Silvain Rideau
Bureau T7

silvain.rideau@ens.fr
www.eleves.ens.fr/home/srideau

TD de Logique 9 (Fonctions récursives)

26 novembre 2012

Les exercices précédés d'une ✠ sont là pour vous aider à comprendre le cours et devront être fait avant le TD pour être corrigé au début du TD. Les exercices qui ne sont pas abordés en cours, seront (certainement) corrigés sur ma page personelle (voir ci-dessus).

(✠) **Exercice 1** (Ensembles récursifs et image de fonctions récursives) **:**

1. Soit $f \in \mathcal{F}_1$ récursive totale et croissante. Montrer que l'image de f est un ensemble récursif.
2. Réciproquement, montrer que tout sous-ensemble récursif infini de $\mathbb{N}$ est l'image d'une fonction unaire récursive totale strictement croissante.

Exercice 2 (Machines de Turing) **:**
Donner une machine de Turing qui calcule les fonctions suivantes :

1. $\lambda x \lambda y\ x + y$,
2. $\lambda x \lambda y\ x \times y$,
3. La fonction caractéristique des entiers pairs.

Exercice 3 (Existence de racines) **:**

1. Soit n un entier. Montrer que l'ensemble $\{(a_0, ..., a_n) \in \mathbb{N}^{n+1} \mid$ le polynôme $a_0 + a_1 X + a_2 X^2 + ... + a_n X^n$ admet une racine dans $\mathbb{Z}\}$ est primitif récursif.
2. Qu'en est-il si on remplace "racine dans $\mathbb{Z}$" par "racine dans $\mathbb{Q}$".

Exercice 4 (Fonction universelle primitive récursive) **:**

1. Soit $f \in \mathcal{F}^1$, montrer que f est primitive récursive s'il existe un entier i et une fonction $g \in \mathcal{F}^1$ primitive récursive telle que la machine d'indice i calcule f et $T(i, x) \leq g(x)$.
2. En déduire qu'il existe une fonction récursive $\varphi \in \mathcal{F}^2$ telle que
$$\{\varphi_i \mid i \in \mathbb{N}\} = \{f \in \mathcal{F}^1 \mid f \text{ primitive récursive}\}$$
[Indication : considérer $g(i, a, n, x) = \mu y \leq \max(a, \xi_n(x))[\exists t \leq \max(a, \xi_n(x))(i, t, x, y) \in C^1]$ où ξ est la fonction d'Ackermann.]
3. Montrer que cette fonction φ ne peut pas être primitive récursive.
[Indication : Comme on pourrait s'en douter c'est un argument diagonal, regardez $\lambda x\ \varphi(x, x) + 1$.]

Exercice 5 (Une bijection primitive récursive d'inverse non primitif récursif, ou comment refaire la fin du Td précédent de façon moins ad-hoc) **:**

1. (Rappel du Td précédent) Montrer que l'ensemble des bijections récursives de $\mathbb{N}$ dans $\mathbb{N}$ forme un sous-groupe du groupe des permutations de $\mathbb{N}$.
2. Soit $f \in \mathcal{F}_1$ une fonction récursive (totale), mais non récursive primitive, et soit i l'indice d'une machine de Turing $\mathcal{M}$ qui calcule f. On considère la fonction T qui à x associe le temps de calcul de $f(x)$ par $\mathcal{M}$, i.e. $T(x) = \mu t[(i, t, x) \in B^1]$, avec les notations du cours. Montrer que le graphe de T est récursif primitif. En déduire qu'il n'y a pas de fonction récursive primitive $g \in \mathcal{F}_1$ telle que $g(x) \geq T(x)$ pour tout $x \in \mathbb{N}$.
3. On pose $g_0(x) = \sup\{T(y); y \leq x\} + 2x$. Montrer que g_0 est récursive, mais non primitive récursive, et que le graphe G_0 de g_0 ainsi que l'image I_0 de g_0 sont des ensembles primitifs récursifs.

4. Montrer qu'il existe une (unique) fonction strictement croissante g_1 récursive primitive dont l'image soit égale à $\mathbb{N} \setminus I_0$.
5. Montrer que la fonction $h \in \mathcal{F}_1$ donnée par $h(2x) = g_0(x)$, $h(2x+1) = g_1(x)$ est bijective, récursive, mais non récursive primitive. Montrer que son inverse h^{-1} est récursive primitive.

Exercice 6 (Complexité) **:**
Étant donnée une machine de Turing $\mathcal{M}$ et une entrée n, on définit la complexité de $(\mathcal{M}, n)$ comme étant le nombre d'états de $\mathcal{M}$ plus $\log(n)$. La complexité d'un entier N est la complexité minimale d'un couple $(\mathcal{M}, n)$ tel que la machine de Turing $\mathcal{M}$ avec l'entrée n termine et renvoie N en sortie.

1. Soit g la fonction qui à N associe sa complexité. Montrer que g tend vers $+\infty$.
2. Soit f une fonction récursive tendant vers $+\infty$. Montrer qu'il existe N tel que $f(N) > g(N)$. En particulier, g n'est pas récursive.

Exercice 7 (Le retour des structures finies) **:**
Soit $\mathcal{L}$ un langage fini et T une $\mathcal{L}$-théorie finie, montrer que l'ensemble :

$$\{n \in \mathbb{N} \mid \exists M \models T,\ |M| = n\}$$

est primitif récursif.

Exercice 8 (Ensembles des fonctions récursivement énumérables) **:**
Soit $\mathfrak{F} \subseteq \mathcal{F}^1$, on dit que $\mathfrak{F}$ est récursivement énumérable si il existe $\psi \in \mathcal{F}^2$ récursive telle que

$$\{\psi_i \mid i \in \mathbb{N}\} = \mathfrak{F}.$$

On a montré dans l'exercice 4 que l'ensemble des fonctions primitives récursives à une variable est récursivement énuméré.

1. Montrer que l'ensemble des fonctions récursives totales à une variable n'est pas récursivement énuméré.
2. Montrer que l'ensemble des fonctions récursives primitives strictement croissantes (en une variable) est récursivement énuméré.
3. Montrer que l'ensemble des fonctions récursives primitives injectives (en une variable) est récursivement énuméré.
4. Montrer que l'ensemble des fonctions récursives totales injectives (respectivement strictement croissantes) en une variable n'est pas récursivement énuméré.