Calculabilité et complexité

Université de Montpellier
Examen partiel
9 novembre 2015

Durée 2 heures 30 minutes
Aucun document n'est autorisé
Pas de calculatrice, téléphone portable, montre programmable,
appel à un ami, consultation de l'avis du public, *etc.*

Justifiez vos réponses avec soin !

Exercice 1 échauffement

On note $A + B = \{x + y,\ x \in A \text{ et } y \in B\}$.
Montrez que si A et B sont énumérables, alors $A + B$ l'est aussi.

Exercice 2 archi-classique

Soit $A = \{x,\ [x \mid \cdot]$ est une bijection sur les entiers naturels$\}$.

1. En utilisant avec soin le théorème de Rice, montrez que A n'est pas récursif.
2. Sans utiliser le théorème de Rice, montrez que A n'est pas récursif.

Exercice 3 classique

Soit $A = \{x,\ [x \mid 0] \downarrow \text{ et } [x \mid 1] \uparrow\}$.
A et son complémentaire $\bar{A}$ sont-ils récursifs ? énumérables ?

Exercice 4 facile

Soit g une fonction calculable.

1. Montrez qu'il existe une fonction calculable *totale* G telle que $[G(n) \mid \cdot] = n + g(.)$
2. Montrez que $\exists n\ [n \mid \cdot] = n + g(.)$.

Exercice 5 tout aussi aisé

1. Montrez qu'il existe une fonction calculable *totale* f telle que $[f(n) \mid \cdot] = [n \mid \cdot] + 1$
2. Quelles fonctions sont calculées par les points fixes de f ?

Exercice 6 à peine plus dur

1. Montrez qu'il existe 2 programmes consécutifs qui calculent la même fonction.
2. Proposez un système de programmation dans lequel 3 programmes consécutifs ne calculent jamais la même fonction.

Exercice 7 un peu moins dur

On s'intéresse au problème de décision suivant :
Tiling-with-borders
Entrée : Un ensemble fini de tuiles τ et une suite de couleurs s.
Question : Existe-t-il un pavage d'un carré par les tuiles de τ tel que la suite des couleurs des bords du carré soit exactement s ?

Montrez que ce problème est NP-complet.

Exercice 8 Alice prend le tramway

En 2031, dans l'état du Capitaland, Alice est employée comme programmeuse[1] : son travail consiste à réécrire plus clairement, en langage *C+++*, et en utilisant des algorithmes rénovés, des programmes écrits en *Fortran 77* qui datent des années 80. L'entreprise qui emploie Alice est rachetée par Bob. Bob propose au syndicat majoritaire de l'entreprise de négocier un avenant aux contrats des programmeurs : un employé sera licencié si et seulement si Bob parvient à prouver que le programme que l'employé a écrit calcule quelque chose de différent de ce que calculait le programme initial. Dans la négociation, le syndicat exige que les preuves de Bob soient exprimées dans ZFC, ce que Bob accepte. Pour profiter du "seulement si" de la clause, le syndicat signe l'avenant.

Alice paniquée se souvient de l'éxercice suivant, quelle avait résolu à l'automne 2015 en partiel de calculabilité :

1. Montrez qu'il n'existe pas de programme qui prend en entrée $< x, y >$ et qui décide si $[x \mid \cdot] = [y \mid \cdot]$

Elle se précipite alors à l'Université pour consulter son (très) vieux professeur de calculabilité et lui demande si, par hasard il n'aurait pas une idée pour empêcher Bob de la licencier trop facilement. Celui-ci lui répond :
"Ah ah ah, et vous auriez donc suivi mon cours en 2015 ?! Mais si vous aviez correctement composé au partiel, vous sauriez qu'il existe un programme-miracle : quelque soit le programme en *Fortran 77* qu'on vous donne à réécrire, vous pouvez rendre à votre patron ce programme-miracle. Bien sûr, ce programme ne fait pas ce qu'il faut, mais il ne pourra jamais le prouver…".
Alice demande timidement où elle peut trouver le programme-miracle et le vieux professeur hausse les épaules : "Please Alice, use your brain !". Alice réfléchit dans le tramway, oublie de descendre à son arrêt, mais en arrivant au terminus s'écrie : "Nom d'un petit Kleene, j'ai trouvé !".

On rappelle que les théorèmes de ZFC constituent un ensemble énumérable (c'est assez évident : on construit les preuve dans le cadre de systèmes formels à partir d'un nombre fini d'axiomes, et en suivant des règles de déduction ; autrement dit il existe un programme qui permet de vérifer les preuves). L'expression ZFC $\vdash T$ signifie que ZFC permet de prouver la proposition T, ou, autrement dit, elle signifie que T est un théorème de ZFC.

2. En raisonnant par l'absurde, montrez que $\exists \mu \forall q$ ZFC $\nvdash [\mu \mid \cdot] \neq [q \mid \cdot]$. Indication : énumérez les preuves et trouvez une fonction récursive qui à un programme p associe un programme q qui ne calcule pas la même fonction (justifiez qu'on a alors une contradiction).

Un programme μ tel que défini ci-dessus est appelé *proramme-miracle*.

3. Quelles fonctions sont calculées par les programmes-miracles ?

On a ainsi prouvé l'existence d'un programme-miracle μ ; reste à le construire…

4. Proposez une fonction récursive dont les points fixes sont des programmes-miracles. On pourrait remarquer que si $[p \mid x] \neq [q \mid x]$ et que les deux convergent, alors il existe une preuve de ceci dans ZFC.
5. Montrez qu'un programme-miracle peut être trouvé récursivement en fonction du système de preuve considéré.

Epilogue : Alice a pu garder son emploi grâce à son cours de calculabilité d'une part, et son cerveau d'autre part…

[1] Je trouve ce terme hideux mais il semble que ce soit le terme politiquement correct…