C1: 19 abril

C2: 24 mayo

C3: 14 junio

aux: Lunes

Martes 23: no hay cátedra

Autómatas y Gramáticas conocen el mundo real:

identificador C:

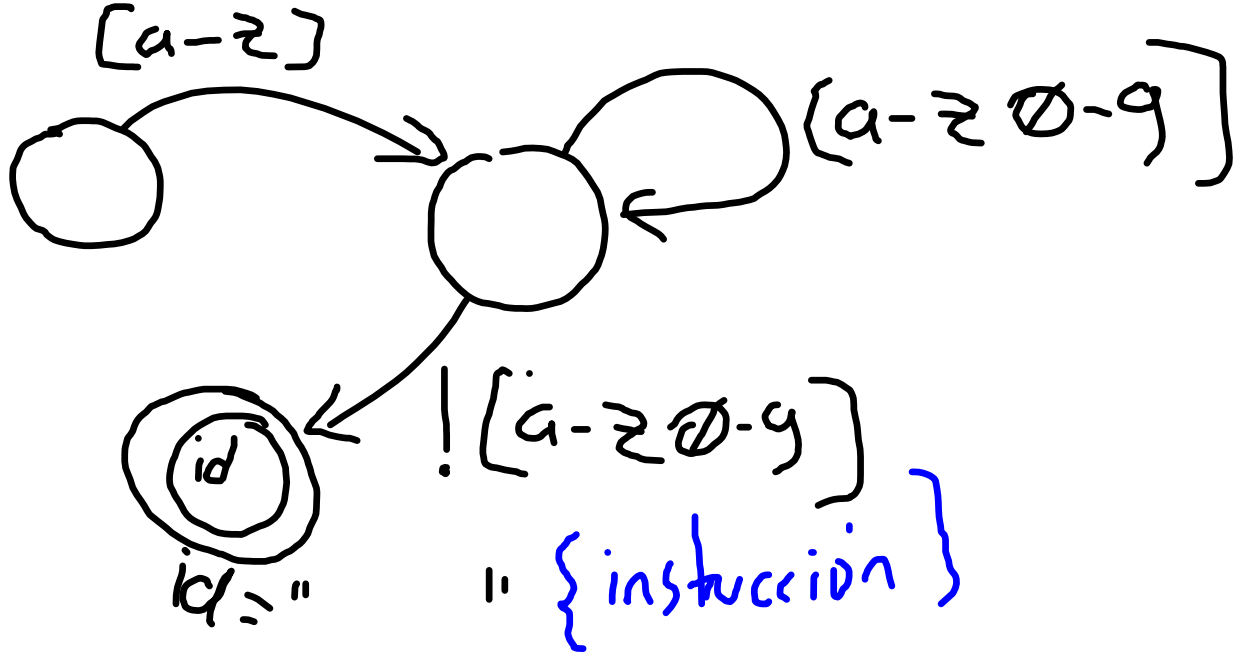
[_a-z][_a-zØ-9]* ⟶ autómata ⟶ léxico

Expresión:

E ⟶ E + E ⟶ sintáctico

Lexico:

id : [a-z][a-z0-9]*



[+] [0-9] [0-9]*

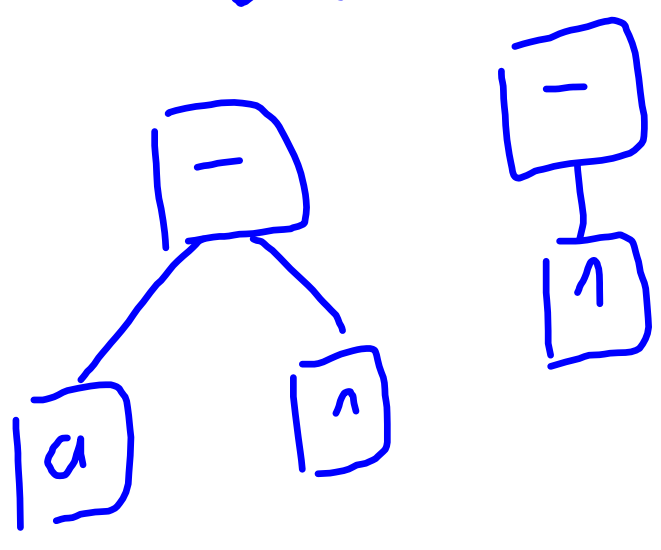
a - 1

a: | id | a | op | - | cte | 1 |

b: | id | a | cte | -1 |

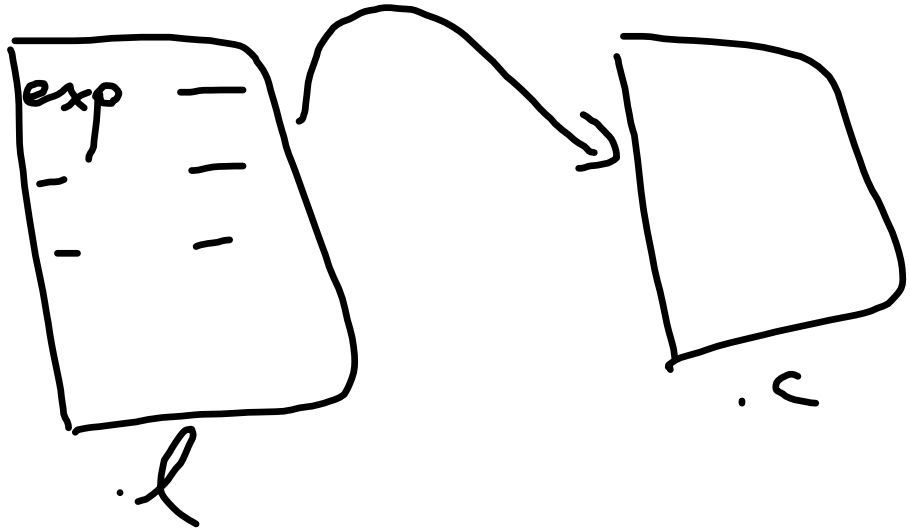
op:

- unario



Unix:

flex:



Sintaxis:

$S \longrightarrow id$

$S \longrightarrow (S)$

$\longrightarrow id$ ✓

$\longrightarrow (id)$ ✓

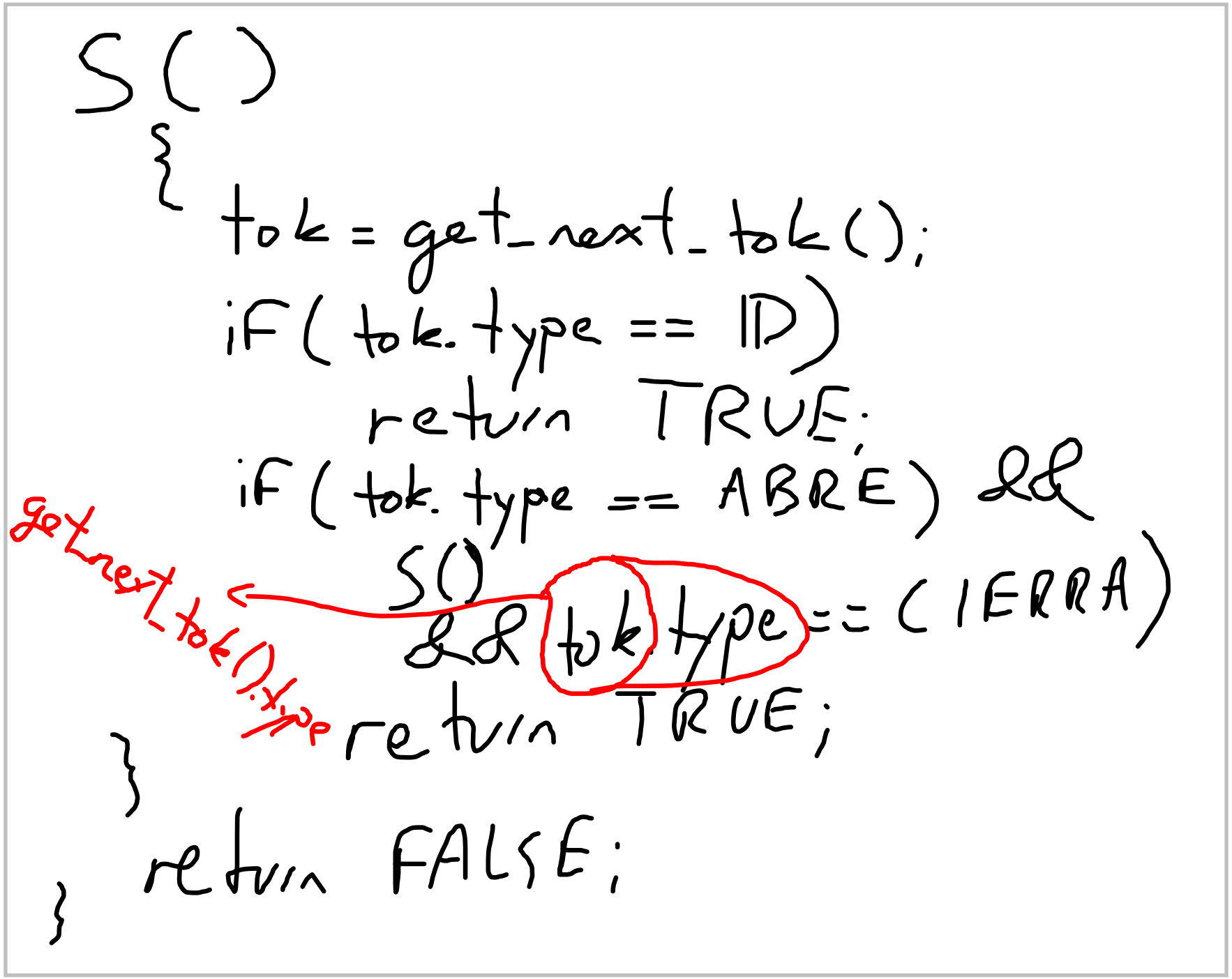
$\longrightarrow ((id))$ ✓

$((id)$ ✗

$()$ ✗

$id(id)$ ✗

$S \longrightarrow \varepsilon$


S()
{
tok = get_next_tok();
if(tok.type == ID)
return TRUE;
if(tok.type == ABRE) &&
S()
&& tok.type == CIERRA)
return TRUE;
}
return FALSE;
}
get_next_tok().type

↳ Descenso Recursivo:

Gramáticas simples

pueden ser detectadas así

→ trivial modificar y

hacer un evaluador /

generador del árbol sintáctico