TD 6

8 et 10 Novembre 1999

Fonctions calculables?

1. Montrer qu'il n'existe pas de fonction récursive f telle que :

   (a) $f(x) = \begin{cases} 1 & \text{si } \varphi_x(x) \text{ est défini,} \\ 0 & \text{sinon.} \end{cases}$

   (b) $f(x) = \begin{cases} 1 & \text{si } \varphi_x \text{ est totale,} \\ 0 & \text{sinon.} \end{cases}$

Soyons clean...

1. Montrer qu'il n'existe pas de machine de Turing qui décide si

   (a) une machine de Turing ne s'arrête jamais;

   (b) le domaine de la fonction calculée par une machine de Turing est infini;

   (c) deux machines de Turing calculent la même fonction.

2. Montrer le théorème de Kleene (pour toute fonction f récursive totale, il existe n tel que $\varphi_n = \varphi_{f(n)}$) en utilisant le théorème s-n-m.

Championnat du castor affairé

Soit M_n une machine de Turing d'alphabet $\{0, 1\}$ (0 étant le blanc) possédant $n+1$ états $q_0, \ldots, q_n$, q_0 étant l'état initial et q_n l'état d'acceptation (il n'y a pas d'état de refus), et un unique ruban bi-infini avec une seule tête de lecture-écriture, vide au départ.

Le "championnat du castor affairé" est un jeu proposé aux machines de Turing du type de M_n : gagne celle qui réussit à inscrire le plus grand nombre de 1 sur le ruban avant de s'arrêter.

1. Montrer que pour n fixé, l'ensemble des nombres de 1 que peuvent écrire les machines de M_n admet un maximum $\Sigma(n)$ (la machine réalisant ce maximum est déclarée vainqueur de la compétition du castor affairé en catégorie n). Montrer que la fonction $\Sigma : \mathbb{N}^\star \to \mathbb{N}$ est strictement croissante.

2. Montrer que si f est une fonction Turing-calculable[1], alors $g_f : x \mapsto \max[f(2x+2), f(2x+3)]$ est Turing-calculable par une machine de Turing que l'on notera F_f ; on notera k_f son nombre d'états.

[1] L'entrée et la sortie étant codées en unaire

3. Pour $x \in \mathbb{N}$, et f Turing-calculable construire une machine de Turing $N_{x,f}$ écrivant x fois 1 sur le ruban vide puis applique F_f ; calculer son nombre d'états.

4. En déduire que pour toute fonction Turing-calculable f, il existe $x_0 \in \mathbb{N}$ vérifiant $\forall x \geq x_0 \; f(x) < \Sigma(x)$; en déduire que Σ n'est pas récursive.

5. Lin et Rado (1975) ont montré que $\Sigma(1) = 1$, $\Sigma(2) = 4$ et $\Sigma(3) = 6$.

   On sait également

$\Sigma(4)$	$=$	13		(Brady 1966 - 1975)
$\Sigma(5)$	$>$	500		(U. Schult 1983)
$\Sigma(8)$	$>$	8.10^{44}		(GREEN 1964)
$\Sigma(12)$	$>$	$6.4096^{4096^{\cdot^{\cdot^{\cdot^{4096^4}}}}}$	où 4096 apparaît 166 fois	(U. Schult 1983)

   (a) Construire les machines de Turing vainqueurs de la compétition du castor affairé en catégorie 1 et 2.

   (b) Que fait la machine de Turing de M_3 définie ci-dessous ?

$$
\begin{array}{lclclcl}
\delta(q_0, B) & = & (q_1, 1, \rightarrow) & \delta(q_0, 1) & = & (q_2, 1, \leftarrow) \\
\delta(q_1, B) & = & (q_0, 1, \leftarrow) & \delta(q_1, 1) & = & (q_1, 1, \rightarrow) \\
\delta(q_2, B) & = & (q_1, 1, \leftarrow) & \delta(q_2, 1) & = & (q_3, 1, \leftarrow)
\end{array}
$$