

## DIAGRAMAS DE TRANSICIONES

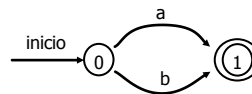
- Son grafos dirigidos que especifican el reconocimiento de un token como elemento del lenguaje.
- En un diagrama de transición (**DT**) se compila un token.
- Partes:



- Se dice que un token se compila en un DT, si este es reconocido o cumple una ruta desde el estado inicial hasta un estado de aceptación (estado final).
- DT básicos:

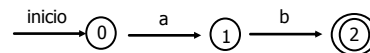
- **ER:**  $a \mid b$

$$L(a \mid b) = L(a) \mid L(b) = \{a, b\}$$



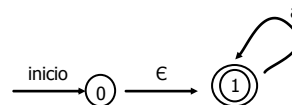
- **ER:**  $ab$

$$L(ab) = L(a) L(b) = \{ab\}$$



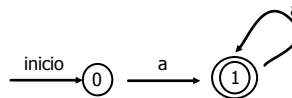
- **ER:**  $a^*$

$$L(a^*) = L(a)^* = \{\epsilon, a, aa, aaa, aaaa, \dots\}$$

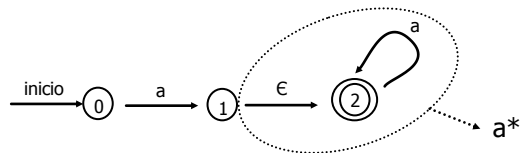


- **ER:**  $a^+$

$$L(a^+) = L(a)^+ = \{a, aa, aaa, aaaa, \dots\}$$



$$a^+ \rightarrow aa^*$$

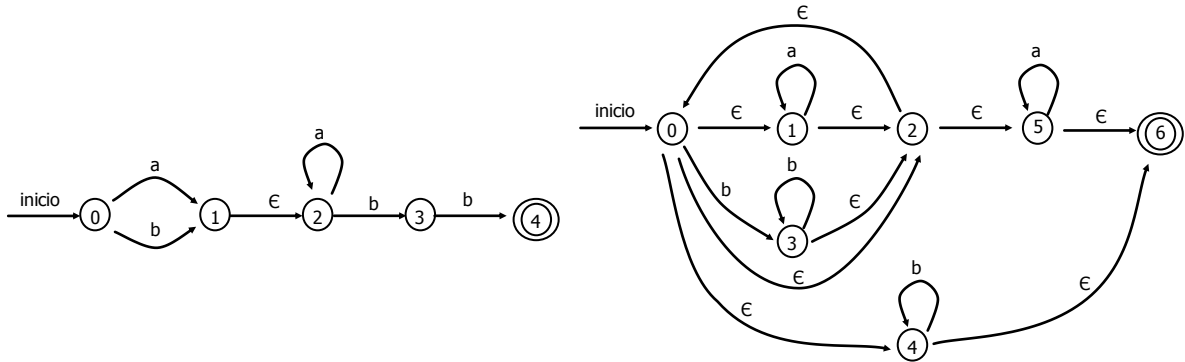


## EJERCICIOS

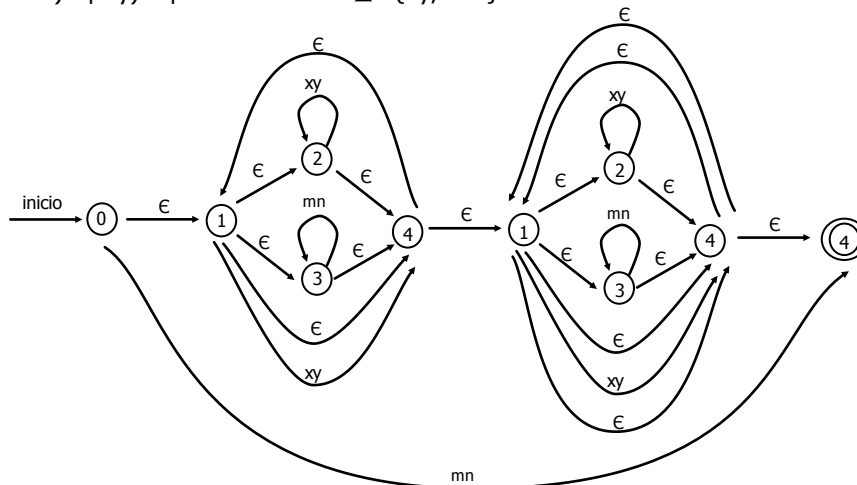
### I. Desarrollar los Diagramas de Transición (DT) de las siguientes Expresiones Regulares (ER).

1.  $(a \mid b) a^* b b$  sobre  $\Sigma = \{a, b\}$

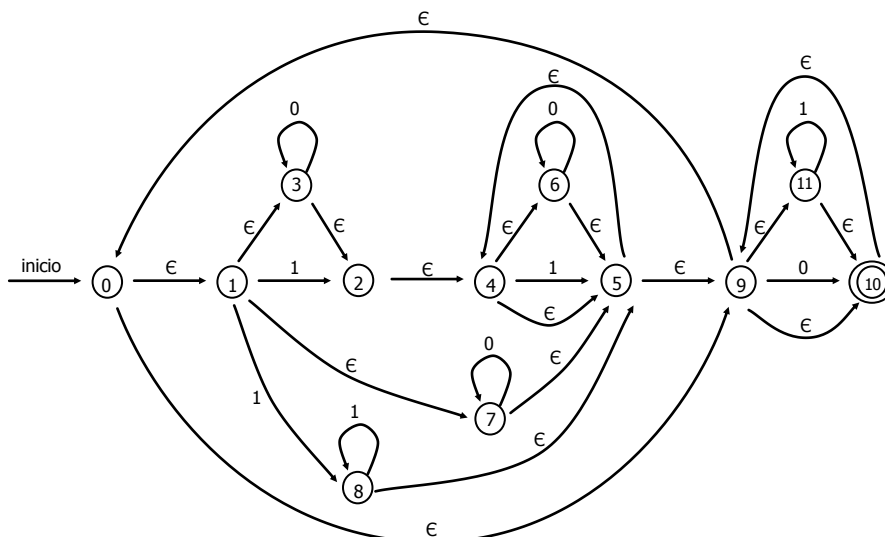
2.  $(a^* \mid b^+) a^* \mid b^*$  sobre  $\Sigma = \{a, b\}$



3.  $((xy^* \mid mn^*)^* \mid xy)^+ \mid mn$  sobre  $\Sigma = \{xy, mn\}$



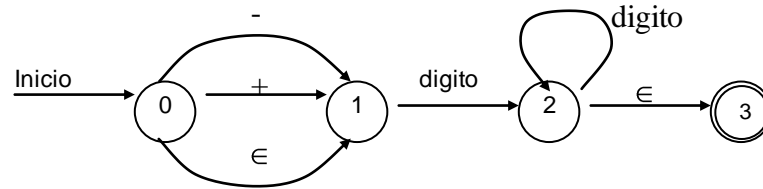
4.  $((0^* \mid 1)^+ \mid 0^* \mid 1^+)^* (0 \mid 1^*)^*$  sobre  $\Sigma = \{0, 1\}$



## II. Desarrollar los Diagramas de Transición (DT) de las siguientes Definiciones Regulares (DR).

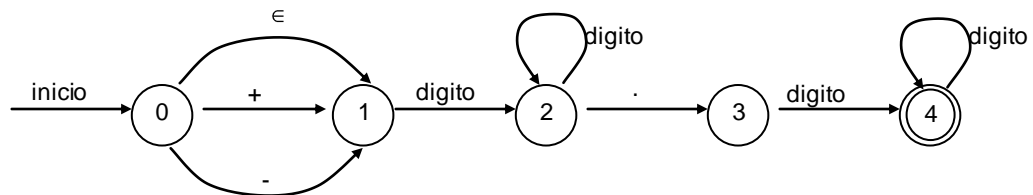
1. Números enteros con signo y sin signo.

**DR:** num\_entero  $\rightarrow (-|+|\epsilon) \text{ digito}^+$



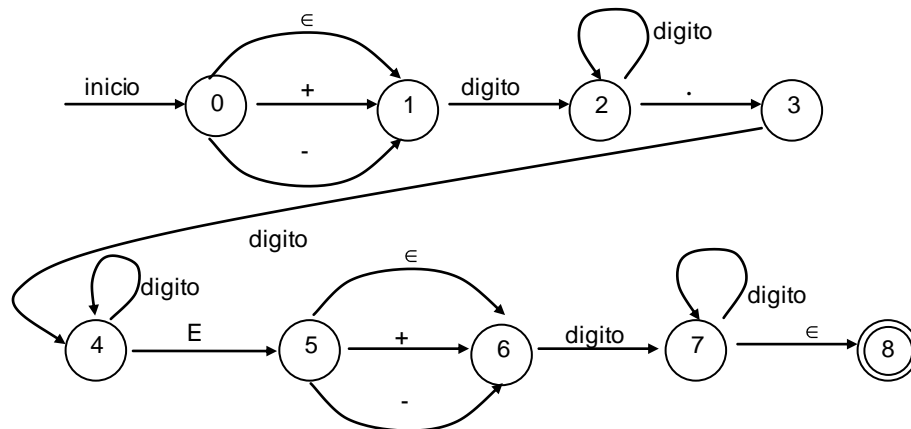
2. Números reales con signo y sin signo.

**DR:** num\_real  $\rightarrow (-|+|\epsilon) \text{ digito}^+ . \text{ digito}^+$



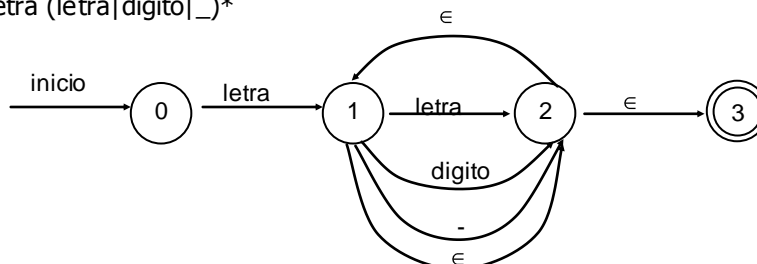
3. Numero en notación científica con signo y sin signo.

**DR:** num\_cient  $\rightarrow (-|+|\epsilon) \text{ digito}^+ . \text{ digito}^+ E (-|+|\epsilon) \text{ digito}^+$

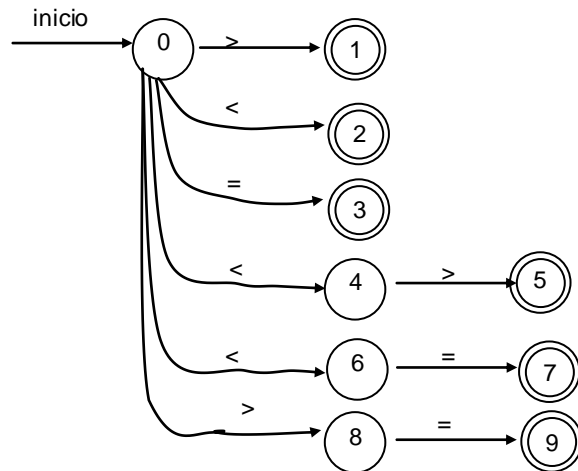


4. Identificadores

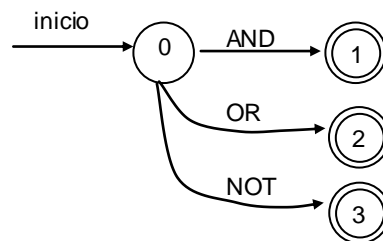
**DR:** id  $\rightarrow \text{letra} (\text{letra}|\text{digito}|\_)^*$



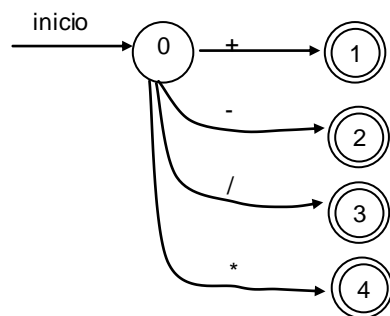
## 5. Operadores relacionales



## 6. Operadores lógicos



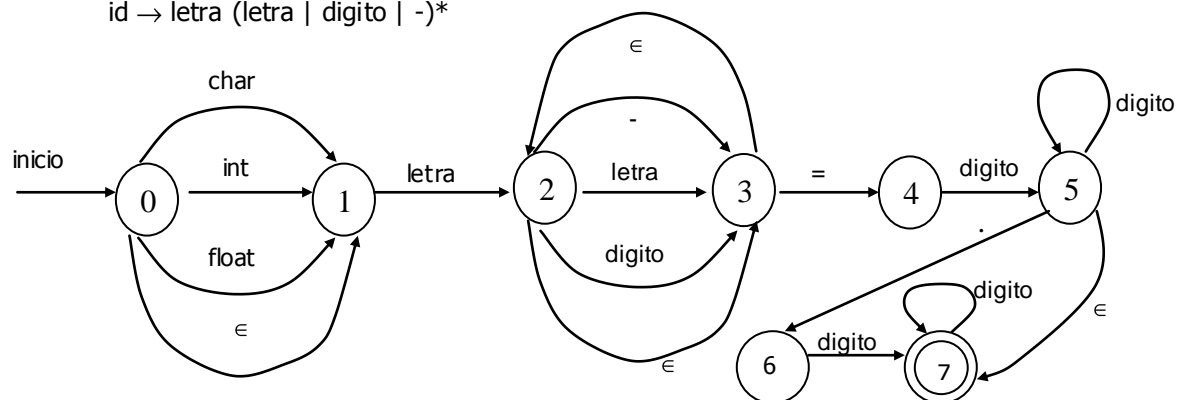
## 7. Operadores aritméticos



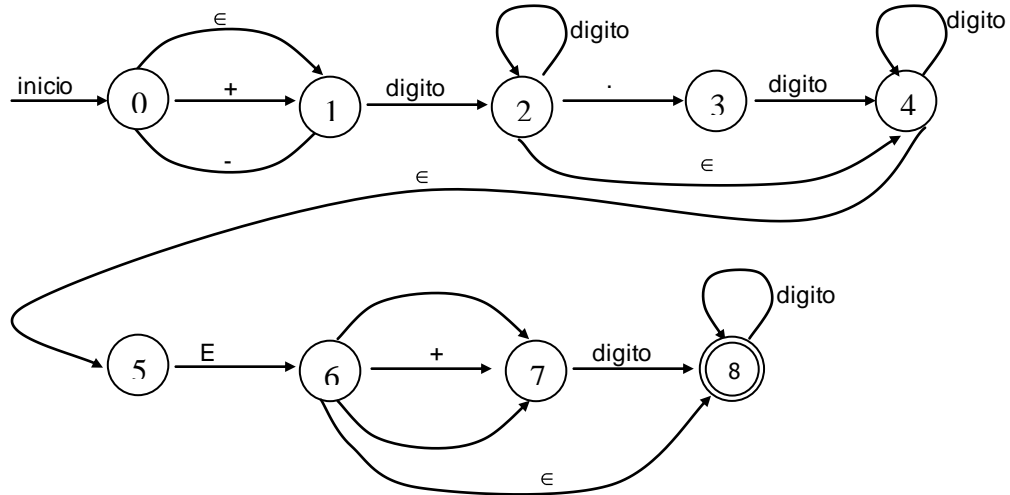
## 8. Constantes numéricas

**DR:**  $\text{const} \rightarrow (\text{int} \mid \text{flota} \mid \text{char} \mid \epsilon)$   $\text{id} = \text{digito}^+(\text{digito}^+ \mid \epsilon)$

$\text{id} \rightarrow \text{letra} (\text{letra} \mid \text{digito} \mid -)^*$



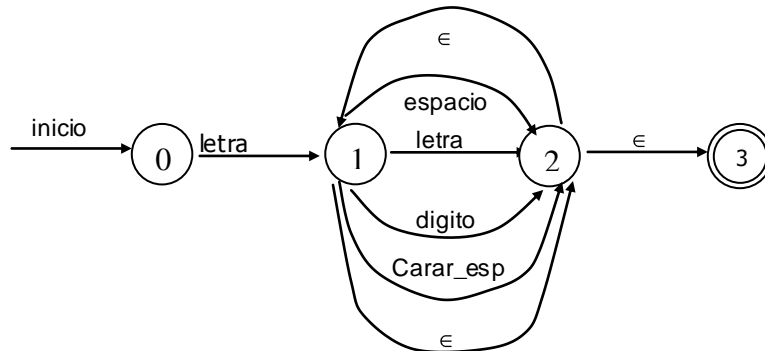
## 9. Numero entero, real y notación científica, todo al mismo tiempo con y sin signo

$$\text{num\_ent\_real\_not} \rightarrow (+|-|\epsilon) \text{ digito}^+ (. \text{ digito}^+ |\epsilon) (E(+|-|\epsilon) \text{ digito}^+ |\epsilon)$$


## 10. Cadena de caracteres

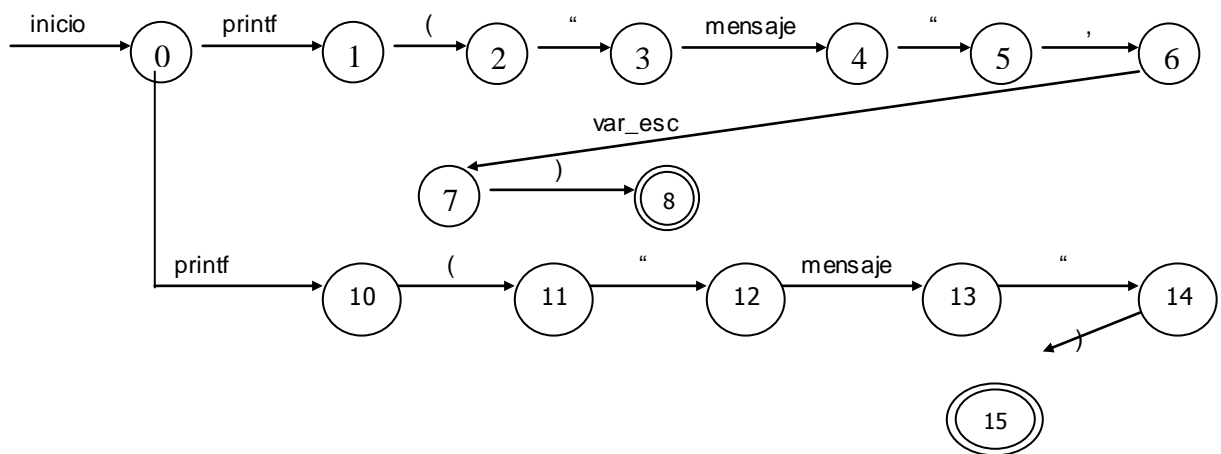
$$\text{cadena} \rightarrow \text{letra}(\text{letra} | \text{digito} | \text{espacio} | \text{carar\_esp})^*$$

$$\text{letra} \rightarrow [A-Z] - [a-z]$$

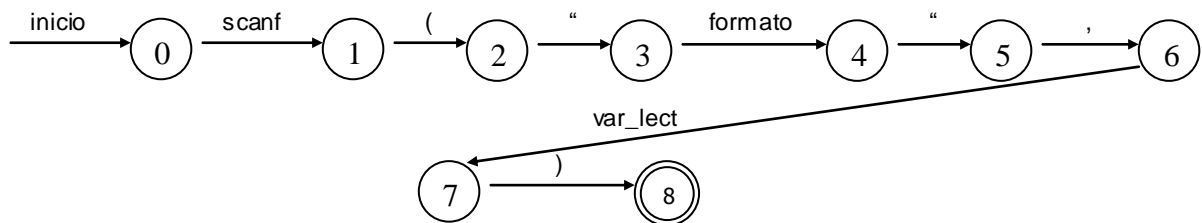
$$\text{carar\_esp} \rightarrow * | ! | ) | \& | \dots$$


## 11. Sentencia printf()

prop\_printf  $\rightarrow$  printf("mensaje", var\_esc) | printf("mensaje")



## 12. Sentencia scanf()



## 13. Definición de una función de usuario

func  $\rightarrow$  t\_dato id (param |  $\epsilon$ )  
 param  $\rightarrow$  t\_dato id ( , t\_dato id ) \*  
 t\_dato  $\rightarrow$  int | float | char  
 id  $\rightarrow$  letra (letra | digito | \_ ) \*

