Théorie des langages

an alphabet (somme) is a set of characters of languages

if, else, (, ) ect….

a word ‘w’ on an alphabet (somme) is sequence of characters form (somme)

“.” is the concat between two words

(somme) (lambda) a,d “.” is call monoid

/ Important \ Arden Theorem

a recursive definition of language:

L = aL + b

can be transformed into ER using Arden's theorem

L = aL + b ⇔L = a\* b

right version of arden theorem

L = L a + b ⇔ L = b a\*

exemple 1:

L = a b\*

L0 = a b\* = a L1

L1 = b\* = b\* . lambda = b L1 + lambda

Exemple 2:

Der = a b+ a\* c

L0 = a b+ a\* c = a L1

L1 = b+ a\* c = b b\* a\* c = b L2

L2 = b+ a\* = b L2 + a\* c = b L2 + (a a\* + lambda) c = b L2 + a a\* c + c lambda = b L2 + a L3 + c L4

L3 = a\* c = (a a\* + lambda) c = a a\* c + c = a L3 + c L4

L4 = lambda



Exemple 3:

a+ + b(b + a)\* c



Exercise cours

| Λ | if ( id == int ) id = id ;$ | Shift if |
| --- | --- | --- |
| If | ( id == int ) id = id ;$ | Shift ( |
| If ( | id == int ) id = id ;$ | Shift id |
| If ( id | == int ) id = id ;$ | Reduce 8 |
| If ( F | == int ) id = id ;$ | Reduce 6 |
| If ( T | == int ) id = id ;$ | Reduce 4 |
| If ( E | == int ) id = id ;$ | Shift == int |
| If ( E == int | ) id = id ;$ | Reduce 8 |
| If ( E == F | ) id = id ;$ | Reduce 6 |
| If ( E == T | ) id = id ;$ | Reduce 4 |
| If ( E == E | ) id = id ;$ | Shift ) |
| If ( E == E ) | id = id ;$ | Reduce |
| If ( B ) | Id = id ;$ | Shift id = id |
| If ( B ) id = id | ;$ | Reduce |
| If ( B ) id = E | ; | Shift ;$ |
| If ( B ) id = E; | $ | Reduce |
| If ( B ) S | $ | Reduce |
| S | $ | Shift $ |
| S$ |  | Accept |