Automates, bisimilarité, équivalence

Épreuve pratique d'algorithmique et de programmation
Concours commun des Écoles normales supérieures

Durée de l'épreuve: 3 heures 30 minutes

Juin/Juillet 2017

ATTENTION !

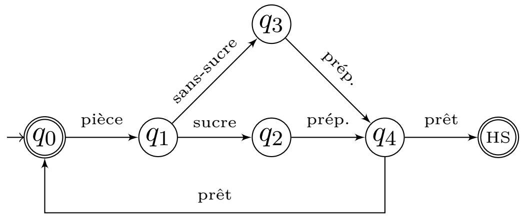
N'oubliez en aucun cas de recopier votre u_0
à l'emplacement prévu sur votre fiche réponse

Important.

Il vous a été donné un numéro u_0 qui servira d'entrée à vos programmes. Les réponses attendues sont généralement courtes et doivent être données sur la fiche réponse fournie à la fin du sujet. À la fin du sujet, vous trouverez en fait deux fiches réponses. La première est un exemple des réponses attendues pour un $\widetilde{u_0}$ particulier (précisé sur cette même fiche et que nous notons avec un tilde pour éviter toute confusion!). Cette fiche est destinée à vous aider à vérifier le résultat de vos programmes en les testant avec $\widetilde{u_0}$ au lieu de u_0. Vous indiquerez vos réponses (correspondant à votre u_0) sur la seconde et vous la remettrez à l'examinateur à la fin de l'épreuve.

En ce qui concerne la partie orale de l'examen, lorsque la description d'un algorithme est demandée, vous devez présenter son fonctionnement de façon schématique, courte et précise. Vous ne devez en aucun cas recopier le code de vos procédures!
Quand on demande la complexité en temps ou en mémoire d'un algorithme en fonction d'un paramètre n, on demande l'ordre de grandeur en fonction du paramètre, par exemple: $O(n^2)$, $O(n \log n)$,...
Il est recommandé de commencer par lancer vos programmes sur de petites valeurs des paramètres et de ***tester vos programmes sur des petits exemples que vous aurez résolus préalablement à la main ou bien à l'aide de la fiche réponse type fournie en annexe***.
Enfin, il est recommandé de lire l'intégralité du sujet avant de commencer afin d'effectuer les bons choix de structures de données dès le début.

Dans ce sujet, on s'intéresse à la modélisation de systèmes informatiques par des automates finis. Dans cette approche, l'état d'un automate représente la configuration du système, et les transitions représentent les évènements reçus ou émis qui modifient cette configuration. L'automate suivant illustre le fonctionnement d'une machine à café.



À partir de l'état initial q_0, l'automate réagit à l'insertion d'une pièce en se déplaçant vers l'état q_1. Il attend ensuite que l'utilisateur choisisse entre sucre et sans-sucre, choix qui a pour effet de déplacer l'automate vers l'état q_2 ou q_3, puis il démarre la préparation du café en allant vers l'état q_4. Cet automate contient du non-déterminisme : quand le café est prêt, il quitte l'état q_4, soit pour revenir à l'état initial, soit pour entrer dans un état hors-service (par exemple s'il n'y a plus de café ou si le filtre est à changer). Les états finaux sont l'état initial et l'état HS, on les représente avec un double contour.

Rappels et notations

On notera les intervalles d'entiers $\{a, a+1, \ldots, b\}$ par $[a, b]$.

Si $a \geq 0$ et $b > 0$ sont deux entiers, on note $a \ \mathsf{mod}\ b$ le reste de la division euclidienne de a par b.

On rappelle qu'un automate fini est un quintuplet $\mathcal{A} = (Q, \Sigma, \delta, I, F)$ où Q est un ensemble fini dont les éléments sont appelés états, Σ est un ensemble fini appelé alphabet dont les éléments sont appelés lettres, I est un sous-ensemble de Q dont les éléments sont appelés états initiaux, F est un sous-ensemble de Q dont les éléments sont appelés états finaux, et $\delta \subseteq Q \times \Sigma \times Q$ est un ensemble dont les éléments sont appelés transitions. On dit que l'automate $\mathcal{A}$ est complet si pour tout $q \in Q$ et $\sigma \in \Sigma$, il existe $q' \in Q$ tel que $(q, \sigma, q') \in \delta$.

Un chemin de $\mathcal{A}$ est une suite $q_0\sigma_0q_1\sigma_1 \ldots \sigma_{k-1}q_k$ où chaque q_i est un état, chaque σ_i est une lettre, $q_0 \in I$, et (q_i, σ, q_{i+1}) appartient à δ pour tout $i \in [0, k-1]$. On dit qu'un état q de $\mathcal{A}$ est accessible s'il existe un chemin $q_0\sigma_0q_1 \ldots q_k$ tel que $q_k = q$. La trace d'un chemin est la projection du chemin vers les lettres de Σ, c'est-à-dire $\sigma_1\sigma_2 \ldots \sigma_{k-1}$. Il s'agit d'une trace acceptante si le chemin aboutit dans un état final. On note $\mathcal{L}(\mathcal{A})$ le langage reconnu par $\mathcal{A}$, c'est-à-dire l'ensemble de ses traces acceptantes.

Si $L, L' \subseteq \Sigma^*$ sont deux langages, on note $L \cdot L' = \{w \cdot w' \mid w \in L, w' \in L'\}$ le langage des mots obtenus en concaténant un mot de L et un mot de L'. De même, L^k est le langage dont les éléments s'obtiennent en concaténant k mots de L. En particulier, L^0 est réduit au mot vide.

Enfin, on pose $L^{\leq k} = \bigcup_{0 \leq \ell \leq k} L^\ell$ et $L^{\geq k} = \bigcup_{\ell \geq k} L^\ell$.

1 Automates

Soit u_0 indiqué sur votre feuille (à reporter sur votre fiche réponse). On définit

$$\forall t \geq 1, \quad u_t = (a \cdot u_{t-1} + c) \bmod m,$$

avec $a = 1103515245$, $c = 12345$ et $m = 2^{15}$.

Question 1 *Calculer* ***a)*** u_2 ***b)*** u_{10} ***c)*** u_{1000} ***d)*** u_{11000}.

Étant donnés $n, m, d \geq 1$, on définit des automates $\mathcal{A}_t(n, m, d) = (Q, \Sigma, \delta, I, F)$ par

$$Q = [0, n-1], \qquad \Sigma = [0, m-1], \qquad I = \{0\}, \qquad F = \{n-1\},$$

$$\delta = \left\{ (i, \sigma, j) \in Q \times \Sigma \times Q \;\middle|\; u_{10t + x(i,j,\sigma)} \bmod \left\lfloor \frac{n^2 m}{d(n-i+1)} + 1 \right\rfloor = 0 \right\}$$

où $x(i, j, \sigma) = (271 u_i + 293 u_j + 283 u_\sigma) \bmod 10000$.

Question 2 *Donner le nombre de transitions de $\mathcal{A}_t(n, m, d)$ pour les paramètres suivants.*

a) $t = 1$, $n = 10$, $m = 5$, $d = 3$ ***b)*** $t = 2$, $n = 100$, $m = 10$, $d = 20$
c) $t = 3$, $n = 900$, $m = 20$, $d = 200$

Question 3 *Donner le nombre d'états accessibles dans l'automate $\mathcal{A}_t(n, m, d)$ pour les paramètres suivants.*

a) $t = 1$, $n = 10$, $m = 5$, $d = 3$ ***b)*** $t = 2$, $n = 100$, $m = 10$, $d = 5$
c) $t = 3$, $n = 200$, $m = 10$, $d = 6$ ***d)*** $t = 4$, $n = 1000$, $m = 25$, $d = 5$

Question à développer pendant l'oral 1 *Présenter l'algorithme que vous avez utilisé, et analyser sa complexité en fonction de n et m.*

Les questions 4 à 6 peuvent être résolues par un même algorithme ou par des algorithmes différents.

On rappelle que le minimum de l'ensemble vide est ∞.

Question 4 *Pour chacun des jeux de paramètres suivants, calculer le minimum et la cardinalité de l'ensemble des $t \in [1, T]$ pour lesquels $\mathcal{A}_t(n, 10, d)$ admet une trace acceptante qui appartient au langage*

$$\phi = (\Sigma \setminus \{1\})^{\geq 0} \cdot \{1\} \cdot \Sigma^{\leq 3} \cdot \{2\},$$

où $\Sigma = [0, 9]$.

a) $n = 10$, $d = 5$, $T = 15$ ***b)*** $n = 50$, $d = 5$, $T = 100$
c) $n = 75$, $d = 7$, $T = 100$ ***d)*** $n = 200$, $d = 10$, $T = 100$

Question à développer pendant l'oral 2 *Donner un automate qui reconnaît le langage ϕ défini dans la question précédente.*

Question 5 *Pour chacun des jeux de paramètres suivants, calculer le minimum et la cardinalité de l'ensemble des $t \in [1, 100]$ pour lesquels $\mathcal{A}_t(n, 10, d)$ admet une trace acceptante qui appartient au langage reconnu par $\mathcal{A}_{t+1}(n, 10, d)$.*

a) $n = 20$, $d = 10$ ***b)*** $n = 50$, $d = 10$ ***c)*** $n = 150$, $d = 8$ ***d)*** $n = 500$, $d = 12$

Question à développer pendant l'oral 3 *Donner un algorithme permettant de répondre à la question précédente. Analyser sa complexité en fonction de n.*

La question 6 est indépendante de la suite du problème. Si le code développé pour répondre à la question 5 ne s'adapte pas facilement à celle-ci, il peut être préférable de traiter la suite du sujet avant d'y revenir.

Question 6 *Pour chacun des jeux de paramètres suivants, calculer le minimum et la cardinalité de l'ensemble des $t \in [1, 10]$ pour lesquels les langages reconnus par les $k+1$ automates $\mathcal{A}_{5t}(10, 2, 5)$, $\mathcal{A}_{5t+1}(10, 2, 5)$, ..., $\mathcal{A}_{5t+k}(10, 2, 5)$ contiennent une trace acceptante commune.*

a) $k = 3$ ***b)*** $k = 4$ ***c)*** $k = 5$

2 Bisimulation

Dans cette section, on souhaite développer un algorithme qui détermine si deux automates donnés décrivent des systèmes équivalents. Pour cela, nous allons commencer par étudier l'équivalence d'un même automate qui démarre à partir de deux états différents.

Une façon de formaliser l'équivalence de deux automates est la bisimulation. Intuitivement, deux états q et q' d'un automate sont bisimilaires s'il est possible d'imiter toute exécution à partir de q par une exécution à partir de q', et inversement, de telle sorte qu'à chaque étape de l'exécution, les deux automates permettent d'exécuter les mêmes actions. Formellement, pour un automate $\mathcal{A} = (Q, \Sigma, \delta, I, F)$ donné, une relation $R \subseteq Q \times Q$ est une bisimulation si c'est une relation d'équivalence et si pour tout $(q, q') \in R$, les conditions suivantes sont satisfaites :

1. on a $q \in F$ si et seulement si $q' \in F$,
2. pour tout $\sigma \in \Sigma$, s'il existe $p \in Q$ tel que $(q, \sigma, p) \in \delta$, il existe $p' \in Q$ tel que $(q', \sigma, p') \in \delta$ et $(p, p') \in R$.

On dit que deux états q, q' sont bisimilaires s'il existe une bisimulation R telle que $(q, q') \in R$.

Considérons l'exemple suivant. Ici, 0 et 1 sont bisimilaires, et 3 et 4 aussi. Aucune autre paire d'états n'est bisimilaire.

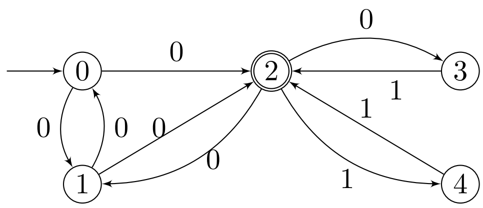


Figure 1 – Un automate où l'état 0 est l'état initial, et l'état 2 l'état final.

Question à développer pendant l'oral 4 *Justifier que les paires d'états $\{0, 1\}$ et $\{3, 4\}$ sont bisimilaires dans l'exemple précédent.*

Toujours pour un automate $\mathcal{A} = (Q, \Sigma, \delta, I, F)$, si $\sigma \in \Sigma$ et $P \subseteq Q$, notons

$$\mathsf{Pred}(P, \sigma) = \{q \in Q \mid \exists q' \in P, (q, \sigma, q') \in \delta\}$$

l'ensemble des prédécesseurs de P par σ. Considérons l'algorithme suivant, qui prend en entrée l'automate $\mathcal{A}$:

- $\mathcal{P} \leftarrow \{Q \setminus F, F\}$
- Tant qu'il existe $P, P' \in \mathcal{P}$ et $\sigma \in \Sigma$ tels que $P \cap \mathsf{Pred}(P', \sigma) \neq \emptyset$ et $P \not\subseteq \mathsf{Pred}(P', \sigma)$:
  - Choisir un tel couple (P, P')
  - $\mathcal{P} \leftarrow \mathcal{P} \setminus \{P\} \cup \{P \cap \mathsf{Pred}(P', \sigma), P \setminus \mathsf{Pred}(P', \sigma)\}$
- Renvoyer $\mathcal{P}$

Question à développer pendant l'oral 5 *Montrer que l'algorithme ci-dessus se termine et renvoie une partition de Q. Montrer que la relation d'équivalence associée à cette partition est une bisimulation.*

Question 7 *Donner le nombre de classes d'équivalence de la bisimulation calculée par l'algorithme précédent dans l'automate $\mathcal{A}_t(n, m, d)$ pour*

a) $t = 1$, $n = 20$, $m = 2$, $d = 40$, ***b)*** $t = 2$, $n = 40$, $m = 3$, $d = 70$,
c) $t = 3$, $n = 75$, $m = 3$, $d = 150$, ***d)*** $t = 4$, $n = 100$, $m = 5$, $d = 200$.

On définit l'union disjointe de deux automates $\mathcal{A} = (Q, \Sigma, \delta, \{q_0\}, F)$ et $\mathcal{A}' = (Q', \Sigma', \delta', \{q_0'\}, F')$ avec $Q \cap Q' = \emptyset$ comme l'automate

$$\mathcal{U} = (Q \cup Q', \Sigma \cup \Sigma', \delta \cup \delta', \{q_0, q_0'\}, F \cup F').$$

On dit que les deux automates $\mathcal{A}$ et $\mathcal{A}'$ sont bisimilaires si et seulement si les états q_0 et q_0' de $\mathcal{U}$ sont bisimilaires.

Question à développer pendant l'oral 6 *Démontrer que deux automates bisimilaires reconnaissent le même langage. Démontrer que la réciproque est vraie pour les automates déterministes complets, mais pas pour les automates non-déterministes même complets.*

On voudrait comprendre la « robustesse » de nos automates aux erreurs survenues lors des transitions : si on change les lettres d'une petite partie des transitions, le nouvel automate est-il toujours équivalent à l'automate de départ ?

Pour cela, on définit une version erronée de nos automates. Pour tout $t, n, m, d \geq 1$ et $p \geq 1$, soit $\mathcal{A}_t^{\mathsf{err}(p)}(n, m, d)$ l'automate obtenu à partir de $\mathcal{A}_t(n, m, d)$ en remplaçant chaque transition (i, σ, j) par la transition suivante :

$$\begin{cases} (i, \sigma', j) & \text{avec } \sigma' = u_{5t+y(i,j,\sigma)+100} \mod m, \text{ si } u_{5t+y(i,j,\sigma)} \mod p = 0, \\ (i, \sigma, j) & \text{sinon,} \end{cases}$$

où $y(i, j, \sigma) = 41i + 31j + \sigma$.

Question 8 *Pour les paramètres suivants, déterminer le nombre de transitions de $\mathcal{A}_t^{\mathsf{err}(p)}(n, m, d)$.*

a) $t = 1$, $n = 10$, $m = 2$, $d = 50$, $p = 25$ ***b)*** $t = 2$, $n = 20$, $m = 2$, $d = 100$, $p = 50$
c) $t = 3$, $n = 25$, $m = 2$, $d = 100$, $p = 50$ ***d)*** $t = 4$, $n = 30$, $m = 2$, $d = 100$, $p = 50$

Question 9 *Pour chacun des jeux de paramètres suivants, calculer le minimum et la cardinalité de l'ensemble des $t \in [1, 100]$ pour lesquels $\mathcal{A}_t(n, m, d)$ et $\mathcal{A}_t^{\mathsf{err}(p)}(n, m, d)$ sont bisimilaires.*

a) $n = 10$, $m = 2$, $d = 50$, $p = 25$ ***b)*** $n = 20$, $m = 2$, $d = 100$, $p = 50$
c) $n = 25$, $m = 2$, $d = 100$, $p = 50$ ***d)*** $n = 30$, $m = 2$, $d = 100$, $p = 50$

3 Équivalence de traces

Question à développer pendant l'oral 7 *Étant donnés deux automates finis non déterministes $\mathcal{A}$ et $\mathcal{B}$, expliquer comment on détermine si $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$. Quelle est la complexité de cette procédure ? Jusqu'à quelle taille d'automates peut-on espérer aller en pratique ?*

Question 10 *Pour chacun des jeux de paramètres suivants, calculer le minimum et la cardinalité de l'ensemble des $t \in [1, 50]$ pour lesquels on a $\mathcal{L}(\mathcal{A}_{2t}(n, 2, d)) \subseteq \mathcal{L}(\mathcal{A}_{2t+1}(n, 2, d'))$.*

a) $n = 10,\ d = 3,\ d' = 8$ ***b)*** $n = 15,\ d = 5,\ d' = 10$
c) $n = 60,\ d = 5,\ d' = 10$ ***d)*** $n = 90,\ d = 10,\ d' = 15$

Question 11 *Pour chacun des jeux de paramètres suivants, calculer le minimum et la cardinalité de l'ensemble des $t \in [1, 100]$ pour lesquels $\mathcal{A}_t(n, 2, d)$ et $\mathcal{A}_t^{\mathsf{err}(p)}(n, 2, d)$ reconnaissent le même langage mais ne sont pas bisimilaires.*

a) $n = 10,\ d = 50,\ p = 50$ ***b)*** $n = 15,\ d = 75,\ p = 50$
c) $n = 20,\ d = 100,\ p = 50$ ***d)*** $n = 25,\ d = 100,\ p = 50$

Question à développer pendant l'oral 8 *Comment peut-on se servir des résultats de la section précédente pour améliorer la performance de cet algorithme ?*

Fiche réponse type: Automates, bisimilarité, équivalence
$\widetilde{u_0}$: 1

Question 1

a) 12519

b) 21983

c) 27481

d) 32201

Question 2

a) 19

b) 1116

c) 92390

Question 3

a) 9

b) 86

c) 195

d) 993

Question 4

a) (2, 7)

b) (3, 25)

c) (4, 29)

d) (1, 43)

Question 5

a) (1, 82)

b) (1, 79)

c) (2, 35)

d) (1, 93)

Question 6

a) (2, 7)

b) (2, 6)

c) (5, 4)

Question 7

a) 12

b) 34

c) 52

d) 83

Question 8

a) 164

b) 635

c) 907

d) 1172

Question 9

a) $(2,\ 39)$

b) $(1,\ 29)$

c) $(1,\ 15)$

d) $(14,\ 8)$

Question 10

a) $(1,\ 20)$

b) $(1,\ 27)$

c) $(2,\ 20)$

d) $(7,\ 11)$

Question 11

a) $(1,\ 46)$

b) $(9,\ 55)$

c) $(2,\ 70)$

d) $(2,\ 84)$

Fiche réponse: Automates, bisimilarité, équivalence

Nom, prénom, u_0: ....

Question 1

a)

b)

c)

d)

Question 2

a)

b)

c)

Question 3

a)

b)

c)

d)

Question 4

a)

b)

c)

d)

Question 5

a)

b)

c)

d)

Question 6

a)

b)

c)

Question 7

a)

b)

c)

d)

Question 8

a)

b)

c)

d)

Question 9

a)

b)

c)

d)

Question 10

a)

b)

c)

d)

Question 11

a)

b)

c)

d)