

## Devoir maison, Modèles de calcul, 2011-12.

*On attachera une grande importance à la clarté et à la concision des justifications.*

*Rappels :*

- Un ensemble  $X \subseteq U$  est *dénombrable* s'il est fini, ou il existe une fonction  $f : \mathbb{N} \rightarrow U$  telle que  $f(\mathbb{N}) = U$ .
- Une fonction  $f : \mathbb{N}^k \rightarrow \mathbb{N}$  est *calculable*, s'il existe un programme WHILE (ou GOTO) qui la calcule.
- Un ensemble  $X \subseteq \mathbb{N}$  est *semi-décidable*, s'il existe un programme WHILE (ou GOTO) qui retourne 1 sur  $x \in \mathbb{N}$  si et seulement si  $x \in X$ . (Si  $x \notin X$  le programme peut soit ne pas terminer, ou retourner une valeur différente).
- Un ensemble  $X \subseteq \mathbb{N}$  est *décidable*, s'il existe un programme WHILE (ou GOTO) qui retourne 1 sur  $x \in \mathbb{N}$  si et seulement si  $x \in X$ , et qui *termine toujours*.
- Un *problème de décision* est défini par un ensemble d'instances positives (c-à-d, instances où la réponse est "oui").
- Pour les ensembles, les notions *récuratif* et *décidable* sont équivalentes. De même pour *récurivement-énumérable* et *semi-décidable*.
- Le problème universel *UNIV* est défini par :  $(n, m)$  est une instance positive si le programme  $P_n$  termine sur l'entrée  $m$ .
- Une réduction  $P_1 \leq P_2$  du problème  $P_1$  vers le problème  $P_2$  est une fonction calculable  $f$  qui transforme des instances de  $P_1$  en instances de  $P_2$  tel que  $x$  est une instance positive de  $P_1$  si et seulement si  $f(x)$  est une instance positive de  $P_2$ .

**Exercice 1** Est-ce que les ensembles suivants sont dénombrables? Justifiez votre réponse.

- L'ensemble des sous-parties de  $\mathbb{N}$ .
- L'ensemble des sous-parties *finies* de  $\mathbb{N}$ .
- L'ensemble des graphes orientés finis.
- L'ensemble des graphes orientés infinis.
- L'ensemble des formules booléennes propositionnelles en forme normale conjonctive (CNF). Exemple de formule en CNF :  $(x_1 \vee \neg x_3 \vee \neg x_5) \wedge (\neg x_1 \vee x_2 \vee x_3) \wedge (x_2 \vee x_5)$ .
- L'ensemble des programmes C.

**Exercice 2** Justifiez les assertions suivantes :

1. Toute fonction primitive-réursive est calculable.
2. Toute fonction totale  $f : \mathbb{N} \rightarrow \{0, 1\}$  telle que  $f(n) = 1$  ssi  $n \geq 42$  est primitive-réursive.
3. Toute fonction  $f : \mathbb{N} \rightarrow \{0, 1\}$  telle que  $f(n)$  n'est pas défini si  $n > 42$ , est calculable.

**Exercice 3** Montrez qu'un problème  $P$  est semi-décidable *si et seulement si* il existe une réduction  $P \leq UNIV$ . Justifiez bien votre réponse.

**Exercice 4** Dans cet exercice on montre que tout programme GOTO  $\Pi$  qui calcule une fonction  $f : \mathbb{N} \rightarrow \mathbb{N}$  est équivalent à un programme GOTO  $\Pi'$  qui

- utilise seulement 2 variables  $x, y$ ; au début,  $x$  contient l'entrée et  $y = 0$ ,
- les seules instructions arithmétiques de  $\Pi'$  sont  $z := z + 1$  (incrément) et  $z := z - 1$  (décrement, si  $z \neq 0$ ).

1. Montrer comment simuler les instructions  $x_i := x_j + c$  et  $x_i := x_j - c$  (où  $c > 1$ ) par des instructions  $x_i := x_j + 1$  et  $x_i := x_j - 1$ .
2. Montrer comment simuler  $x_i := x_j + 1$  et  $x_i := x_j - 1$  (où  $i \neq j$ ) par un programme GOTO où les seules instructions arithmétiques sont  $x_k := x_k + 1$  et  $x_k := x_k - 1$ . Vous pouvez utiliser des variables additionnelles.
3. Montrer comment simuler un programme GOTO avec  $k$  variables et dont les seules instructions arithmétiques sont des incréments et décrements, par un programme GOTO avec 2 variables uniquement, et la même propriété pour les instructions arithmétiques.  
*Indication* : vous avez vu en TD comment coder  $\mathbb{N}^k$  dans  $\mathbb{N}$ . Il suffit de réaliser le codage/décodage par un programme GOTO.