

Language L:

Arithmetic expressions with:

"integers", "+", "-", "x", "/", "(", ")"

$$X = \{0-9, +, -, \times, /, (, )\}$$

our tokens are regular so we use finite automata (DFA)

Tokens:

INT, PLUS, MUL, LPAREN, RPAREN  
0-9      +      x      (      )

language Grammar:

$$\begin{aligned} S &\rightarrow S + T \mid T \\ T &\rightarrow T \times F \mid F \\ F &\rightarrow (E) \mid \text{INT}. \end{aligned}$$

Grammar ~~not in FNG~~ Pour appliquer FNG

removing left recursion:

$$S \rightarrow T S'$$

$$S' \rightarrow + T S' \mid$$

$$T \rightarrow F T'$$

$$T' \rightarrow \times F T'$$

$$F \rightarrow (E) \mid \text{INT}.$$

← our FA  
our automata in FNG  
f.

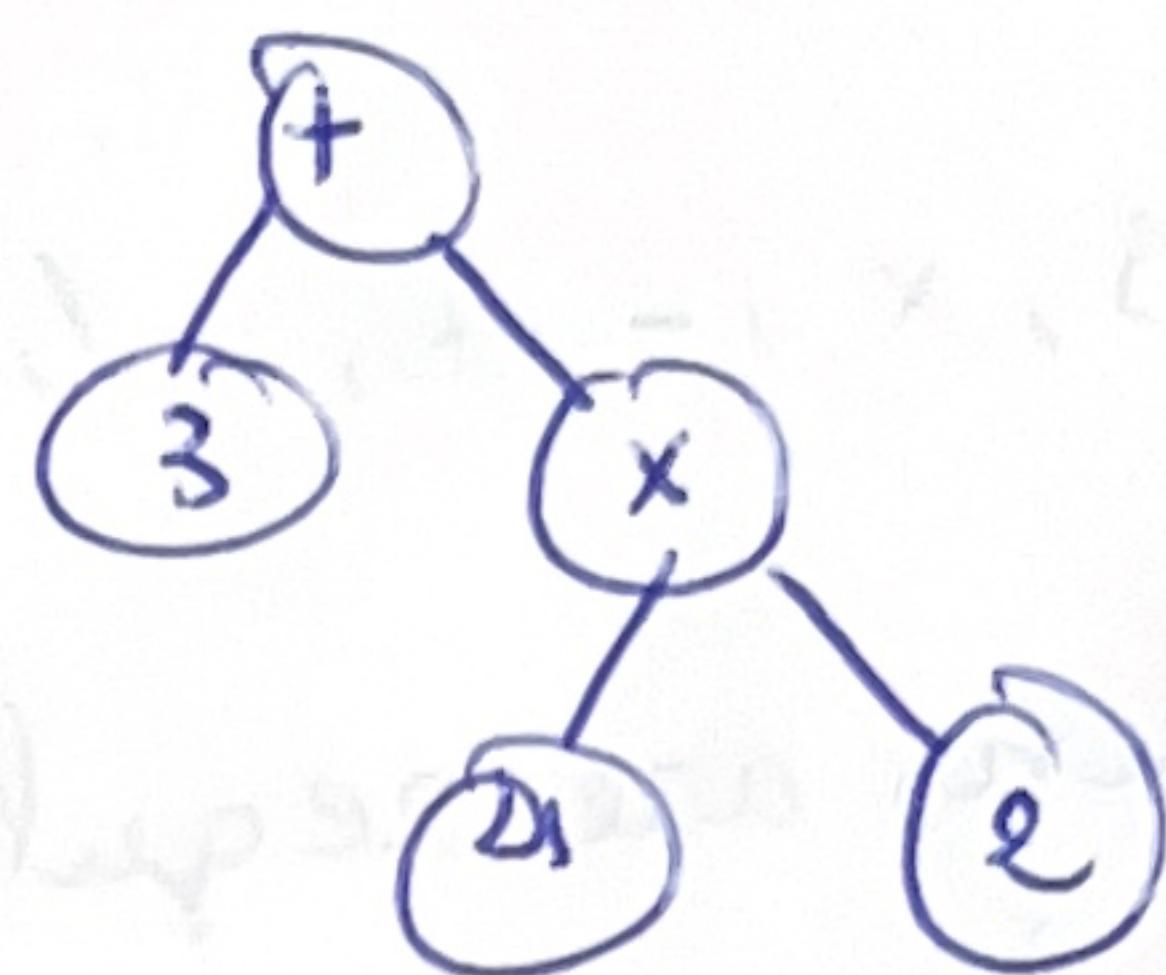
## Abstract Syntax Tree.

2 Types of Nodes :- binary Expression (+, ×)

- Number

example

$$3 + 2 \times 2 \rightarrow$$



$$\begin{aligned} T &| T + T \leftarrow T \\ T &| T \times T \leftarrow T \\ T &| 3 \leftarrow T \end{aligned}$$

using keep going with ~~substitution~~ forward substitution of previous

$$T \leftarrow T$$

$$T | T + T \leftarrow T$$

$$T | T \times T \leftarrow T$$

$$T | 3 \leftarrow T$$

$$T | T + T \leftarrow T$$

where  
domestic tree