Calculabilité 2005
Partiel du 21 novembre

24 novembre 2005

Les exercices sont indépendants. Les documents sont autorisés. Seuls les résultats vus en cours, en devoir ou en TD peuvent être utilisés. Durée : 3h. La longueur des solutions et le barême sont donnés à titre indicatif.

1

1. Montrer que la fonction d'exponentiation sur les entiers (x^y ; par convention $0^0 = 1$) est une fonction primitive récursive.
2. Montrer que l'égalité des fonctions primitives récursives est indécidable (i.e. donnée : deux fonctions primitives récursives f, g sous la forme $\mathsf{Comp}_m(\mathsf{Prim}...)$; question : a-t-on $\forall \vec{n} \in \mathbb{N}^k,\ f(\vec{n}) = g(\vec{n})$?)

[8 lignes, 4 points]

1.1 Solution

1. La multiplication est primitive récursive (vu en TD)

$$x^y = \begin{cases} 1 & \text{si } y = 0 \\ x * x^{y-1} & \text{Sinon} \end{cases}$$

2. Vu en devoir : l'indécidabilité des systèmes Diophantiens exponentiels. Or toute composée de multiplications, d'exponentielles, d'additions et l'égalité sont récursifs primitifs (d'après la question précédente et les TDs). On peut donc réduire l'existence d'une solution d' un système Diophantien exponentiel à l'égalité d'une fonction récursive primitive à la fonction nulle. D'où l'indécidabilité de l'égalité des fonctions récursives primitives.

2

Donner explicitement la table d'une machine de Turing qui, étant donné un mot $w \in \{0,1\}^*$, accepte w si w contient au moins autant de 0 que de 1 et rejette sinon. (On prendra soin de démontrer que la machine fait bien ce qu'elle est censée faire).

[22 lignes, 5 points]

2.1 Solution

États, alphabet et transitions sont donnés par la table suivante ; q_0 est l'état initial.

	0	1	B	\$	2
q_0	$0, q_0, \rightarrow$	$2, q_1, \leftarrow$	**accept**	$\$, q_0, \rightarrow$	$2, q_0, \rightarrow$
q_1	$0, q_1, \leftarrow$	$1, q_1, \leftarrow$	**reject**	$\$, q_2, \rightarrow$	$2, q_1, \leftarrow$
q_2	$2, q_3, \leftarrow$	$1, q_2, \rightarrow$	**reject**	**reject**	$2, q_2, \rightarrow$
q_3	$0, q_3, \leftarrow$	$1, q_3, \leftarrow$	**accept**	$\$, q_0, \rightarrow$	$2, q_3, \leftarrow$

Tout d'abord, la machine s'arrête sur toute entrée. En effet : le nombre de symboles distincts de $B, 2$ inscrits sur le ruban est décroissant. Et lorsqu'il reste constant, l'état décroit pour l'ordre $q_1 \geq q_2 \geq q_3 \geq q_0$. Enfin, si l'état reste identique, le nombre de symboles à gauche de la tête de lecture décroît strictement dans les états q_1, q_3 et le nombre de symboles à droite de la tête de lecture décroît strictement dans les états q_0, q_4.

Montrons ensuite, par récurrence sur la longueur de la séquence de transitions à partir de la configuration initiale $(q_0, \epsilon, \$w)$ que, selon la configuration atteinte :

1. ou bien **accept** et, dans ce cas $|w|_0 \geq |w|_1$
2. ou bien **reject** et $|w|_0 < |w|_1$
3. ou bien (q, w_1, w_2) avec $q \in \{q_0, q_3\}$ et, dans ce cas $|w_1 w_2|_0 - |w_1 w_2|_1 = |w|_0 - |w|_1$ et, si $q = q_0$, $|w_1|_1 = 0$
4. ou bien (q, w_1, w_2) avec $q \in \{q_1, q_2\}$ et, dans ce cas $|w_1 w_2|_0 - |w_1 w_2|_1 + 1 = |w|_0 - |w|_1$ et, si $q = q_2$, $|w_1|_0 = 0$

Il s'agit d'une simple vérification. Par exemple, dans la configuration (q_0, w_1, B), $|w_1|_1 = 0$ et $|w_1|_0 - |w_1|_1 = |w|_0 - |w|_1$ et donc $|w|_0 - |w|_1 \geq 0$. Ce qui permet de traiter le premier cas ci-dessus.

3

Parmi les problèmes suivants, lesquels sont (in)décidables ? (Justifier succinctement la réponse)

1. Étant donnés une machine de Turing M, un mot w et un entier n en base 2, M s'arrête-t-elle sur w en moins de n étapes de calcul ?
2. Étant donnés une machine de Turing M, un entier n, M s'arrête-t-elle en moins de n étapes pour tout mot w ?

[11 lignes, 4 points]

3.1 Solution

1. C'est un problème décidable : il suffit de simuler M et, à chaque étape de calcul, décrémenter n de 1 (ce qui est bien sûr faisable par une machine de Turing). Si M accepte, on accepte. Si n atteint 0, on s'arrête en rejettant. La machine s'arrête alors après au plus n étapes de simulations de M.

2. C'est un problème décidable : il suffit de remarquer que, en n étapes la machine M ne peut que parcourir un préfixe de longueur au plus n de w. M s'arrête en au plus n étape sur tout mot w ssi elle s'arrête en au plus n étapes sur tout mot w de longueur au plus n. Il suffit donc d'énumérer les mots de longueur au plus n (ce que l'on peut faire à l'aide d'une machine de Turing) et, pour chacun d'eux, utiliser la machine de la question précédente. On rejette dès que sur l'une des entrées la machine de la question précédente rejette. Sinon, on accepte.

4

1. On dit qu'une machine de Turing M *boucle* sur la donnée w si M passe deux fois dans la même configuration : $q, \epsilon, \$w \vdash_M^* q_1, w_1, w_2 \vdash_M^+ q_1, w_1, w_2$. Montrer que le problème de savoir si, étant donné M, M boucle sur la donnée vide, est indécidable.
2. Un *automate linéairement borné* est une machine de Turing dont l'espace de travail est limité à droite par l'extrémité du mot d'entrée. Autrement dit, pour tout état q, la transition $\delta(q, B)$, si elle existe, est de la forme $(B, q', \leftarrow)$. Montrer que le problème suivant (arrêt universel des automates linéairement bornés) est indécidable :

   Donnée : Un automate linéairement borné A

   Question : est ce que, pour tout mot w, A s'arrête sur la donnée w ?

[13 lignes, 5 points]

4.1 Solution

1. On réduit le problème de l'arrêt. À M, w, on associe la machine M_w à deux rubans qui commence par écrire w sur le premier ruban et 0 sur le second. Ensuite M simule w sur le premier ruban, en ajoutant 0 à chaque mouvement, sur le second ruban. Quand M va s'arrêter sur w, M_w se met alors à boucler (sans écrire sur le second ruban). Alors M_w boucle si et seulement si M s'arrête sur w. D'où l'indécidabilité du problème.
2. On réduit le problème du bouclage de M sur la donnée vide. Étant donné une machine M, on construit l'automate linéairement borné M' qui, sur la donnée w, simule M sur le mot vide (c'est à dire commence par remplacer les symboles de w par B' puis effectue les transitions de M comme si B' était un blanc), en s'arrêtant quand M va écrire sur B. M' s'arrête pour toute donnée si et seulement si M ne boucle pas sur le mot vide. En effet, un automate linéairement borné ne s'arrête pas si et seulement si il boucle. De plus, si M boucle sur w, elle boucle dans un espace borné. D'où l'indécidabilité du problème.

5

Montrer que le Problème de correspondance de Post reste indécidable lorsque tous les mots des deux séquences ont pour longueur au plus 2.

Qu'en est il si tous les mots ont pour longueur 2 ?

[24 lignes, 5 points]

5.1 Solution

On réduit PCP a PCP avec seulement des mots de longueur 2 au plus. Soit $(u_1,\dots,u_n),(v_1,\dots,v_n)$ une instance de PCP.

Pour chaque mot u_i de longueur supérieure ou égale à 2, $u_i = a_i^1 \cdots a_i^{k_i}$, on remplace la paire (u_i, v_i) par les paires

$$\begin{array}{l}(u_i^1, v_i^1) = (a_i^1, v_i\alpha_1), (u_i^2, v_i^2) = (a_i^2, \overline{\alpha_1}\alpha_2), \dots (u_i^{k_i}, v_i^{k_i}) = (a_i^{k_i}, \overline{\alpha_{k_i-1}}),\\ (u_i^{k_i+1}, v_i^{k_i+1}) = (\alpha_1\overline{\alpha_1}, \epsilon), \dots (u_i^{2k_i-1}, v_i^{2k_i-1}) = (\alpha_{k_i-1}\overline{\alpha_{k_i-1}}, \epsilon)\end{array}$$

où les α_i, $\overline{\alpha_i}$ sont des nouveaux symboles, distincts deux à deux.

Après cette transformation, les mots u_i^j ont une longueur au plus deux et les seuls mots de longueur 2 sont de la forme $\alpha\overline{\alpha}$, le mot v_i^j correspondant étant ϵ. Montrons que la nouvelle instance de PCP a une solution ssi la première instance a une solution.

Si $u_{i_1} \cdots u_{i_k} = v_{i_1} \cdots v_{i_k}$ alors, $s_{i_1} \cdots s_{i_k} = \phi(t_{i_1} \cdots t_{i_k})$ où $s_j = u_j^1 \cdots u_j^{k_j}$, $t_j = v_j^1 \cdots v_j^{k_j}$ et ϕ est le morphisme qui envoie tous les mots $\alpha\overline{\alpha}$ sur ϵ. On obtient donc $t_{i_1} \cdots t_{i_k}$ par insertion dans $s_{i_1} \cdots s_{i_k}$ de mots $\alpha\overline{\alpha}$: $t_{i_1} \cdots t_{i_k} = s'_{i_1} \cdots s'_{i_k}$ où $s'_j = \theta_j^1 u_j^1 \cdots \theta_j^{k_j} u_j^{k_j} \theta_j^{k_j+1}$ et θ_j^i est un produit de mots de la forme $\alpha\overline{\alpha}$. On obtient alors une solution pour la nouvelle instance de PCP.

Réciproquement, si la nouvelle instance de PCP a une solution $s_{i_1} \cdots s_{i_k} = t_{i_1} \cdots t_{i_k}$, alors, comme toutes les lettres α sont suivies de $\overline{\alpha}$ dans les mots s_j, si $t_m = v_i\alpha_1$ alors il existe $p \geq 1$ tel que $t_m \cdots t_{m+p} = v\alpha_1 \cdot \overline{\alpha_1}\alpha_2 \cdots \overline{\alpha_r}$ et $\phi(s_m \cdots s_{m+p}) = u_i$. Il en résulte que $\phi(s_{i_1} \cdots s_{i_k}) = u_{j_1} \cdots u_{j_l} = \phi(t_{i_1} \cdots t_{i_k}) = t_{j_1} \cdots t_{j_l}$ et on a ainsi une solution de l'instance originale de PCP.

On procède ensuite de même pour chaque v_i de longueur strictement supérieure à 2. On obtient alors deux suites de mots, chacun d'eux étant de longueur au plus 2. Par symétrie, le raisonnement ci-dessus s'applique et cette nouvelle instance de PCP a une solution ssi l'instance de départ a une solution.

Exemple de codage :

i	1	2	3	4
u_i	a	b	ca	abc
v_i	ab	ca	a	c

Cette instance de PCP a une solution (12314).

Codage :

i	1	2	3	4	5	6	7	8	9	10
s_i	a	b	c	a	$\alpha_1\overline{\alpha_1}$	a	b	c	$\alpha_2\overline{\alpha_2}$	$\alpha_3\overline{\alpha_3}$
t_i	ab	ca	$a\alpha_1$	$\overline{\alpha_1}$	ϵ	$c\alpha_2$	$\overline{\alpha_2}\alpha_3$	$\overline{\alpha_3}$	ϵ	ϵ

Avec pour solution (1234156789[10])

Si tous les mots sont de longueur 2, le problème devient décidable. En fait, il n'a de solution que si, pour au moins un indice i, $u_i = v_i$.