

Modèles de Calcul : TD2

Ensembles semi-décidables, réductions, PCP

1 Discussion

Les questions suivantes ne sont pas des exercices. On vous propose d'y réfléchir en pensant aux algorithmes (en particulier, aux algorithmes d'énumération) d'une manière informelle. "Programme" signifie "programme WHILE".

Questions :

1. Soit un programme P . Comment le transformer en un autre programme P' qui simule P et qui calcule en même temps le nombre d'opérations faites par P ?
2. Comment simuler un programme P non pas sur une donnée particulière, mais sur toutes ses données ?
3. Comment simuler tous les programmes, sur toutes leurs données ?

2 Ensembles décidables et semi-décidables

Exercice 2.1

1. Justifiez que tout ensemble fini est décidable.
2. Soit X l'ensemble des suites de 5 chiffres décimaux qui sont des facteurs de longueur 5 du développement décimal du nombre π . Cet ensemble est-il décidable ?

Exercice 2.2

Est-ce que le programme suivant termine sur au moins une entrée ? sur toutes ses entrées ?

Fermat(N)

```
a := 1 ; d := 0 ;
while (d = 0) do
  b := 1 ;
  loop(a) do
    c := a ;
    loop(b) do
      if ( $a^N + b^N = c^N$ ) then d := 1 fi
      c := c + 1 ;
    od
    b := b + 1 ;
  od ;
  a := a + 1 ;
od
```

On dit qu'un programme P énumère un ensemble X si P , en travaillant sans arrêt, retourne au fur et à mesure tous les éléments de X . On ne demande pas que P s'arrête (même si X est un ensemble fini).

Exercice 2.3

1. Soit un ensemble semi-décidable $X \subseteq \mathbb{N}$. Décrivez un programme qui l'énumère.
2. Justifiez que tout ensemble semi-décidable $X \subseteq \mathbb{N}$ peut être énuméré sans répétitions.
3. Justifiez que tout ensemble décidable $X \subseteq \mathbb{N}$ peut être énuméré en ordre croissant.
4. Justifiez que tout langage décidable $L \subseteq \Sigma^*$ peut être énuméré en ordre lexicographique.

Rappel : Pour deux mots $u, v \in \Sigma^*$ on dit que $u < v$ en ordre lexicographique si $|u| < |v|$, ou $|u| = |v|$ et u précède v dans l'ordre du dictionnaire.

Exercice 2.4

Soit P un programme qui prend en entrée un entier.

- Montrez que l'ensemble des entiers sur lesquels P termine est semi-décidable.
- Montrez que l'ensemble des valeurs calculées par P est semi-décidable.

Exercice 2.5

Soit P un programme qui prend en entrée un entier et qui termine sur toute entrée.

1. On suppose que pour toute entrée n , la fonction $P(n)$ calculée par P sur n satisfait $P(n) \geq n$ pour tout n .
2. On suppose que P calcule une fonction strictement croissante.

Montrez que l'ensemble des valeurs calculées par P est un ensemble décidable, pour chacune des hypothèses ci-dessus.

Exercice 2.6

Soit X et Y deux ensembles semi-décidables. Comment construire deux ensembles semi-décidables et *disjoints* X' et Y' tels que $X' \subseteq X$, $Y' \subseteq Y$, et $X' \cup Y' = X \cup Y$?

Exercice 2.7

Montrez que pour tout ensemble semi-décidable $S \subseteq \mathbb{N}$, il existe un ensemble décidable $D \subseteq \mathbb{N}^2$ tel que

$$S = \{x \in \mathbb{N} \mid \exists y \in \mathbb{N} : (x, y) \in D\}.$$

3 Réductions

Exercice 2.8

On considère les problèmes suivants (dans les deux cas, les programmes prennent en entrée un seul entier) :

1. HALT_0 : étant donné un programme P , est-ce que P s'arrête sur 0 ?
2. VAL_0 : étant donné un programme P , est-ce que $P(0) = 0$?

Justifiez que les fonctions suivantes sont bien des réductions (calculables) de HALT_0 vers VAL_0 (fonction R_1) et vice-versa (fonction R_2) :

1. R_1 transforme un programme P en un programme P' défini comme il suit :

$P; \text{res} := 0$

2. R_2 transforme un programme P en un programme P' défini comme il suit (z étant une nouvelle variable, res la variable “résultat”) :

$P;$
 WHILE ($\text{res} \neq 0$) DO $z := z + 1$ OD

Exercice 2.9

Dans cet exercice on montre que le problème suivant est indécidable :

Entrée : programme P qui prend en entrée un entier, et entier n .

Question : est-ce que P termine sur toute entrée $m \leq n$?

Donnez une réduction de HALT_0 vers le problème ci-dessus.

Exercice 2.10

Dans cet exercice on montre que le problème suivant est indécidable :

Équivalence de programmes LOOP :

Entrée : deux programmes LOOP P, Q

Question : est-ce que P et Q sont équivalents (c-à-d, calculent la même fonction) ?

1. Etant donné un programme R et une entrée n pour R , montrez qu’il existe un programme LOOP P qui sur l’entrée k retourne 1 si R termine sur n après au plus k opérations, et 0 sinon.
2. Justifiez que l’équivalence de programmes LOOP est un problème indécidable.

Exercice 2.11

Pour tout entier n , on appelle P_n le programme dont le code est l’entier n . On suppose que les programmes ci-dessous prennent des entrées dans \mathbb{N} . Les problèmes suivants sont-ils décidables, semi-décidables ? (Justifiez chaque cas).

1. ENTRÉE : n
 QUESTION : $P_n(0)$ est-il défini ?
2. ENTRÉE : n
 QUESTION : $P_n(0) = 0$?
3. ENTRÉE : n
 QUESTION : P_n est-elle croissante ?
4. ENTRÉE : n
 QUESTION : $\text{Dom}(P_n)$ est fini ?
5. ENTRÉE : n
 QUESTION : $P_n(m) \leq m^2$ pour tout $m \in \mathbb{N}$?

4 Problème de Correspondance de Post

Exercice 2.12

On donne des instances du problème de correspondance de Post. Pour chaque instance donner une solution (toutes les instances en ont une) :

1. $u_1 = abc, u_2 = ca, u_3 = b$ et $v_1 = a, v_2 = bcca, v_3 = ccb$.
2. $u_1 = aab, u_2 = cdabdc, u_3 = d$ et $v_1 = aa, v_2 = a, v_3 = bcd$.
3. $u_1 = c, u_2 = cc, u_3 = ababa$ et $v_1 = ab, v_2 = cccc, v_3 = a$.

Exercice 2.13

Soit u, v deux mots. On écrit $u \subseteq v$ quand u peut s'obtenir à partir de v en effaçant des lettres (par exemple $aaa \subseteq ababca$). On considère le problème d'inclusion de Post qui est une version légèrement modifiée du problème de correspondance :

ENTRÉE : $(u_1, v_1), \dots, (u_n, v_n)$
QUESTION : $\exists i_1, \dots, i_k$ tels que $u_{i_1} \cdots u_{i_k} \subseteq v_{i_1} \cdots v_{i_k}$.

1. Soient u, u', v et v' des mots, montrez que si $uu' \subseteq vv'$ alors $u \subseteq v$ ou $u' \subseteq v'$.
2. Montrez que si le problème d'inclusion de Post a une solution alors il a une solution de longueur 1.
3. Montrez que le problème d'inclusion de Post est décidable.

Exercice 2.14

On considère les grammaires hors-contexte. Une grammaire hors-contexte G est donnée par :

- Un alphabet terminal $\Sigma = \{a_1, \dots, a_n\}$.
- Un alphabet non-terminal $V = \{A_1, \dots, A_m\}$.
- Un ensemble de règles de la forme : $A \rightarrow \alpha$ avec $A \in V$ et $\alpha \in (\Sigma \cup V)^*$.

On définit l'ensemble $L(G)$ des mots générés par G comme suit : pour tout $u \in \Sigma^*$ on a $u \in L(G)$ si on peut obtenir u à partir de A_1 en utilisant les règles de G .

On demande de montrer que le problème suivant est indécidable :

ENTRÉE : Deux grammaires hors-contexte G_1, G_2
QUESTION : $L(G_1) \cap L(G_2) = \emptyset$?