

SERIE DE TD N°:1 COMPILATION

ANALYSE LEXICALE

Exercice 1

Proposer des définitions régulières et des automates à états finis correspondant aux unités lexicales suivantes :

1. des nombres entiers ou réels non signés tels que 2013, 12.33, 314.3E-2, 0.314E+1, 0.314E2, 314E-2 dans lesquels la partie décimale, si elle existe, commence par le point décimal, qui doit être suivi par un chiffre au minimum. La partie exposant, si elle existe, commence par un E, qui est suivi ou non par un signe (+ ou -), puis un chiffre au minimum. Le nombre de chiffres dans les parties entière, décimale et exposant n'est pas limité.
2. des identificateurs qui ne dépassent pas 4 caractères (lettres ou chiffres) dont le premier est obligatoirement une lettre.
3. des nombres entiers ou réels non signés écrits en utilisant une notation scientifique sans point décimal. La partie exposant, si elle existe, commence par un E qui est obligatoirement suivi par un signe (+ ou -), puis un chiffre au minimum. Le nombre de chiffres est limité à 3 pour les entiers. Pour les réels, il y a au maximum 3 chiffres avant le E et 2 chiffres après. Exemples : 257, 36E+02, 245E-3
4. des noms de procédures et de fonctions qui commencent obligatoirement par les 4 lettres **P R O C** pour les procédures et **F O N C** pour les fonctions. Ces 4 premières lettres peuvent être en majuscules ou en minuscules et sont suivies par 7 autres caractères au maximum dont les 4 premiers sont des lettres (au minimum deux lettres, la première étant obligatoirement une majuscule) et les 3 restants sont des chiffres (les chiffres ne sont pas obligatoires). Exemples : *ProcTest10, foncMin, FONCFc2, PROCPr36, procRech*

Exercice 2

Soit un automate d'états finis dont l'ensemble des états est $\{0, 1, 2, 3, 4\}$, état initial: 0, états finaux: 0 et 3, le vocabulaire étant $\{a, b\}$. Notons T la fonction de transition de cet automate.

$T(0, a) = \{2\}$, $T(0, b) = \{1\}$, $T(1, a) = \{2\}$, $T(1, b) = \{1\}$, $T(2, a) = \{3\}$, $T(2, b) = \{2\}$,

$T(3, a) = \{2, 4\}$, $T(3, b) = \{1, 3\}$, $T(4, a) = \{4\}$, $T(4, b) = \{4\}$

1. Etablir la table de transition et la représentation graphique de l'automate.
2. Etablir la table de transition et la représentation graphique de l'automate déterministe équivalent.
3. Donner le chemin de reconnaissance des chaînes *bababb* et *abb* par l'automate déterministe obtenu.
4. Donner le chemin de reconnaissance de la chaîne *aaab* par l'automate initial puis proposer une minimisation de cet automate qui n'affecte pas le langage qu'il reconnaît.

Exercice 3

Soit un langage, représentant un petit sous-ensemble du langage Pascal, et constitué par les unités lexicales suivantes :

- Des identificateurs commençant par une lettre suivie d'une combinaison quelconque de lettres ou de chiffres
- Des constantes numériques entières non signées (sans limitation de longueur)
- L'affectation (:=)

Construire l'analyseur lexical correspondant à ce langage en utilisant l'approche basée sur les automates finis.

Exercice 4

Soient les expressions régulières suivantes: *ab*, *ab⁺* et *(a|b)^{*}*

Appliquer les étapes 2 à 5 de la démarche de construction d'un analyseur lexical basée sur les automates d'états finis, afin de construire un analyseur lexical pour l'ensemble des trois expressions régulières précédentes, correspondant à des unités lexicales.

Exercice 5

Soient les expressions régulières suivantes: *a*, *abb*, *a^{*}b⁺*

1). Appliquer les étapes 2 à 5 de la démarche de construction d'un analyseur lexical basée sur les automates d'états finis, afin de construire un analyseur lexical pour l'ensemble des trois expressions régulières précédentes, correspondant à des unités lexicales.

2). Décrire l'analyse lexicale des chaînes *aaba* et *abb* en donnant leur chemin de reconnaissance (utiliser l'algorithme de simulation d'un AFD qui représente la 6^{ème} étape de construction de l'analyseur).