

Théorie des Langages Solutions des Exercices de la Série 1

Exercice 4 :

Les grammaires qui génèrent les langages suivants :

1) L'ensemble des nombres binaires (un langage infini qui ne contient pas le mot vide)

Exemple : $L = \{0, 1, 01, 10, 00, 11, 001, 111, 00000, 10010, 00110, \dots\}$

$G = \langle T, N, S, P \rangle$ où $T = \{0, 1\}$ $N = \{S\}$ et $P =$

$S \rightarrow 0S / 1S / 0 / 1$

Remarque : Les nombres commencent par un 0 ou 1. Après la première lettre, il y a soit 0 soit 1 et ainsi de suite. ϵ n'appartient pas au langage (il n'est pas un nombre binaire)

2) L'ensemble des nombres binaires sans 0 inutiles en tête

Exemple : $L = \{0, 1, 10, 101, 100110, 111, 100000, \dots\}$

$G = \langle T, N, S, P \rangle$ où $T = \{0, 1\}$ $N = \{S, A\}$ et $P =$

$S \rightarrow 1A / 0$ /*La génération commence par 1 suivi par une suite aléatoire de 0 et 1*/

$A \rightarrow 0A / 1A / \epsilon$ /*La suite aléatoire qui se trouve après le premier 1*/

ou bien

$G = \langle T, N, S, P \rangle$ où $T = \{0, 1\}$ $N = \{S, A\}$ et $P =$

$S \rightarrow A / 0$

$A \rightarrow A0 / A1 / 1$

Remarque : Dans la première grammaire la construction des mots se fait de la gauche vers la droite tandis que dans la deuxième elle se fait de la droite vers la gauche. 0 est un cas unique qu'on peut avoir directement à partir de l'axiome.

3) L'ensemble des nombres binaires de longueur paire.

Exemple : $L = \{00, 01, 10, 11, 1001, 0010, 1111, 0000, 100101, 111111, \dots\}$

$G = \langle T, N, S, P \rangle$ où $T = \{0, 1\}$ $N = \{S, A\}$ et $P =$

$S \rightarrow 1A / 0A$ /* on génère le premier 0 ou le premier 1*/

$A \rightarrow 0S / 1S / 0 / 1$ /* soit on génère 0 ou 1 et on s'arrête. Soit on génère 0 ou 1 et on reviens à S pour continuer la génération d'un autre doublet*/

Théorie des Langages Solutions des Exercices de la Série 1

ou bien

$G = \langle T, N, S, P \rangle$ où $T = \{0, 1\}$ $N = \{S\}$ et $P =$

$S \rightarrow 00S / 01S / 11S / 10S / 00 / 01 / 11 / 10$

Remarque : Dans la première grammaire la génération se fait lettre par lettre. Dans la deuxième deux par deux, en considérant toutes les possibilités : 00, 01, 11, 10.

4) Les nombres décimaux éventuellement signés n'ayant pas de 0 inutiles. Rappelons que la partie (optionnelle) après la virgule ne se termine pas par un 0.

Exemple : $L = \{0, 2020, +123.45, -67.89, 0.98, +0.14, 987.65, +4321, \dots\}$

$G = \langle T, N, S, P \rangle$ où $T = \{0, \dots, 9\}$ $N = \{S, A, B, D\}$ et $P =$

$S \rightarrow A / +A / -A$ /* Les nombres peuvent être positif, négatifs ou sans signe*/

$A \rightarrow 1B / 2B / \dots / 9B / 0 / 0.D$ /* La partie entière qui ne commence pas par 0*/

$B \rightarrow 0B / 1B / 2B / \dots / 9B / .D / \epsilon$ /* soit on s'arrête, soit on passe à la partie décimale */

$D \rightarrow 0D / 1D / \dots / 9D / 1 / \dots / 9$ /* La partie décimale qui ne se termine pas par 0 */

5) L'ensemble des noms de variables (identificateurs) en Java. Un nom de variable en Java commence par une lettre alphabétique ou le caractère underscore (`_`) suivi par une suite quelconque de lettres alphabétiques, de chiffres et l'underscore.

Exemple : $L = \{a, \text{nom}, \text{nom_pere}, p1, p2, _prix, _16_04_2020, \text{thl_2020}, \dots\}$

Remarque : Chaque mot est composé de deux parties : début (premier symbole) qui peut être une lettre ou le underscore, et le reste du nom qui est une suite de lettres, de chiffres et de l'underscore.

$G = \langle T, N, S, P \rangle$ où $T = \{a, \dots, z, A, \dots, Z, 0, \dots, 9, _ \}$,

$N = \{ \langle \text{NomJava} \rangle, \langle \text{Suite} \rangle, \langle \text{Lettre} \rangle, \langle \text{Chiffre} \rangle \}$ et P est :

$\langle \text{NomJava} \rangle \rightarrow \langle \text{Lettre} \rangle \langle \text{Suite} \rangle / _ \langle \text{Suite} \rangle$

/*commence par une lettre alphabétique ou underscore suivi d'une suite aléatoire formée de lettres, chiffres et/ou underscore*/

$\langle \text{Suite} \rangle \rightarrow \langle \text{Lettre} \rangle \langle \text{Suite} \rangle / \langle \text{Chiffre} \rangle \langle \text{Suite} \rangle / _ \langle \text{Suite} \rangle / \epsilon$

/*une suite est une lettre alphabétique ou un chiffre ou un underscore suivi d'une suite. Minimum un mot vide*/

Théorie des Langages Solutions des Exercices de la Série 1

<Lettre> → a /... / z / A /... / Z /* les différentes possibilités d'une lettre */

<Chiffre> → 0 /... / 9 /* les différentes possibilités d'un chiffre */

6) L'ensemble des tableaux de caractères alphabétiques. Un tableau commence par { et se termine par } et les caractères sont séparés par une virgule. Chaque caractère est compris entre deux quotes simples. Le tableau peut être vide.

Exemple : L= { {}, {'a'}, {'b','c','d'}, ... }

G=<T, N, S, P> où T = {a, ..., z, A, ..., Z, ', \{, \}, \, }

N={<Tableau>, <SuiteCar>, <Car>} et S= <Tableau>, P=

<Tableau> → { <SuiteCar> } / { } /* on génère les accolades et au milieu, soit une suite de caractères, soit le vide*/

<SuiteCar> → '<Car>', <SuiteCar> / 'Car' /* Une suite de caractères est un caractère entre deux quotes suivi d'une suite de caractères. Minimum un caractère */

<Car> → a / b / ... / z / A / ... / Z /* les différentes possibilités d'un caractère */