Examen de *FDI2*
Natacha Portier, le 26 mai 2009

Notes de cours et de TD interdites
Rédaction soignée indispensable ! Les résultats du cours peuvent être utilisés à condition de les énoncer soigneusement. Les résultats de TD doivent être redémontrés.

Notation :
Soit $(M_i)_{i\in\mathbb{N}}$ une énumération des machines de Turing qui soit un système acceptable de programmation. Pour tout i, on appelle φ_i la fonction calculée par M_i.

Exercice 1 :
Question 1,1
Énoncer le théorème de compacité du calcul propositionnel.
Question 1,2
Énoncer le théorème de complétude du calcul propositionnel.
Question 1,3
Admettre le théorème de complétude du calcul propositionnel et montrer le théorème de compacité du calcul propositionnel.

Exercice 2 : équivalence
On fixe le langage égalitaire $\mathcal{L} = (R; =)$ où R est un prédicat binaire. On rapelle qu'une théorie est un ensemble de formules closes du premier ordre sur $\mathcal{L}$.
Question 2,1
Existe t-il une théorie des relations d'équivalence n'ayant que des classes finies, i.e. une théorie dont l'ensemble des modèles soit exactement l'ensemble des ensembles munis d'une relation d'équivalence n'ayant que des classes finies ?
Question 2,2
Existe t-il une théorie des relations d'équivalence n'ayant qu'un nombre fini de classes ?
Question 2,3
Donner une axiomatisation de la théorie des relations d'équivalences ayant une infinité de classes infinies.
Question 2,4
Une théorie T est finiment axiomatisable s'il existe une théorie finie T_1 telle que $T \vdash T_1$ et $T_1 \vdash T$. Montrer que dans ce cas, il existe un sous-ensemble fini T' de T tel que $T' \vdash T$.

Question 2,5
La théorie des relations d'équivalences ayant une infinité de classes infinies est-elle finiment axiomatisable ?

Exercice 3 :
Donner une définition de récursivement énumérable. Montrer qu'il existe un ensemble qui n'est pas récursivement énumérable.

Exercice 4 : les numéros minimaux
Soit A l'ensemble des entiers i tel que pour tout entier $j < i$ on ait $\varphi_i \neq \varphi_j$. On peut voir A comme l'ensemble des codes minimaux qui calculent une fonction.

Question 4,1
Énoncer le théorème du point fixe effectif, appelé aussi deuxième théorème de la récursion de Kleene.

Question 4,2
Supposons A récursif. Montrer qu'il existe un entier c tel que la machine M_c fasse la chose suivante : sur une entrée ω, trouver un entier $d > c$ dans A et calculer $\varphi_d(\omega)$.

Question 4,3
Montrer, au choix, que A est récursif ou que A n'est pas récursif.

Question 4,4
Montrer, au choix, que A n'est pas récursivement énumérable ou que A est récursivement énumérable.

Question 4,5
Montrer, au choix, que tout sous-ensemble infini de A n'est pas récursivement énumérable ou que tout sous-ensemble infini de A est récursivement énumérable.

Exercice 5 :
Montrer qu'il existe deux entiers i et j différents tels que pour toute entrée ω on ait $\varphi_i(\omega) = j$ et $\varphi_j(\omega) = i$. On peut remarquer par exemple que M_i est alors une machine qui sur toute entrée calcule le numéro de la machine qui sur toute entrée calcule i.

Exercice 6 :
Les ensembles suivants sont-ils récursifs ? récursivement énumérables ? co-récursivement énumérables (c'est-à-dire de complémentaire récursivement énumérable) ?

1. $\{i \in \mathbb{N}, \varphi_i(676)$ n'est pas défini $\}$
2. $\{x \in \mathbb{N}, \varphi_{676}(x) = 42\}$
3. $\{\langle i, j\rangle, \varphi_i = \varphi_j\}$
4. $\{i \in \mathbb{N}, \varphi_i(x) = 676$ si $\varphi_x(x)$ s'arrête et 42 sinon$\}$

Exercice 7 : LBA

Question 7,1
Soit L_H l'ensemble des (M, ω) où M est une machine de Turing à un ruban acceptant le mot ω, i.e. s'arrêtant dans un état acceptant. Montrer que ce problème est indécidable sans utiliser de théorème du cours.

Question 7,2
Un automate linéaire borné (LBA) est une machine de Turing dont la tête de lecture ne peux sortir de la portion de ruban qui a contenu l'entrée. On peut le formaliser de la manière suivante. Supposons que Σ soit l'alphabet d'entrée de la machine. L'alphabet d'écriture de la machine contient alors deux lettres G et D qui ne sont pas dans Σ et qui jouent le rôle de délimiteurs. Un calcul de LBA commence alors nécessairement par le fait de placer un G juste à la gauche du mot d'entrée et un D juste à la droite, puis à revenir se positionner sur la première lettre de l'entrée sans avoir modifié d'autres cases du ruban. Par la suite, la machine reste entre ces deux délimiteurs : quelque soient les états q et q', la lettre a et le déplacement ε de la tête de lecture, si la fonction de transition δ de la machine vérifie $\delta(q, G) = (q', a, \varepsilon)$ alors $a = G$ et $\varepsilon = 1$ et si $\delta(q, D) = (q', a, \varepsilon)$ alors $a = D$ et $\varepsilon = -1$. On voit alors que l'ensemble des machines M qui sont des LBA est calculable.

Montrer que l'ensemble des couples (M, ω), où M est un LBA et ω un mot d'entrée accepté par M, est calculable.

Question 7,3
Soit M une machine de Turing fixée à un ruban et ω un mot d'entrée pour M. Soit $\#$ un symbole qui ne soit pas dans les alphabets de M. Soit $L_{M,\omega}$ l'ensemble des $\#C_0\#C_1\#\dots\#C_l\#$ tels que l soit un entier, chaque C_i est un code d'une configuration finie de M, C_0 est le code de la configuration initiale de M sur ω, C_l est le code d'une configuration acceptante, et pour tout i on passe de C_i à C_{i+1} en un pas de calcul. Le langage $L_{M,\omega}$ est donc l'ensemble des calculs acceptants de la machine M sur l'entrée ω.

Expliciter un code pour les configurations finies de M. Montrer que $L_{M,\omega}$ est décidable par un LBA, et que ce LBA est calculable à partir de M.

Question 7,4
Soit E_{LBA} l'ensemble des M tels que M soir un LBA qui reconnaisse le langage vide. Montrer que E_{LBA} n'est pas calculable en utilisant les questions précédentes.