Examen
Théorie des langages (INF105)

Pierre SENELLART
pierre.senellart@telecom-paristech.fr

L'examen du module INF105 dure une heure et demie. Tous les documents sont autorisés, mais calculatrices, ordinateurs et objets communicants sont interdits. L'énoncé comporte un unique problème noté sur 20 points (+ 2 points bonus).

Tailles de description

Dans ce problème, nous allons nous intéresser à la taille de description de certains langages *rationnels* par différents formalismes : expressions rationnelles, automates déterministes ou non-déterministes, grammaires hors-contextes, programmes d'un langage de programmation. Nous allons ainsi montrer que, même si un langage est rationnel, il peut être décrit de manière beaucoup plus concise dans un formalisme plus puissant que les formalismes servant à décrire spécifiquement les langages rationnels.

Préliminaires

Dans tout le sujet, l'alphabet est $\Sigma = \{a, b\}$. Quand une expression rationnelle est demandée, on ne s'autorisera que les opérateurs de disjonction (que l'on écrit $+$ ou $|$), de concaténation, et d'étoile *, ainsi que les symboles ε, $\emptyset$ et a, b et les parenthèses. On exclut donc les raccourcis syntaxiques souvent utilisés dans les expressions rationnelles : on n'écrira pas « a^5 » mais « $aaaaa$ », pas « $a^?$ » mais « $(a + \varepsilon)$ », pas « a^+ » mais « aa^* », pas les classes de caractères « . » ou « $[ab]$ » de grep mais « $a + b$ ».

On s'intéressera à la *taille* de formalismes de représentation, définie conventionnellement comme suit :

— La taille d'une expression rationnelle est le nombre de symboles (opérateurs, symboles de l'alphabet, etc.) utilisés pour l'écrire, sans compter les parenthèses (ou les espaces).
— La taille d'un automate fini (déterministe ou non-déterministe) est son nombre d'états.
— La taille d'une grammaire est la somme des tailles des règles de production de cette grammaire ; la taille d'une règle de grammaire $\alpha \to \beta$ est $|\alpha| + |\beta|$.
— La taille d'un programme écrit dans un langage de programmation donné est le nombre de caractères utilisés pour son écriture.

On rappelle les notations $O(\cdot)$ et $\Omega(\cdot)$:

— Une fonction $f : \mathbb{N} \to \mathbb{R}^+$ est telle que $f(n) = O(\varphi(n))$ si et seulement s'il existe un entier n_0 et un réel A strictement positif tels que $\forall n \geqslant n_0$, on a $f(n) \leqslant A \times \varphi(n)$.
— Une fonction $f : \mathbb{N} \to \mathbb{R}^+$ est telle que $f(n) = \Omega(\varphi(n))$ si et seulement s'il existe un entier n_0 et un réel A strictement positif tels que $\forall n \geqslant n_0$, on a $f(n) \geqslant A \times \varphi(n)$.

On rappelle que $f(n) = O(g(n))$ si et seulement si $g(n) = \Omega(f(n))$.

1. *(13,5 points)* Pour $n \in \mathbb{N}$, soit L_n le langage sur Σ des mots u de longueur $|u| > n$ tels que le $(|u| - n)$-ième symbole de u (c'est-à-dire le $(n+1)$-ième symbole en partant de la fin du mot) est un a, les autres symboles de u étant arbitraires.

   a) *(1,5 points)* Considérons d'abord L_0.

      i. *(0,5 point)* Décrire en français le langage L_0.

      ii. *(1 point)* Donner (sans justification) une expression rationnelle e_0 décrivant le langage L_0. Quelle est la taille de cette expression ?

   b) *(6,5 points)* Considérons maintenant L_1.

      i. *(1 point)* Donner (sans justification) une expression rationnelle e_1 décrivant L_1. Quelle est la taille de cette expression ?

      ii. *(1,5 points)* Donner l'automate de Thompson A_1 pour l'expression rationnelle e_1. On utilisera la construction de Thompson telle que présentée dans le cours, sans simplifications. Quel est la taille de A_1 ?

      iii. *(2 points)* En appliquant une méthode du cours, déterminiser A_1 en un automate déterministe *complet* A_1'. Quelle est la taille de A_1' ?

      iv. *(2 points)* Minimiser A_1' pour obtenir un automate déterministe *complet* minimal A_1'' reconnaissant le langage L_1. Quelle est la taille de A_1'' ?

   c) *(5 points)* Fixons maintenant un $n \in \mathbb{N}$ arbitraire et considérons le langage L_n.

      i. *(1 point)* Décrire (sans justification) une expression rationnelle e_n décrivant L_n. Quelle est la taille de cette expression ?

      ii. *(1 point)* Sans le construire, donner la taille d'un automate non-déterministe à transitions spontanées pour L_n (on ne demande pas la taille minimale, juste une taille possible). Prouver votre affirmation.

      iii. *(1 point)* Sans le construire, donner la taille d'un automate non-déterministe *sans transitions spontanées* pour L_n (on ne demande pas la taille minimale, juste une taille possible). Prouver votre affirmation.

      iv. *(2 points, question plus difficile)* Prouver que tout automate déterministe reconnaissant L_n a au moins 2^n états.

   d) *(0,5 point)* Déduire des questions précédentes que pour tout $n \in \mathbb{N}$, il existe un langage K_n tel que K_n admet une description sous forme d'expression rationnelle de taille $O(n)$, sous forme d'automate non-déterministe (avec ou sans transitions spontanées) de taille $O(n)$ mais tel que toute description de K_n sous forme d'automate déterministe a une taille $\Omega(2^n)$.

2. *(6,5 points + 2 points bonus)* Pour $n \in \mathbb{N}$, soit $L_n' = \{a^{2^n}\}$ (c'est-à-dire, L_n' est le langage formé du seul mot a^{2^n}).

   a) *(0,5 point)* Par un argument simple, montrer que L_n' est rationnel pour tout $n \in \mathbb{N}$.

   b) *(1,5 points)* Pour $n \in \mathbb{N}$ arbitraire, donner un automate déterministe complet reconnaissant L_n' ; justifier que cet automate reconnaît le bon langage. Quelle est la taille de cet automate ?

   c) *(1,5 points)* Montrer que tout automate non-déterministe avec ou sans transitions spontanées reconnaissant L_n' pour $n \in \mathbb{N}$ a au moins $2^n + 1$ états. *Indication : on pourra commencer par traiter le cas des automates sans transitions spontanées, et s'inspirer de la preuve du lemme de pompage.*

   d) *(1 point)* Donner (sans justification) une grammaire hors-contexte pour L_n' pour $n \in \mathbb{N}$ dont la taille est en $O(n)$.

e) *(2 points)* On se fixe un entier n arbitraire. Dans le langage de programmation de votre choix parmi C, Java, Python, écrire une fonction reconnaissant le langage L'_n (c'est-à-dire, prenant en argument un mot u et retournant un booléen indiquant si $u \in L'_n$). On ne se préoccupera pas des problèmes de dépassement mémoire et on supposera que les entiers n'ont aucune limitation en taille, ni dans l'écriture du programme, ni au cours de son exécution. *Attention : n n'est pas un paramètre de la fonction ni une variable accessible dans le code, c'est une constante fixée dont la valeur doit apparaître d'une manière ou d'une autre dans le code de la fonction.* En utilisant la notation $O(\cdot)$, décrire la taille de ce programme.

f) *(2 points bonus)* On admettra sans preuve le résultat suivant : pour toute grammaire hors contexte de taille m engendrant un langage fini, la taille des mots engendrés par la grammaire est en $O(m^m)$.

Déduire de ce fait et des questions précédentes que pour tout $n \in \mathbb{N}$, il existe un langage K'_n tel que K'_n admet une description sous forme de programme d'un langage de programmation de taille $O(\log n)$, sous forme de grammaire hors-contexte de taille $O(n)$, sous forme d'automate fini déterministe de taille $O(2^n)$, mais tel que toute description sous forme de grammaire hors-contexte est de taille $\Omega(n^{1-\varepsilon})$ pour $\varepsilon > 0$ arbitraire, et toute description sous forme d'automate fini non-déterministe à transitions spontanées de taille $\Omega(2^n)$.