UNIVERSITÉ DE STRASBOURG

Janvier 2009

# Examen de Théorie des Langages
# Licence Informatique 3e année

*durée : 3 heures*
*documents de cours et TD autorisés*
*aucun appareil électronique n'est autorisé*

*Les 4 parties sont indépendantes. Le soin apporté à la copie sera pris en compte dans la notation.*

**Partie I** *(4 points)*

Montrez la rationalité ou non-rationalité des langages suivants sur $\Sigma = \{a, b\}$ .

a) $L1 = \{ w \in \Sigma^\star \mid w \text{ contient autant de fois la sous-chaîne } aaa \text{ que la sous-chaîne } bbb \}$

b) $L2 = \{ w \in \Sigma^\star \mid w \text{ contient autant de fois la sous-chaîne } ab \text{ que la sous-chaîne } ba \}$

**Partie II** *(3 points)*

Soit $L = \{ w \in \{a, b\}^\star \mid w \text{ commence par } aa \text{ ou termine par } aa \}$ (le 'ou' n'est pas exclusif, un mot peut commencer et terminer par *aa*).

a) Proposez une expression régulière *E* telle que *L(E) = L*
b) A partir de E, construisez un AFND *A* reconnaissant L
c) Convertissez *A* en AFD

**Partie III** *(4 points)*

Pour chacun de ces deux langages sur $\Sigma = \{a, b\}$ , proposez une grammaire et un automate à pile le reconnaissant :

a) $L1 = \{ a^n b a^{3n} \mid n > 0 \}$
b) *L2* est le langage des mots contenant plus de *a* que de *b*.

**Partie IV** *(9 points)*

**Sous-partie 1**

a) Proposez l'exemple d'un langage rationnel sur $\Sigma = \{a, b\}$ contenant quelques mots de la forme $ww^R$, celui-ci devra également contenir des mots qui ne sont pas de cette forme. Donnez un automate $(K, \Sigma, \Delta, s, F)$ reconnaissant ce langage.

b) Soit un AFND $M = (K, \Sigma, \Delta, s, F)$, on souhaite construire un automate qui reconnaît les mots $w$ tel que $ww^R$ soit reconnu par *M*. Décrivez un tel automate $M' = (K', \Sigma, \Delta', s', F')$. *M'* simulera le comportement de *M* en déclenchant dans *M*, à la lecture d'un caractère, une transition en avant (en commençant par l'état initial de *M*) et une transition en arrière (démarrant par les états finaux). Intuitivement, une transition en arrière « remonte » les flèches de l'automate *M*.

*Indication :* $K' = K \times P(K)$ *avec P(K) l'ensemble des parties de K.*

**Sous-partie 2**

Pour un mot *w* de longueur paire, on note *moitié*(*w*) la première moitié de *w*. Pour un langage *L*, on note :

$$moitié(L) = \{ moitié(w) \mid w \in L \text{ et } |w| \text{ est pair} \}$$

c) Si *L* est un langage rationnel, *moitié*(*L*) est-il rationnel ? Si tel est le cas, montrez comment construire un automate reconnaissant *moitié*(*L*) à partir d'un automate reconnaissant *L*. Dans le cas contraire présentez un exemple simple de langage *L* tel que *moitié(L)* n'est pas rationnel.

d) Montrez que si *L* est rationnel, *moitié(L)* est algébrique. Produisez un automate à pile à partir d'un automate reconnaissant *L*. Attention : dire que *moitié(L)* est algébrique ne signifie pas que ce langage n'est pas rationnel (la question c. garde donc tout son sens).
Dans le cas où la question c. a montré la rationalité de *moitié(L)*, l'automate à pile ne devra pas simplement reprendre l'automate de cette question mais utiliser au mieux la structure de pile.