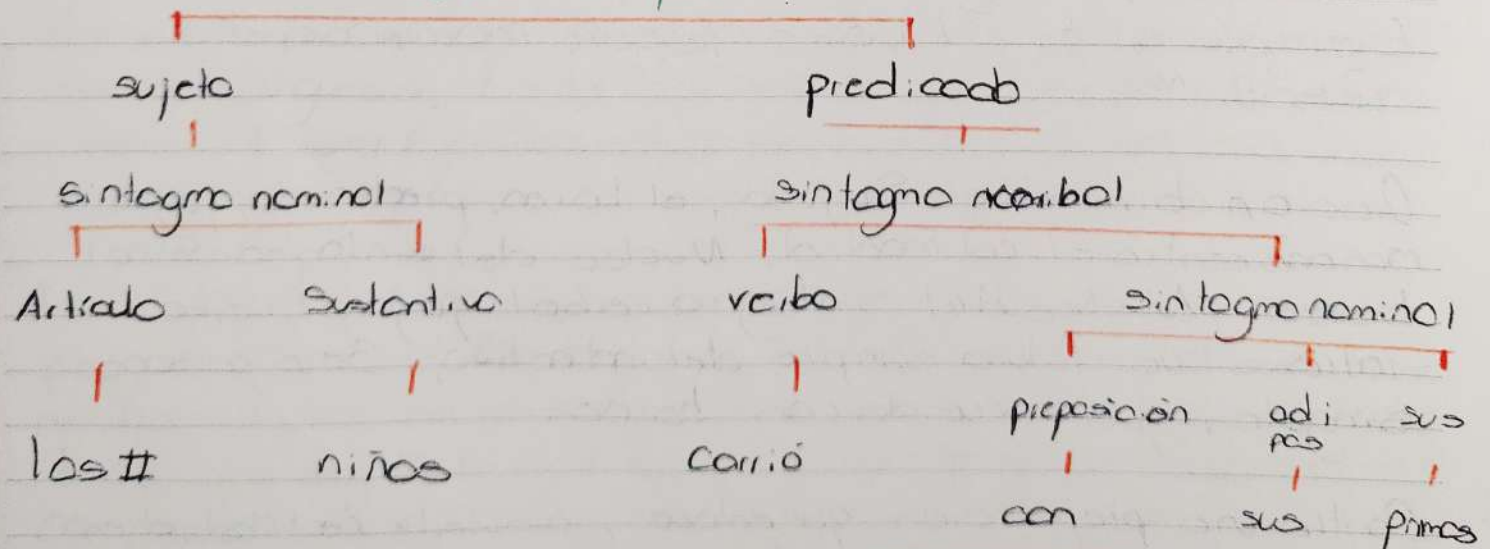
Mañana, es un adverbio de tiempo, indica un momento futuro, y por eso el verbo se conjuga en futuro.

El Real Madrid es otro sintagma nominal dentro del sintagma verbal, precedido por el determinante El. El núcleo es Real Madrid, y concuerdan ambos en género y número. Completa función sintáctica de ser el término del sintagma preposicional, es decir, el objetivo dentro de la oración: un equipo juega contra otro, que es Real Madrid.

Es una oración simple, predicativa, intransitiva, declarativa, afirmativa.

Los	Artículo	}	Analizado, Léxico
niños	sustantivo		
juegan	verbo		
con	preposición		
sus	adj. posesivo		
primas	sustantivo		

Oración simple



3 - Sep

Gonzalez Morales Victor

## Ejercicios

video: análisis morfológico de oración

1. Realiza el análisis morfológico de las sig. oraciones:

- La niña llevaba un vestido rosa

La: artículo, femenino, singular  
niña: sustantivo, común, femenino, singular  
llevaba: verbo, pasado imperfecto, indicativo, 3era persona singular  
un: masculino, determinante, indefinido  
vestido: sustantivo, masculino, singular, común  
rosa: adjetivo, singular, masculino.

- Mi perro está enfermo desde ayer.

Mi: Adjetivo posesivo, singular  
perro: sustantivo, singular, masculino  
está: verbo presente, indicativo, 3ra persona, singular  
enfermo: adjetivo, masculino, singular  
desde: preposición  
ayer: adverbio de tiempo.

### Oración simple

