

Master Ingénierie Informatique. Automates avancés.
Travaux dirigés

Expressions rationnelles
(semaine du 08/2/05)

Exercice 1.1 On veut détecter les commentaires se trouvant dans des programmes Java et C. Les commentaires Java commencent par `//` et se terminent en fin de ligne. On s'intéresse aux commentaires C compris entre `/*` et `*/`. Décrire des expressions rationnelles pour les commentaires

- de programmes Java ne contenant pas de chaînes de caractères.
- de programmes C ne contenant pas de chaînes de caractères.
- de programmes Java pouvant contenir des chaînes de caractères.
- de programmes C pouvant contenir des chaînes de caractères.

Exercice 1.2 Trouver une expression rationnelle décrivant les chaînes de caractères du langage C. On rappelle que pour insérer le caractère `'` dans une chaîne, on doit le faire précéder du caractère `'\'`.

Exercice 1.3 Simplifier les expressions rationnelles suivantes.

- $a^*b^* \cup b^*$,
- $((aba)^*)^*$,
- $(a^*b)^*a^*$.

Exercice 1.4 Trouver des expressions rationnelles pour les langages de mots suivants, sur l'alphabet $A = \{a, b\}$.

- ensemble des mots de longueur paire.
- ensemble des mots commençant par a et finissant par b .
- ensemble des mots commençant et finissant par aa .
- ensemble des mots ayant au moins deux occurrences du facteur aba .

Exercice 1.5 Trouver des expressions utilisant produit, union, étoile, intersection et complément pour langages de mots suivants, sur l'alphabet $A = \{a, b\}$. Trouver ensuite des expressions rationnelles correspondantes.

- ensemble des mots ne contenant pas le facteur abb .
- ensemble des mots contenant une unique occurrence du facteur ab .
- ensemble des mots contenant une unique occurrence du facteur bab .

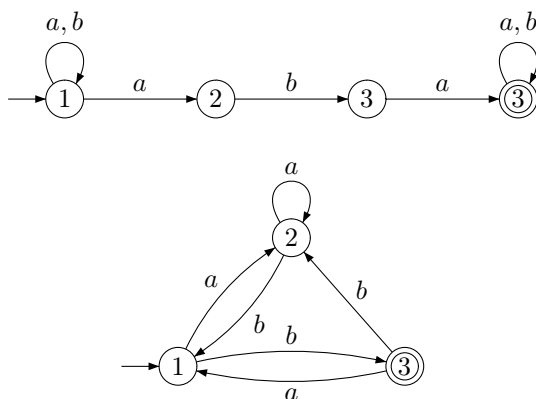
Automates et langages reconnaissables

(semaine du 15/2/05)

Exercice 2.1 Construire des automates associés aux expressions des deux premiers points de l'exercice 1.1, et à l'expression de l'exercice 1.2.

Exercice 2.2 Écrire un automate déterministe reconnaissant le langage des mots sur l'alphabet $\{0, 1\}$ qui représentent les entiers en base 2 dont le reste modulo 3 est 1 ou 2.

Exercice 2.3 Calculer, pour chacun des automates suivants, une expression régulière décrivant le même langage. Que donne l'algorithme de McNaughton-Yamada sur le deuxième automate ?



Exercice 2.4 Écrire un automate sur $A = \{a, b, c\}$ reconnaissant les mots dans lesquels la chaîne abc apparaît un nombre impair de fois.

Exercice 2.5 À partir d'un automate $\mathcal{A} = (Q, A, E, I, F)$, construire un automate émondé reconnaissant le même langage. Construire ensuite un automate complet dont tous les états sont accessibles à partir d'un état initial et ayant au plus un état à partir duquel on ne peut pas atteindre un état final.

Exercice 2.6 A partir d'un automate \mathcal{A} , construire une grammaire linéaire qui engendre $L(\mathcal{A})$.

Exercice 2.7 Montrer que si L est reconnaissable, alors $\{w \in L \mid |w| \text{ est pair}\}$ est aussi reconnaissable.

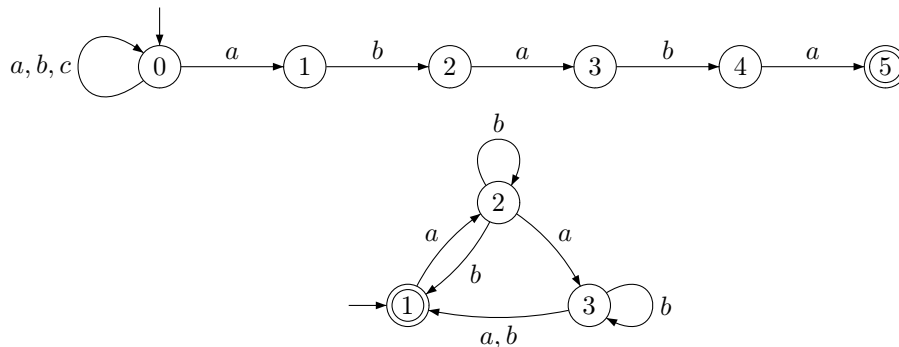
Exercice 2.8 Montrer que le langage $\{a^n b^n \mid n \in \mathbb{N}\}$ n'est pas reconnaissable. Même question pour le langage $\{a^{2^n} \mid n \in \mathbb{N}\}$. Même question pour le langage des mots sur l'alphabet $\{0, 1\}$ qui représentent un entier premier (en base 2). Même question pour l'ensemble des palindromes sur un alphabet a au moins deux lettres.

Exercice 2.9 Pour $K, L \subseteq A^*$, on note $K^{-1}L = \{u \in A^* \mid \exists v \in K, vu \in L\}$. Montrer que si K, L sont reconnaissables, alors $K^{-1}L$ est reconnaissable. En déduire que si K est reconnaissable, il en est de même de $\text{Pref}(K)$ l'ensemble des préfixes des mots de K . Comment construire un automate pour $\text{Pref}(K)$ à partir d'un automate pour K ?

Déterminisation, traductions entre automates et expressions

(semaine du 22/2/05)

Exercice 3.1 Déterminiser les automates suivants.



Exercice 3.2 Trouver un automate non-déterministe correspondant reconnaissant $ab^* \cup aac(a)^*(ab^*a)^*$ en utilisant l'algorithme de Glushkov. Même question pour l'expression $(a(ab)^*)^* + (ba)^*$.

Exercice 3.3 Soit $K, L \subseteq A^*$ donnés et X satisfaisant l'équation $X = KX + L$. On suppose que le mot vide est dans K . Montrer que les solutions sont les $X = K^*M$ avec $M \supseteq L$.

Exercice 3.4 Soit \mathcal{A} un automate dont les transitions sont étiquetées sur $A \cup \{\varepsilon\}$. Construire un automate équivalent ayant le même nombre d'états que \mathcal{A} .

Exercice 3.5 Soit $A = \{a_1, \dots, a_n\}$ et $\# \notin A$. Soit

$$L_1 = \{w\#a \mid w \in A^* \text{ et } a \text{ apparaît dans } w\}$$

$$L_2 = \{w\#a \mid w \in A^* \text{ et } a \text{ n'apparaît pas dans } w\}$$

Décrire des automates déterministes et non-déterministes pour L_1 et L_2 . Y a-t-il un automate non-déterministe de taille $O(n)$ pour L_1 ? pour L_2 ?

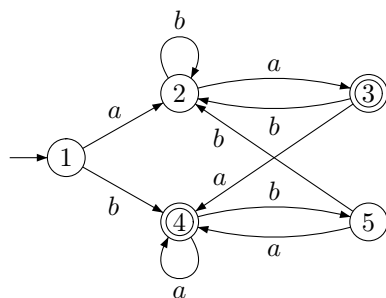
Exercice 3.6 Trouver les automates minimaux des langages de l'exercice 3.2.

Exercice 3.7 Montrer de plusieurs façons que l'automate minimal du langage A^*x , où x est un mot, a exactement $|x| + 1$ états. Calculer l'automate minimal de $A^*\text{ananas}$, où $A = \{a, \dots, z\}$.

Exercice 4.1 (Algorithme de Brzozowski) Soit $\mathcal{A} = (Q, A, E, I, F)$ un automate. On définit l'automate renversé de \mathcal{A} par $\mathcal{A}^r = (Q, A, E^r, I^r, F^r)$ où

- $(p, a, q) \in E^r$ si et seulement si $(q, a, p) \in E$,
- $I^r = F$,
- $F^r = I$.

1. Montrer que si \mathcal{A} est déterministe accessible, alors l'automate déterministe accessible de \mathcal{A}^r est minimal.
2. En déduire une procédure utilisant uniquement la déterminisation et le renversement pour minimiser un automate. Cet algorithme fonctionne-t-il aussi à partir d'un automate non déterministe ?
3. Donner une borne supérieure pour la complexité de l'algorithme.
4. Trouver l'automate minimal équivalent à l'automate suivant.



Exercice 4.2 Comparer les algorithmes de Moore et celui de Hopcroft pour l'automate suivant.

