

COMPILADORES E INTÉRPRETES

SAN JOSÉ GRP. 40

APUNTES DEL 10/03/2017

PROFESOR

DR. FRANCISCO TORRES ROJAS

APUNTADOR

MARIO CHINCHILLA TORUÑO

2013101565

I SEMESTRE 2017

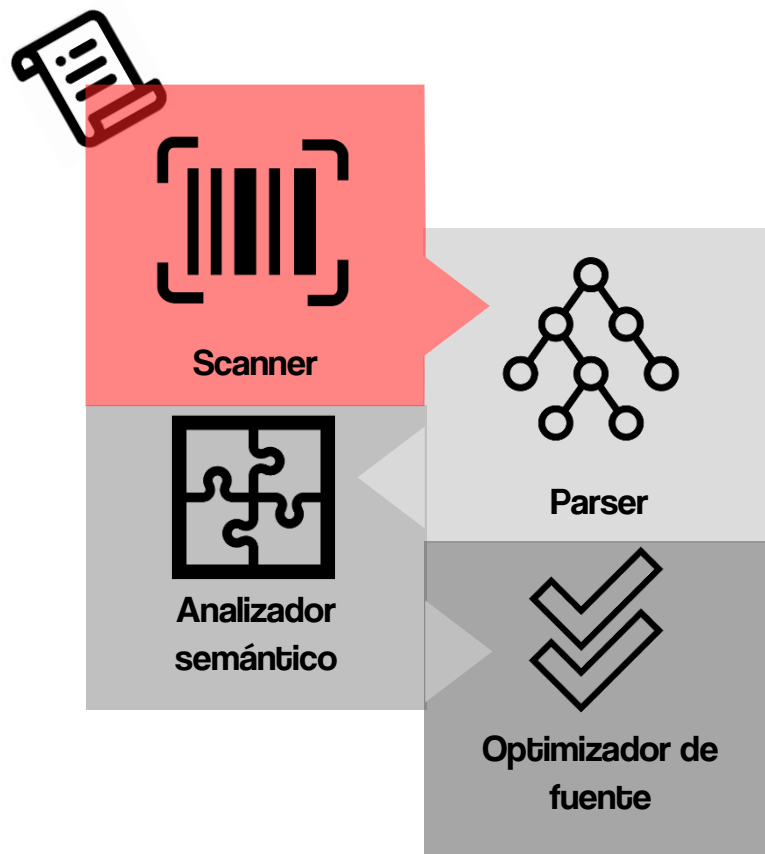
CONTENIDOS

ANÁLISIS LÉXICO	3
CORRIENDO GETTOKEN()	7
SCANNER Y ESPACIOS	7
El scanner no sabe si es DO o DO10J se debe avanzar hasta eliminar la ambigüedad.	8
GENERADORES DE SCANNERS	8
DETECCIÓN DE TOKENS	8
LENGUAJE	9
PREFIJOS DE UNA HILERA	12
SUFIJOS DE UNA HILERA	12
SUBHILERA DE UNA HILERA	13
CONCATENACIÓN DE HILERAS	13
CONCATENACIÓN DE HILERAS	14
POTENCIA N-ÉSIMA DE UNA HILERA	14
MULTIPlicAR DE ALFABETOS	15
CONJUNTO Σ^*	15
CONJUNTO Σ^+	17

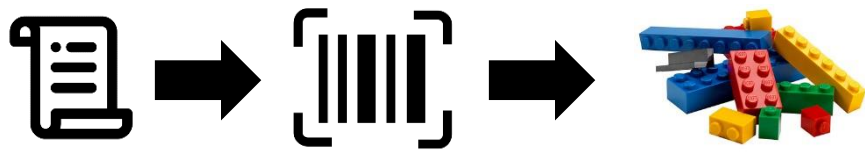
OBJETIVO DEL CURSO

Comprarle a Torres un lapicero que escanea colores, para que su existencia tenga sentido.

ANÁLISIS LÉXICO



Toma el código fuente y los descompone en tokens (categoría léxica mínima).



Equivalen a **palabras** de un lenguaje natural.

Las palabras usualmente son lo que aprendemos primero de un idioma foráneo, lo más básico que podemos entender.

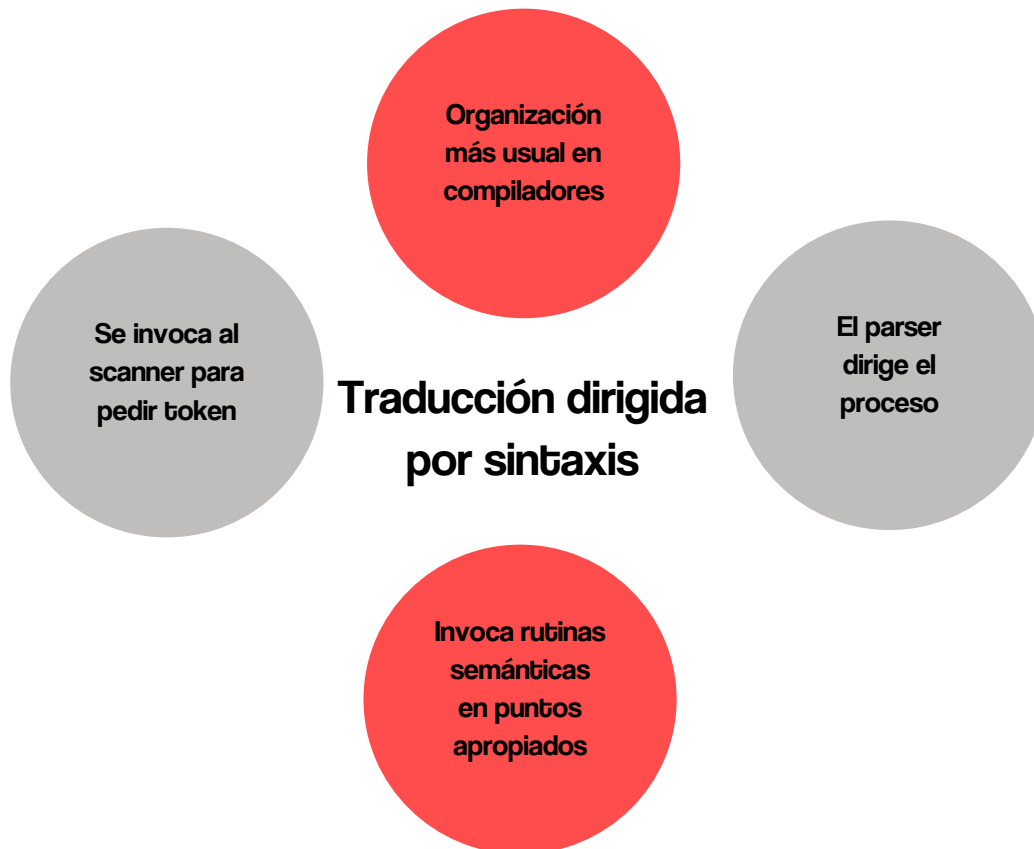
Los seres humanos a partir de la información que escuchamos creamos el lenguaje.

我的上帝，这是他妈的
Scarlett Johansson 我不
能相信这，我不能相信。



Lo más importante del analizador léxico es reconocer las palabras (Tokens) y el tipo de estas.

- Identificadores
- Literales
- Palabras reservadas
- Operadores



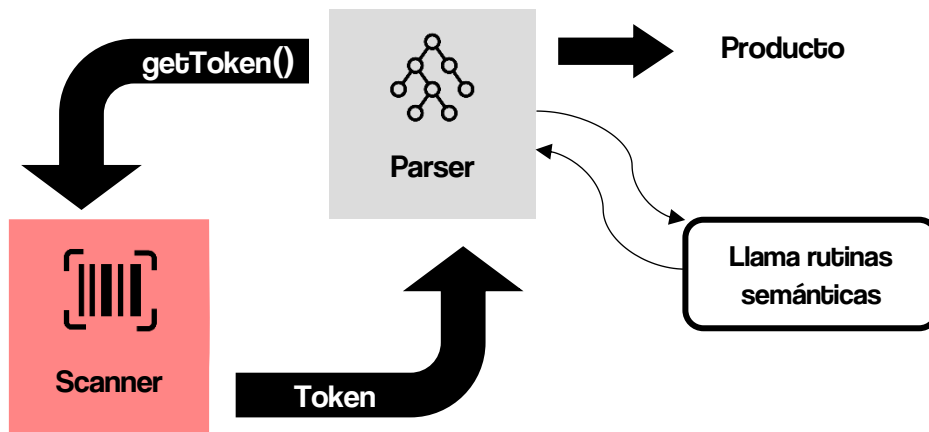
El trabajo termina al generar código al generar código intermedio o inclusive puede generar lenguaje máquina.

No es la única organización posible, pero es la más usada.

Parser es el cliente del scanner (solicita cosas al scanner).

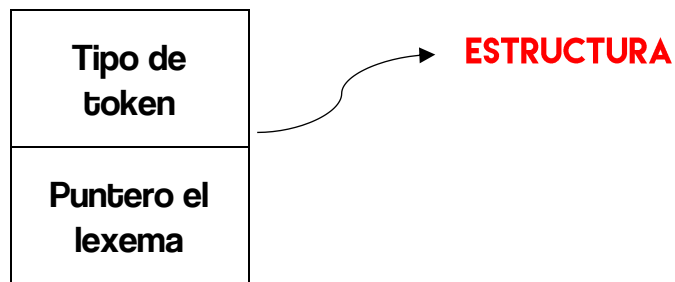


El Parser llama o utiliza getToken() para obtener el **siguiente** token



Token familia de lexemas, es una categoría.

- Un solo elemento (LEFT_PARENTHESSES).
- Cantidad pequeña de elementos (INT, FLOAT, CHAR).
- Conjunto infinito (Números, hileras).



Lexema ejemplo particular de la variable; k, cont, numPollitos, Izcar_1

- Concatenación de letras símbolos.

Hay una relación de 1 a N entre el token y los lexemas.

CORRIENDO GETTOKEN()

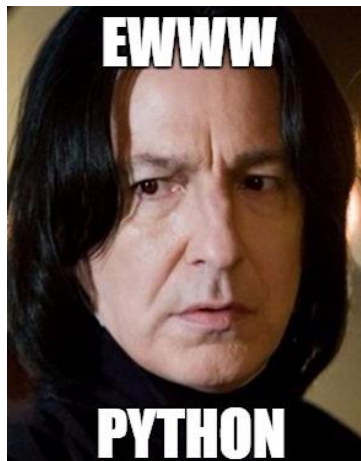
					t	a	b	l	a	[j]		=		4		*		2	;	
--	--	--	--	--	---	---	---	---	---	---	--	---	---	--	---	--	---	--	---	--	---	---	--

El scanner mantiene una posición actual.

Omite los espacios (blancos, saltos de línea, tabs, comentarios), avanza hasta reconocer un token completo y lo devuelve (la **ESTRUCTURA**).

SCANNER Y ESPACIOS

Usualmente no hay un token que represente espacios, pero en Python estos son importantes.



Hay que ignorarlos, en **Fortran** se eliminaban antes de compilar, pero no era una buena idea, se generaban ambigüedades que sobrecargaban de tareas al scanner, por lo tanto, los espacios son nuestros amigos.

AMBIGÜIDADES

Fortran

Ejemplo

$k = 0$

DO 10 J = 1, 10, 1

$k = k + J$

10 CONTINUE

D	O	1	0	J	=	1	,	1	0	,	1
---	---	---	---	---	---	---	---	---	---	---	---

El scanner no sabe si es DO o DO10J se debe avanzar hasta eliminar la ambigüedad.

D	O	1	0	J	=	1	,	1	0	,	1
---	---	---	---	---	---	---	---	---	---	---	---

Aquí se da cuenta que es DO y no una variable.

GENERADORES DE SCANNERS

Hay mucha teoría y experiencia, la mayor parte se puede automatizar (Lex, Flex).

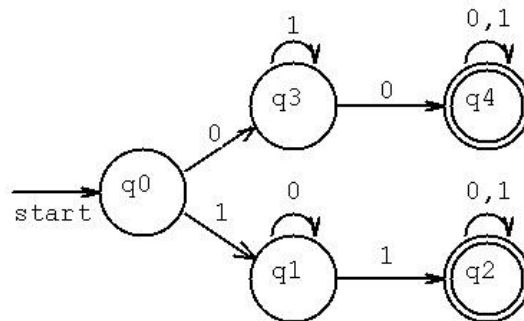
Generarlos es fácil, pero siempre es necesario agregar código extra.

DETECCIÓN DE TOKENS

Es la función principal del scanner (Tokenizador).

¿Cómo lo logra?

- Lenguajes formales
- Teoría de Autómatas
- DFA



Recomendaciones: Leer "Introduction to automata theory languages and computation".

LENGUAJE

Lenguaje:

- Habilidad para adquirir y usar complejos sistemas de comunicación.
- Conjunto de sonidos articulados con que el humano manifiesta lo que piensa o siente.

Lingüística: ciencia que estudia el lenguaje humano y las lenguas.

Lenguajes Natural

Español, inglés, francés.

Lenguajes de computadora

Español, inglés, francés.

Lenguaje Matemático

Lenguajes Formales

Símbolos primitivos y reglas para unir esos símbolos están formalmente especificados.

Símbolos (termino primitivo = algo que se asume)

Es la representación **convencional** de un concepto, idea, aunque realmente es arbitrario.

Alfabeto

Conjunto de símbolos / finito no vacío.

Σ : alfabeto

Σ : {0, 1}

Σ : {A, T, C, G}

Σ : {Ala, Arg, Asn, Cys, Gln, Leu, Trp} -> aminoácidos

Hilera sobre Σ , **secuencia** (palabras, frases) de longitud arbitraria formada con símbolos tomados de un alfabeto Σ .

Ejemplos:

Sea $\Sigma: \{0, 1\}$

10101111

NULL

1111

101010111001

Tiene un orden,
puede haber
símbolos
repetidos.

$\Sigma: \{A, T, C, G\}$

ATCGTTTCG

A

NULL

GG

Longitud de hilera

Cantidad de símbolos.

$|w|$: sea w una hilera sobre Σ .

$|GATA| = 4$

$|TGCCGTATGCATGACTAGAGT| = 21$

$|\epsilon| = 0$ es una hilera **vacía** se puede representar con **épsilon** o **lambda**.

PREFIJOS DE UNA HILERA

w sobre alfabeto Σ , el **prefijo** de w es una hilera de los primeros k símbolos desde la **izquierda**.

- $0 \leq k \leq |w|$.
- $k < |w|$ prefijo propio.
- $k=0$ prefijo vacío.
- ϵ es prefijo de cualquier hilera.

$\Sigma: \{A, T, C, G\}$

$k = 6$

TGCCGTATGCATGACTAGAGT

SUFIJOS DE UNA HILERA

w sobre alfabeto Σ , el **prefijo** de w es una hilera de los últimos k símbolos desde la **derecha**.

- $0 \leq k \leq |w|$.
- $k < |w|$ sufijo propio.
- $k=0$ sufijo vacío.
- ϵ es sufijo de cualquier hilera.

$\Sigma: \{A, T, C, G\}$

$k = 6$

TGCCGTATGCATGACT**AGAGT**

SUBHILERA DE UNA HILERA

w sobre alfabeto Σ , el **prefijo** de w es una hilera de los k símbolos desde la **derecha** a partir de una posición j .

- $0 \leq k, j \leq |w|$.
- $0 \leq k + j \leq |w|$.
-
- $k + j < |w|$ y $j > 0$ subhilera propia.
- $k=0$ subhilera vacía.
- ϵ es subhilera de cualquier hilera.

$$\Sigma: \{A, T, C, G\}$$

$$k = 6, j = 6$$

TGCCGT**ATGCAT**GACTAGAGT

CONCATENACIÓN DE HILERAS

Sean v y w sobre alfabeto Σ .

Si

$$v = a_1, a_2, a_3, \dots, a_k$$

$$w = b_1, b_2, b_3, \dots, b_k$$

entonces la **concatenación** de v y w es $a_1, a_2, a_3, \dots, a_k b_1, b_2, b_3, \dots, b_k$ y se denota como **vw** .

Ejemplos

Sea $\Sigma = \{0, 1\}$

$v = 11$

$w = 01000$

$x = 1101$

$y = 011101110$

Entonces

$vw = 1101000$

$xy = 1101011101110$

$yy = 011101110011101110$

CONCATENACIÓN DE HILERAS

- Es asociativa.

$$vwx = (vw)x = v(wx)$$

- No es conmutativa.
- Longitudes se suman.

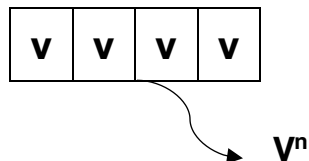
$$|vw| = |v| + |w|$$

- Hay un elemento neutro de la concatenación.

$$\xi v = v \xi = v$$

POTENCIA N-ÉSIMA DE UNA HILERA

- Sea v una hilera sobre Σ .
- La potencia n -ésima de v es la hilera resultado de concatenar n copias de v .



- $v^1 = v$
- $v^0 = \xi$

Ejemplos

$$v = \text{ATTAC}$$

$$v^2 = \text{ATTACATTAC}$$

MULTIPLICAR DE ALFABETOS

Σ^k conjunto de todas las hileras sobre el alfabeto tales que tengan longitud de k .

$$\text{Sea } \Sigma = \{0, 1\}$$

$$\Sigma^2 = \{00, 01, 10, 11\}$$

$$\Sigma^3 = \{000, 001, 010, 011, 100, 101, 110, 111\}$$

$$\Sigma^0 = \{\epsilon\}$$

CONJUNTO Σ^*

Conjunto de hileras de cualquier longitud formadas únicamente a partir de símbolos del alfabeto Σ .

Conjunto infinito.

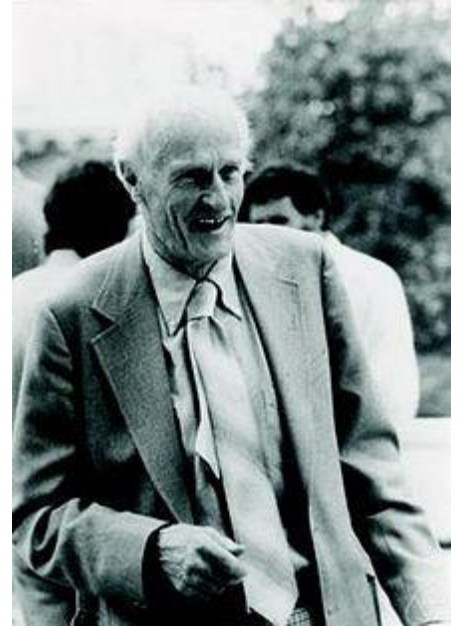
1. $\epsilon \in \Sigma^*$
2. Si $w \in \Sigma^*$ y $a \in \Sigma$, entonces $wa \in \Sigma^*$

3. $w \in \Sigma^*$ solo si se puede ser construida desde ϵ usando el paso 2 repetidamente. Equivalentemente $\Sigma^* =$

$$\bigcup_{i=0}^{\infty} \Sigma^i = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \dots \cup \Sigma^{\infty}$$

Stephen Cole Kleene (1909-1994)

- Fue un lógico y matemático estadounidense.
- PhD Matemática de Princeton University.
- Estudio con Alonzo Church.
- Tenía trabajos en lógica.
- Creador de lenguaje formal.
- Cierre de Kleene Σ^* .



CONJUNTO Σ^+

Conjunto de hileras de longitud mayor a cero formadas únicamente a partir de símbolos del alfabeto Σ .

Conjunto infinito.

1. $a \in \Sigma, \Rightarrow a \in \Sigma^+$
2. $w \in \Sigma, \Rightarrow a \in \Sigma$, entonces $wa \in \Sigma^+$
3. $w \in \Sigma^+$ solo si se puede ser construida desde ϵ usando el paso 2 repetidamente. Equivalentemente $\Sigma^+ =$

$$\bigcup_{i=0}^{\infty} \Sigma^i$$