Magistère de Mathématiques et d'Informatique
ENS de Cachan *23 octobre 2006*
TD de Calculabilité et de Complexité

Fonctions récursives

Exercice 1
Montrer que les fonctions suivantes sont récursives primitives (les prédicats sont vus comme des fonctions à valeur dans $\{0,1\}$).
- la factorielle
- le test d'égalité

Exercice 2
Étant données f, g et h récursives primitives, montrer que la fonction k définie par :

$$\forall x \in \mathbb{N}, k(x) = \begin{cases} f(x) \text{ si } h(x) = 0 \\ g(x) \text{ sinon} \end{cases}$$

est récursive primitive

Exercice 3
F est *clos par maximisation bornée* si, pour toutes fonctions $\zeta, \psi \in F$ telles que

$$\forall \overrightarrow{n}, \exists m \leq \psi(\overrightarrow{n}).\zeta(\overrightarrow{n}, m) = 0$$

alors la fonction ϕ définie par :

$$\phi(\overrightarrow{n}) = \max_{m \leq \psi(\overrightarrow{n})} (\zeta(\overrightarrow{n}, m) = 0)$$

Montrer que l'ensemble des fonctions récursives primitives est clos par maximisation bornée.

Exercice 4
Montrer que, comme dans l'exercice précédent, l'ensemble des fonctions récursives primitives est clos par minimisation bornée.

Exercice 5 (Hiérarchie de Grzegorczyk)
On définit par récurrence sur n la suite de fonctions primitives récursives $(\psi_n)_{n \geq 0}$ ainsi :

$$\begin{array}{rcl} \psi_0(m) & = & m+1 \\ \psi_{n+1}(0) & = & \psi_n(1) \\ \psi_{n+1}(m+1) & = & \psi_n(\psi_{n+1}(m)) \end{array}$$

Montrer que :

1. Pour tous n, m, $\psi_n(m) \geq m+1$
2. Les fonctions ψ_k sont croissantes pour tout k
3. Pour tous n, m, k, $\psi_n(m) + k \leq \psi_{n+k}(m)$

Rappel : on note $\mathcal{C}$ est le plus petit ensemble de fonctions sur les entiers qui contient les fonctions initiales, l'addition, la multiplication, l'égalité et qui est clos par composition et minimisation (donc pas de récursion primitive).

Exercice 6
Montrer que les fonctions ou prédicats suivants sont dans $\mathcal{C}$:

1. conjonction, disjonction, négation de prédicats dans $\mathcal{C}$
2. Si $P, f_1, f_2 \in \mathcal{C}, f = \lambda n.\text{if } P(n) \text{ then } f_1(n) \text{ else } f_2(n)$
3. La soustraction entière : $x \setminus y = \max(0, x - y)$
4. L'inégalité $x \geq y$
5. $D(n, m)$ qui est satisfait si $n \neq 0$ et n divise m
6. $\text{rem}(x, y)$ qui calcule le reste de la division euclidienne de x par y lorsque $y \neq 0$ (et est indéfini lorsque $y = 0$)
7. $\forall x \leq f(\overrightarrow{y}).P(x, \overrightarrow{y})$ et $\exists x \leq f(\overrightarrow{y}).P(x, \overrightarrow{y})$ où $P, f \in \mathcal{C}$.
8. $Prime(n)$ qui est satisfait par les nombres premiers.
9. $PP(n)$: "n est une puissance d'un nombre premier"

Exercice 7
Montrer que la fonction J définie par :

$$J(x, y) = x + \frac{(x + y)(x + y + 1)}{2}$$

est une bijection de $\mathbb{N}^2$ dans $\mathbb{N}$. On note $(K(z), L(z)) = J^{-1}(z)$ les fonctions réciproques. Montrer que $J, K, L \in \mathcal{C}$. Généraliser cette construction à $\mathbb{N}^k$.

Exercice 8 (Récursion primitive comme itération)
Définissons l'ensemble des fonctions de codage ainsi :
– pour tout $k \geq 2$, les fonctions $< ., \ldots, . >_k$ codant les tuples de $\mathbb{N}^k$ sur $\mathbb{N}$,
– pour tout $k \geq 2$, et $1 \leq m \leq k$, les fonctions $\pi_{m,k}$ décodant un tuple de $\mathbb{N}^k$ codé sur $\mathbb{N}$ et retournant sa m-ème composante.

Par ailleurs, étant donnée h une fonction primitive récursive, définissons la fonction f suivante :

$$f(n, x) = h^{(n)}(x) = h(h(\ldots h(x) \ldots))$$

$f(n, x)$ est l'itérée n-ème de h appliquée à x.

Montrer que le plus petit ensemble de fonctions contenant les fonctions initiales (successeur, zéro, projection sur une composante), ainsi que les fonctions de codage définies plus haut, et qui de plus est clos par composition et itération, est exactement la classe des fonctions primitives récursives.

Indication : Si f est définie par récursion sur h, définir $s(n, \overrightarrow{x}) =< f(n, \overrightarrow{x}), n, \overrightarrow{x} >$, et montrer que s peut être définie comme une itération.