Université de Montpellier HLIN502 juin 2019

Numéro d'anonymat :

Durée : 2 heures

Examen de langages formels (deuxième session)

Seule, une feuille A4 recto-verso est autorisée

Interdiction de communiquer tout document.

REMPLIR LES CADRES ET RENDRE CE DOCUMENT AINSI COMPLÉTÉ

UN EXCÈS DE REPONSES FAUSSES SERA SANCTIONNÉ PAR DES POINTS NÉGATIFS

TOUTES LES PROPRIETES PRESENTEES EN COURS POURRONT ETRE UTILISEES

Exercice 1 :

Soit la grammaire G d'axiome S, d'alphabet $\Sigma=\{a,b\}$ et de productions : S → bb | aS | bS | Sa | Sb

1) Prouver que $\Pi(n)=\left(S \xrightarrow{\leq n} m \Rightarrow \exists m_1 \in \Sigma^*, \exists m_2 \in \Sigma^* \text{ tel que } S \xrightarrow{*} m_1 bb m_2\right)$ est vraie pour tout entier $n \geq 1$.

Π(1) :

$S \xrightarrow{1} m \Rightarrow S \rightarrow bb \Rightarrow \exists m_1 = m_2 = \epsilon$ tel que $m = m_1 . bb . m_2$

Hypothèse : Π(n) vrai.

Montrons Π(n+1) où $n \geq 1$:

$$S \xrightarrow{n+1} m \Rightarrow \left(\begin{array}{l} S \xrightarrow{n+1} bb \\ ou \\ S \xrightarrow{1} aS \xrightarrow{n} m \\ ou \\ S \xrightarrow{1} bS \xrightarrow{n} m \\ ou \\ S \xrightarrow{1} Sa \xrightarrow{n} m \\ ou \\ S \xrightarrow{1} Sb \xrightarrow{n} m \end{array} \right)$$

Le premier cas est contradictoire avec $n \geq 1$.

Dans le deuxième cas,

$$\begin{aligned} S \xrightarrow{1} aS \xrightarrow{n} m & \underset{L.F.}{\Rightarrow} S \xrightarrow{1} aS \xrightarrow{n} a.m' \text{ et } S \xrightarrow{n} m' \text{ et } m = a.m' \\ & \underset{H.R.}{\Rightarrow} \exists m_1, m_2 \text{ tel que } m' = m_1 bb m_2 \\ & \Rightarrow \exists m_1, m_2 \text{ tel que } m = a.m' = (a.m_1) bb m_2 \end{aligned}$$

Les trois autres cas sont similaires. Faisons en juste un :

$$\begin{aligned} S \xrightarrow{1} Sb \xrightarrow{n} m & \underset{L.F.}{\Rightarrow} S \xrightarrow{1} Sb \xrightarrow{n} m'.b \text{ et } S \xrightarrow{n} m' \text{ et } m = m'.b \\ & \underset{H.R.}{\Rightarrow} \exists m_1, m_2 \text{ tel que } m' = m_1 bb m_2 \\ & \Rightarrow \exists m_1, m_2 \text{ tel que } m = m'.b = m_1 bb (m_2.b) \end{aligned}$$

2) Prouver que :

$$\forall\, p \in \Sigma^*, \text{ il existe une chaîne de dérivation } S \overset{*}{\rightarrow} pS \ .$$

L'induction portera sur le nombre n de lettres de p, et on explicitera obligatoirement la propriété Π(n) à prouver.
Rappelons que, par convention, on pourra écrire : $S \overset{0}{\rightarrow} S$

$\Pi(n) = \left(|p| \leq n \quad \Rightarrow \quad S \overset{*}{\rightarrow} p.S\right)$

Π(0) : |p| ≤ 0 ⇒ p = ε . Dans ce cas, on a : $S \overset{0}{\rightarrow} S = \varepsilon.S$

Hypothèse : Π(n) vrai. Montrons Π(n+1) où $n \geq 0$:

Soit p un mot de n+1 lettres. Ce mot ne peut être le mot vide. Notons c sa première lettre.

$p = c.p'$ avec $c \in \Sigma$

Par hypothèse de récurrence appliquée à p' :

$S \overset{*}{\rightarrow} p'S$

On en déduit alors la chaîne de dérivation :

$S \overset{1}{\rightarrow} cS \overset{*}{\rightarrow} c\,p'S = p.S$

La première dérivation S → cS découle directement de la grammaire pour la lettre c valant a ou b

3) En admettant la propriété suivante :

$$\forall\, q \in \Sigma^*, \text{ il existe une chaîne de dérivation } S \overset{*}{\rightarrow} Sq \ ,$$

prouver que le langage associé la grammaire G est l'ensemble des mots sur l'alphabet {a,b} qui contiennent le facteur bb.

On note L_G le langage associé à la grammaire G.
Il faut prouver que tous les mots de L_G contiennent le facteur bb. C'est ce qui a été fait à la première question de l'exercice.
Inversement, il faut prouver que si un mot contient le facteur bb, il est dans le langage L_G :

Si m contient le facteur bb, alors il existe p et q tel que m=p.bb.q
On peut alors, en utilisant les résultats précédents prouvé ou admis, construire la chaîne de dérivation

$S \overset{*}{\rightarrow} pS \overset{*}{\rightarrow} p\,S\,q \overset{1}{\rightarrow} p\,bb\,q = m$

ce qui prouve que $m \in L_G$ et termine la preuve demandée.

Exercice 2 :

1) Justifier que tout mot sur l'alphabet {a,b} sans facteur bb est de la forme :

$$a^{n_1} b a^{n_2} b \cdots a^{n_{k-1}} b a^{n_k} \quad \text{avec} \quad k \geq 1, n_1 \geq 0, n_k \geq 0 \quad \text{et} \quad \forall i \in [2, \ldots k-1], k_i \geq 1 \quad .$$

ou

$$a^{n_1} \quad \text{avec} \quad n_1 \geq 0$$

La clarté de la justification sera essentielle.

Tout mot m sur l'alphabet {a,b} est une suite de a et de b. Si on explicite les occurrences de b dans m, on a :
$m = a \cdots a \ b^{n_1} \ a \cdots a \ b^{n_2} \ a \cdots a \quad \ldots \quad b^{n_k} \ a \cdots a$ avec $n_i \geq 1$ pour tout i.
Le cas limite où il n'y a aucun b mérite d'être explicité plutôt que considéré comme le cas où $k=0$.
Dans ce cas, le mot n'est constitué que de a. et correspond à la seconde forme a^{n_1} avec $n_1 \geq 0$

Si le mot ne contient pas le facteur bb, cela impose que tous les n_i valent au plus 1.
$m = a \cdots a \ b \ a \cdots a \ b \ a \cdots a \quad \ldots \quad b \ a \cdots a$
Pour que le mot m ne contienne pas le facteur bb, il faut encore que les séquences $a \cdots a$ ne soient pas vides, sauf la première et dernière séquence $a \cdots a$
Cela correspond à la première partie de la forme demandée.

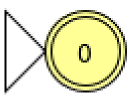
2) En déduire une expression rationnelle e telle que le langage sur l'alphabet {a,b} des mots sans facteur bb soit le langage L(e).

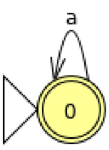
Il y a plusieurs solutions possibles. Par exemple :

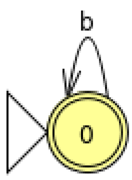
$e = a^* + a^* \ b \ (a^+ b)^* a^*$
ou
$e = a^* + a^* \ (b a^+)^* b a^*$

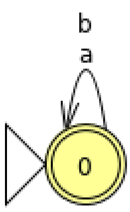
Exercice 3 :

1) Quels sont tous les automates déterministes sur l'alphabet Σ = {a,b} qui n'ont qu'un seul état, cet état étant terminal. Donner pour chacun d'eux leur langage associé.



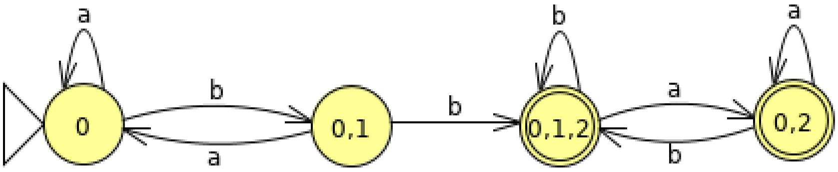






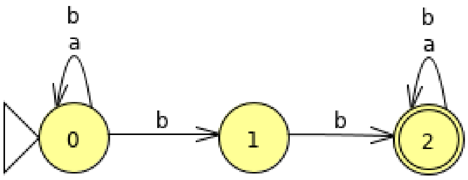
$L = \{\varepsilon\}$ $\quad L = \{a^n, n \in \mathbb{N}\}$ $\quad L = \{b^n, n \in \mathbb{N}\}$ $\quad L = \Sigma^*$

2a) Trouver un automate indéterministe à 3 états 0,1 et 2 reconnaissant les mots sur l'alphabet {a,b} ayant au moins un facteur bb ; et de sorte que sa déterminisation donne l'automate A_0 suivant :



Les noms des états devront être cohérents

En prenant comme automate indéterministe :



sa déterminisation donne bien l'automate A_0

2b) Dans l'automate déterminisé A_0 de la question précédente, trouver deux états e et e' tel que :

$$\forall m \in \Sigma^*, \delta^*(e,m) \in F$$

où δ est la fonction de transition de l'automate déterminisé, et où on notera F l'ensemble de ses états terminaux. Prouver par induction que la propriété $\forall m \in \Sigma^*, \delta^*(e,m) \in F$ est vraie, en prenant pour e au choix un des 2 états.

Les deux états sont les états (0,1,2) et (0,2) qui sont tous les deux des états terminaux.
Prouvons la propriété pour tout n suivante (peu importe le choix de e ou e') :

$$\Pi(n) = \left(|m| \le n \quad \Rightarrow \quad \delta^*(e,m) \in F \right)$$

Π(0) est vrai car $|m| \le 0 \quad \Rightarrow \quad m = \epsilon \quad \Rightarrow \quad \delta^*(e,m) = e \in F$

Hypothèse : Π(n) vraie.

Montrons que Π(n+1) vraie, pour n ≥ 0

Soit m un mot de n+1 lettres. Alors m = m' α avec α une lettre. Et

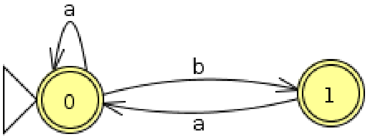
$$\delta^*(e,m) = \delta^*(e,m'\alpha) = \delta(\delta^*(e,m'),\alpha) \underset{H.R.}{\in} \delta(F,\alpha)$$

Et on observera sur l'automate que F={e,e'} et $\delta(e,\alpha) \in F$ et $\delta(e',\alpha) \in F$

De sorte que $\delta(F,\alpha) \subseteq F$ ce qui termine la preuve.

Remarque : le lecteur pourra en déduire que les deux états e et e' sont fusionnables et un déduire un automate équivalent à A_0 plus simple.

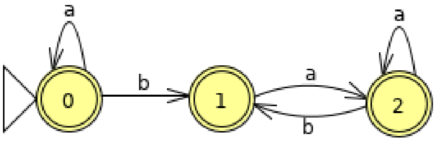
3) Soit l'automate A_1 suivant :



Expliquer, à partir des résultats précédents, pourquoi cet automate reconnaît exactement les mots n'ayant pas de facteur bb,

L'automate déterministe A_0 est complet. Il reconnaît les mots ayant un facteur bb.
Son automate complémentaire s'obtient en remplaçant les états terminaux par des états non terminaux et inversement. Et en supprimant l'état non co-accessible, on obtient alors exactement l'automate A_1, qui reconnaît le complémentaire des mots reconnus par A_0.

4) Soit l'automate A_2 :



4a) Montrer que $\forall n \geq 1,\ \delta^*(1, a^n b) = 1$ où δ est sa fonction de transition.

$\forall n \geq 1,\ \delta^*(1, a^n b) = \delta^*(\delta(1,a), a^{n-1} b) = \delta^*(2, a^{n-1} b) = \delta(\delta^*(2, a^{n-1}), b)$

Et il est « clair » que $\delta^*(2, a^{n-1}) = 2$ pour tout n ≥ 1. Pour le montrer, on peut faire un raisonnement par induction sur n quelque peu trivial avec $\Pi(n) = \left(\delta^*(2, a^{n-1}) = 2\right)$

Π(1) : $\delta^*(2, a^0) = \delta^*(2, \epsilon) = 2$

$\delta^*(2, a^{n-1}) = \delta^*(\delta(2,a), a^{n-2}) = \delta(2, a^{n-2}) \underset{H.R.}{=} 2$

En revenant au premier calcul, on obtient :

$\forall n \geq 1,\ \delta^*(1, a^n b) = \delta(\delta^*(2, a^{n-1}), b) = \delta(2, b) = 1$

4b) Prouver que $\delta^*(1, (a^{k_1} b)(a^{k_2} b)\ldots(a^{k_n} b)) = 1$ avec $\forall i, k_i \geq 1$ par une induction sur n

$\Pi(n) = \left(\delta^*(1, (a^{k_1} b)(a^{k_2} b)\ldots(a^{k_n} b)) = 1\right)$

Π(1) est vrai car cela a été montré à la question précédente.
Hypothèse : Π(n) vrai.
Montrons que Π(n+1) vrai pour n ≥ 1 :

$\delta^*(1, (a^{k_1} b)(a^{k_2} b)\ldots(a^{k_{n+1}} b)) = \delta^*(\delta^*(1, (a^{k_1} b)(a^{k_2} b)\ldots(a^{k_n} b)), (a^{k_{n+1}} b)) \underset{H.R.}{=} \delta^*(1, (a^{k_{n+1}} b)) = 1$

CQFD.

4c) En déduire ce que vaut $\delta^*(0,m)$ en fonction de m un mot sur l'alphabet {a,b} qui est sans facteur bb. On traitera avec soin les cas limites, afin d'obtenir les états 0, 1 ou 2 selon les cas. Se souvenir de l'exercice 2.

Pour $m \neq \epsilon$, on a $m = a^{n_1} b a^{n_2} b \cdots a^{n_{k-1}} b a^{n_k}$ avec $k \geq 1, n_1 \geq 0, n_k \geq 0$ et $\forall i \in [2, \dots k-1], k_i \geq 1$

Calculons dans ce cas $\delta^*(0,m)$:

$$\begin{aligned}
\delta^*(0, a^{n_1} b a^{n_2} b \cdots a^{n_{k-1}} b a^{n_k}) &= \delta^*(\delta^*(0, a^{n_1}), b a^{n_2} b \cdots a^{n_{k-1}} b a^{n_k}) \qquad \text{cela suppose que } k \geq 2 \\
&= \delta^*(0, b a^{n_2} b \cdots a^{n_{k-1}} b a^{n_k}) \\
&= \delta^*(\delta(0,b), a^{n_2} b \cdots a^{n_{k-1}} b a^{n_k}) \\
&= \delta^*(1, a^{n_2} b \cdots a^{n_{k-1}} b a^{n_k}) \\
&= \delta^*(\delta^*(1, a^{n_2} b \cdots a^{n_{k-1}} b), a^{n_k}) \\
&= \delta^*(1, a^{n_k}) \\
&= \begin{cases} 1 & \text{si } n_k = 0 \\ 2 & \text{sinon} \end{cases}
\end{aligned}$$

Il faut traiter le cas où k=1 : $\delta^*(0, a^{n_1}) = 0$

Pour $m = \epsilon$, on a : $\delta^*(0, \epsilon) = 0$

On remarquera que les 3 états sont terminaux. Tous les mots sans facteur bb sont par conséquent reconnus par l'automate.

4d) Montrer que si le mot m contient le facteur bb, i.e. si $m = m_1 b b m_2$, alors $\delta^*(0,m)$ n'existe pas.

$$\delta^*(0,m) = \delta^*(0, m_1 b b m_2) = \delta^*(\delta^*(\delta^*(0, m_1), bb), m_2)$$

En notant e l'état $\delta^*(0, m_1)$ l'expression précédente pour exister doit pouvoir calculer $\delta^*(e, bb)$.
Or, si on regarde l'automate A_2 , on constate qu'il n'existe aucun état dont partent une transition b puis une autre transition b. Donc $\delta^*(e, bb)$ n'existe pas tout comme $\delta^*(0,m)$

5) Justifier que les automates A_1 et A_2 reconnaissent le même langage. Et, si on minimalise l'automate A_2 , quel automate obtient -on et pourquoi ?

L'automate A_1 et A_2 reconnaissent le même langage : les mots qui sont sans facteur bb.
On a montré que c'était le cas pour A_1 à la question 3.
On a montré que c'était le cas pour A_2 à la question 4c qui a montré que si m sans facteur bb alors il est reconnu par l'automate ; Et à la question 4d on a montré l'inverse par un raisonnement par contraposé.
Remarque : ici comme tous les états sont terminaux, l'existence de $\delta^*(0,m)$ est équivalent à dire que m est reconnu par l'automate.

De plus l'automate A_1 est déterministe et minimal car, avec 2 états, un automate plus petit aurait un seul état. Or, la première question de cet exercice a montré que l'on ne pouvait obtenir un tel automate.

La minimalisation de l'automate A_2 donne donc l'automate A_1 .