TRAVAIL PRATIQUE 2
Un bref solutionnaire

IFT-2002 : Informatique théorique   H 2014 (Julien Marcil)

Question 1 (20 + 5 points)

Soit $\Sigma = \{0, 1\}$.

(a) Soit le langage $A = \{xy \mid x, y \in \Sigma^* \text{ et } x = y\}$. Démontrez que A n'est pas un langage hors contexte.

(b) Soit le langage $B = \{xy \mid x, y \in \Sigma^* \text{ et } |x| = |y| \text{ mais } x \neq y\}$. Démontrez que B est un langage hors contexte.

Solution.

(a) Nous allons utiliser le lemme de pompage pour faire une preuve par contradiction.

DÉMONSTRATION.
Supposons que le langage A soit hors contexte. Alors, soit p la longueur de pompage de A. Soit le mot $w = 0^p1^p0^p1^p$. Clairement $w \in A$ car $w = xy$ pour $x = y = 0^p1^p$.
Par le lemme de pompage des langages hors contexte, puisque $|w| \geq p$, il existe des mots u, v, x, y, z tels que $w = uvxyz$, $|vxy| \leq p$, $|vy| > 0$ et pour tout entier $i \geq 0$, $uv^ixy^iz \in L$.
Puisque $|vxy| \leq p$ alors vxy ne peut être soit :

1 - un mot composé uniquement de 0

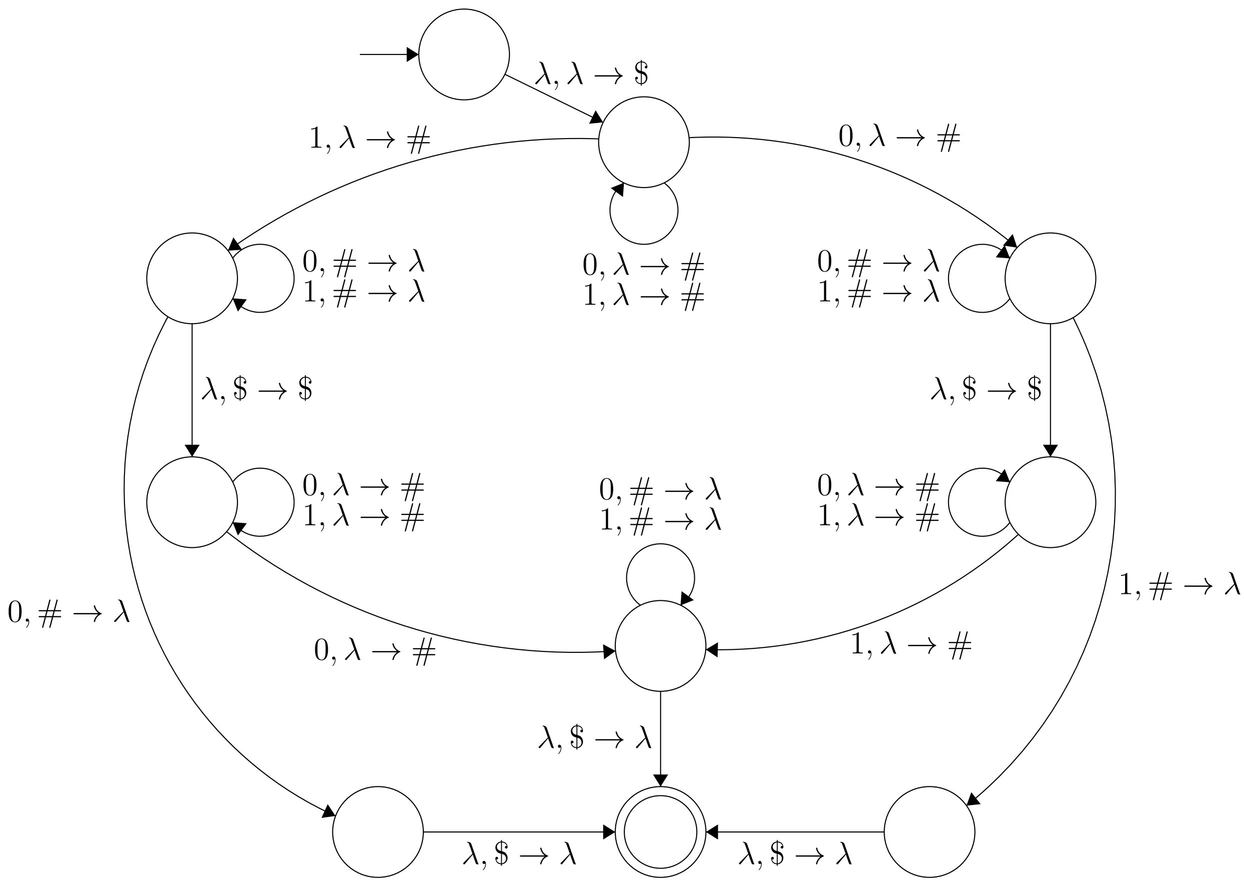
2 - un mot composé uniquement de 1

3 - un mot 0^i1^j

4 - un mot 1^j0^i

Pour le cas 1, le mot composé uniquement de 0 est soit pris dans la permière moité, ou la deuxième moité de w. Dans les deux cas, le uv^2xy^2z ne peut contenir le même nombre de 0 au début et à la fin et donc $uv^2xy^2z \notin A$. Le même raisonement s'applique pour le cas 2 avec les 1.
Pour le cas 3, si v contient des 0 alors le uv^2xy^2z ne peut contenir le même nombre de 0 au début et à la fin et donc $uv^2xy^2z \notin A$. De même si y contient des 1. Mais puisque, $|vy| > 0$ alors il n'est pas possible que v et soit v soit vide. Le même raisonement s'applique pour le cas 4.
Il est donc imposible de trouver u, v, x, y, z tels que $w = uvxyz$ qui respectent les conditions de pompages. Donc A n'est pas hors contexte.
Barême : 6 points pour un bon mot w, 7 points pour une bonne utilisation du lemme, 7 points pour la cohérence de la preuve.

(b) Voici un automate à pile qui accepte le langage B.



Voici une grammaire qui génère le langage B.

$$
\begin{aligned}
S &\to AB \mid BA \\
A &\to CAC \mid 0 \\
B &\to CBC \mid 1 \\
C &\to 0 \mid 1
\end{aligned}
$$

Barème : 5 points pour un automate à pile ou une grammaire hors contexte qui accepte B.

Question 2 (10 + 15 points)

Soit les machines de Turing $M = (S, \Sigma, \Gamma, S_0, S_{accepte})$ qui ont un seul état accepteur $S_{accepte} \in S$. Le mot produit par une machine M est celui qui se trouve sur son ruban lorsqu'elle atteint l'état $S_{accepte}$, donc lorsqu'elle s'arrête. Les machines M utilisent un ruban unique, s'étendant à l'infini à droite et à gauche.

(a) Trouvez une machine de Turing

$$M_3 = (\{S_0, S_1, S_{accepte}\}, \{a, \sqcup\}, \{a, \sqcup\}, \delta, S_0, S_{accepte})$$

qui, lorsqu'exécutée sur un ruban vide, M_3 produira un mot contenant 4 symboles a. Donnez la séquence des configurations de M_3 pour cette exécution.

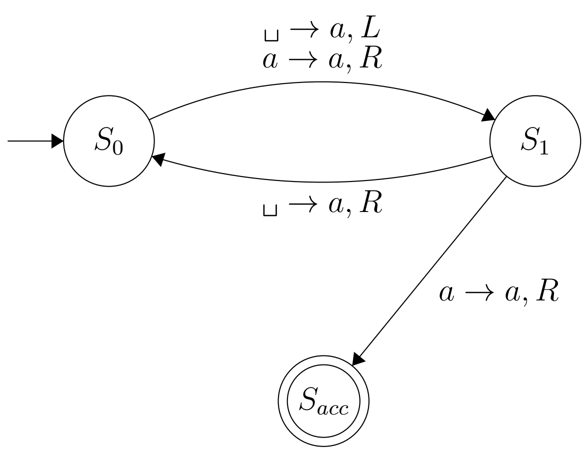
(b) Quel est le plus grand nombre de symboles a que vous arrivez à produire avec une machine

$$M_4 = (\{S_0, S_1, S_2, S_{accepte}\}, \{a, \sqcup\}, \{a, \sqcup\}, \delta, S_0, S_{accepte})$$

exécutée sur un ruban vide ? Exhibez votre machine de Turing M_4 et la séquence des configurations pour son exécution.

Solution.

(a) Voici un diagramme de transitions de la machine M_3.

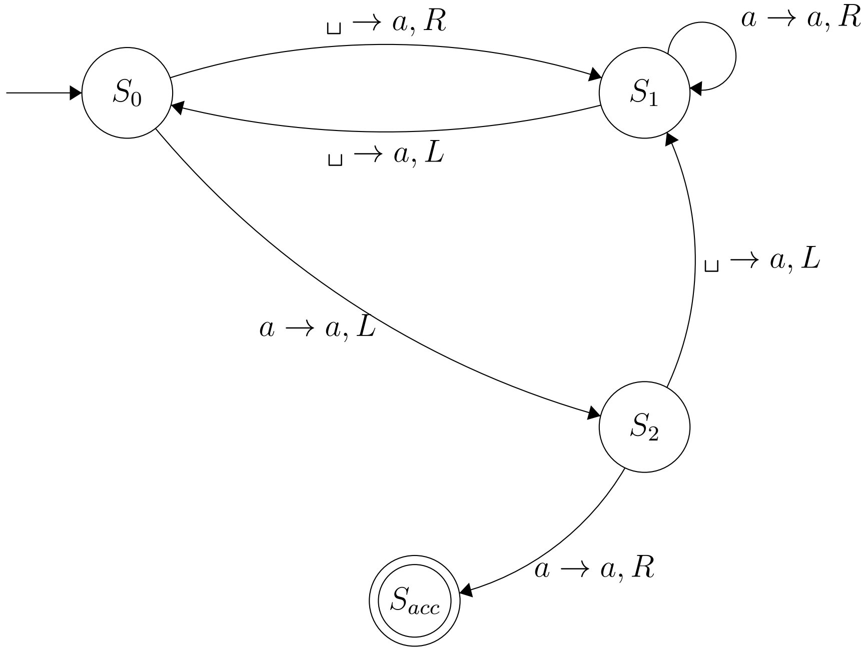


Voici la séquence des configurations pour l'éxécution de M_3 sur un ruban vide.

S_0	$\sqcup\underline{\sqcup}\sqcup\sqcup\sqcup$
S_1	$\underline{\sqcup}a\sqcup\sqcup\sqcup$
S_0	$a\underline{a}\sqcup\sqcup\sqcup$
S_1	$aa\underline{\sqcup}\sqcup\sqcup$
S_0	$aaa\underline{\sqcup}\sqcup$
S_1	$aa\underline{a}a\sqcup$
S_{acc}	$aaa\underline{a}\sqcup$

Barème : 5 points pour une bonne machine, 5 points pour la séquence des configurations

(b) Voici un diagramme de transitions de la machine M_4.



Voici la séquence des configurations pour l'éxécution de M_3 sur un ruban vide.

$$
\begin{array}{rl}
S_0 & \sqcup\sqcup\sqcup\underline{\sqcup}\sqcup\sqcup\sqcup \\
S_1 & \sqcup\sqcup\sqcup a\underline{\sqcup}\sqcup\sqcup \\
S_0 & \sqcup\sqcup\sqcup\underline{a}a\sqcup\sqcup \\
S_2 & \sqcup\sqcup\underline{\sqcup}aa\sqcup\sqcup \\
S_1 & \sqcup\underline{\sqcup}aaa\sqcup\sqcup \\
S_0 & \underline{\sqcup}aaaa\sqcup\sqcup \\
S_1 & a\underline{a}aaa\sqcup\sqcup \\
S_1 & aa\underline{a}aa\sqcup\sqcup \\
S_1 & aaa\underline{a}a\sqcup\sqcup \\
S_1 & aaaa\underline{a}\sqcup\sqcup \\
S_1 & aaaaa\underline{\sqcup}\sqcup \\
S_0 & aaaa\underline{a}a\sqcup \\
S_3 & aaa\underline{a}aa\sqcup \\
S_{acc} & aaaa\underline{a}a\sqcup
\end{array}
$$

Barème : 6 points pour une bonne machine, 6 points pour la séquence des configurations, 3 points pour une machine optimale (6 a).

Question 3 (10 + 10 + 10 points)

Les ensembles d'entiers positifs peuvent être codés facilement pour les programmes `RÉPÉTER`. L'ensemble d'entiers $S = \{i_1, i_2, \ldots, i_t\}$ est codé dans un registre r par la valeur :

$$p_{i_1} \times p_{i_2} \times \cdots \times p_{i_t}$$

où p_{i_j} correspond au i_j-ième nombre premier. Donnez les programmes `RÉPÉTER` pour les macros suivantes (vous pouvez utiliser n'importe quelles macros définies dans les diapositives du cours) :

(a) Donnez la macro $\texttt{CARD}(r_1)$ qui retourne dans r_0 la cardinalité de l'ensemble codé par r_1.

(b) Donnez la macro $\texttt{AJOUT}(r_1, r_2)$ qui retourne dans r_0 l'ensemble codé dans r_1 auquel on ajoute l'entier rangé dans r_2.

(c) Donnez la macro $\texttt{INTER}(r_1, r_2)$ qui retourne dans r_0 l'intersection des ensembles codés par r_1 et r_2.

Solution.

Nous allons premièrement définir la macro $\texttt{DANS?}(r_1, r_2)$ qui retourne $r_0 = \texttt{VRAI}$ si l'entier rangé dans r_2 est dans l'ensemble codé dans r_1. La macro retourne $r_0 = \texttt{FAUX}$ si l'entier dans r_2 n'est pas dans l'ensemble codé dans r_1.

$\texttt{DANS?}(r_1, r_2)$

```
r3 ← PREMIERK(r2)
r4 ← MOD(r1, r3)
r5 ← PG?(1, r4)
si r5 alors [
  r0 ← VRAI
]
```

(a) $\texttt{CARD}(r_1)$

```
repeter r1 fois [
  inc(r2)
  r3 ← DANS?(r1, r2)
  si r3 alors [
    inc(r0)
  ]
]
```

(b) $\texttt{AJOUT}(r_1, r_2)$

$r_0 \leftarrow r_1$
$r_3 \leftarrow \texttt{DANS?}(r_1, r_2)$
$r_3 \leftarrow \texttt{NEG}(r_3)$
$\texttt{si}$ r_3 $\texttt{alors}$ [
  $r_4 \leftarrow \texttt{PREMIERK}(r_2)$
  $r_0 \leftarrow \texttt{MULT}(r_1, r_4)$
]

(c) $\texttt{INTER}(r_1, r_2)$

$r_0 \leftarrow 1$
$r_3 \leftarrow \texttt{PG?}(r_1, r_2)$
$\texttt{si}$ r_3 $\texttt{alors}$ [
  $r_4 \leftarrow r_2$
]
$r_3 \leftarrow \texttt{NEG}(r_3)$
$\texttt{si}$ r_3 $\texttt{alors}$ [
  $r_4 \leftarrow r_1$
]
$\texttt{repeter}$ r_4 $\texttt{fois}$ [
  $\texttt{inc}(r_5)$
  $r_6 \leftarrow \texttt{DANS?}(r_1, r_5)$
  $r_7 \leftarrow \texttt{DANS?}(r_2, r_5)$
  $r_8 \leftarrow \texttt{ET}(r_6, r_7)$
  $\texttt{si}$ r_8 $\texttt{alors}$ [
    $r_0 \leftarrow \texttt{AJOUT}(r_0, r_5)$
  ]
]

Barème : 5 points pour la bonne approche, 5 points pour le bon programme RÉPÉTER.

Question 4 (10 + 10 points)

(a) Soit $S_a = \{\langle M \rangle \mid M$ est un AFD[1] tel que M accepte $w^{\mathcal{R}}$ si M accepte $w\}$. Montez que S_a est un langage décidable.

(b) Soit $S_b = \{\langle M_1, M_2 \rangle \mid M_1$ et M_2 sont des AFD[1] tels que $L(M_1) \subseteq L(M_2)\}$. Montez que S_b est un langage décidable.

Solution.

(a) $S_a \leq_m EQ_{AFD}$

Pour un langage L, nous définissons $L^{\mathcal{R}} = \{w^{\mathcal{R}} \mid w \in L\}$. Nous avons démontrez dans l'examen intra que si L est un langage régulier, alors $L^{\mathcal{R}}$ est régulier aussi. Nous savons également comment construire un automate fini déterminste $M^{\mathcal{R}}$ qui accepte $L^{\mathcal{R}}$ étant donné un automate fini détermiste M qui accepte L.

$$\begin{aligned}\langle M \rangle \in S_a &\Longrightarrow \forall_{w \in \Sigma^*}\ w \in L(M) \Rightarrow w^{\mathcal{R}} \in L(M)\\ &\Longrightarrow L(M) \subseteq L(M)^{\mathcal{R}}\\ &\Longrightarrow L(M) \subseteq L(M^{\mathcal{R}})\end{aligned}$$

et aussi

$$\begin{aligned}\langle M \rangle \in S_a &\Longrightarrow \forall_{w \in \Sigma^*}\ w^{\mathcal{R}} \in L(M) \Rightarrow (w^{\mathcal{R}})^{\mathcal{R}} \in L(M)\\ &\Longrightarrow \forall_{w \in \Sigma^*}\ w^{\mathcal{R}} \in L(M) \Rightarrow w \in L(M)\\ &\Longrightarrow L(M)^{\mathcal{R}} \subseteq L(M)\\ &\Longrightarrow L(M^{\mathcal{R}}) \subseteq L(M)\end{aligned}$$

Donc

$$\langle M \rangle \in S_a \Longleftrightarrow L(M) = L(M^{\mathcal{R}})$$

Pour la réduction $S_a \leq_m EQ_{AFD}$ il nous faut donc un fonction f tel que

$$\begin{aligned}f : S_a &\to EQ_{AFD}\\ \langle M \rangle &\mapsto \langle M, M^{\mathcal{R}} \rangle\end{aligned}$$

avec commme propriété

$$\begin{aligned}\langle M \rangle \in S_a &\Longleftrightarrow \langle M, M^{\mathcal{R}} \rangle \in EQ_{AFD}\\ &\Longleftrightarrow L(M) = L(M^{\mathcal{R}})\\ &\Longleftrightarrow \langle M, M^{\mathcal{R}} \rangle \in EQ_{AFD}\end{aligned}$$

1. automate fini déterministe

(b) $S_b \leq_m EQ_{AFD}$

Premièrement, remarquons que

$$\begin{aligned}\langle M_1, M_2\rangle \in S_b &\Longleftrightarrow L(M_1) \subseteq L(M_2)\\ &\Longleftrightarrow L(M_1) \cap L(M_2) = L(M_1)\end{aligned}$$

De plus, nous savons constuire M_i tel que $L(M_i) = L(M_1) \cap L(M_2)$

Pour la réduction $S_b \leq_m EQ_{AFD}$ il nous faut donc un fonction f tel que

$$\begin{aligned}f : S_b \quad &\to EQ_{AFD}\\ \langle M_1, M_2\rangle &\mapsto \langle M_1, M_i\rangle\end{aligned}$$

avec commme propriété

$$\begin{aligned}\langle M\rangle \in S_b &\Longleftrightarrow L(M_1) \subseteq L(M_2)\\ &\Longleftrightarrow L(M_1) \cap L(M_2) = L(M_1)\\ &\Longleftrightarrow L(M_i) = L(M_1)\\ &\Longleftrightarrow \langle M_1, M_i\rangle \in EQ_{AFD}\end{aligned}$$

Barème : 5 points pour la bonne approche, 5 points pour la démonstration.