TD3

Chargé de TD : Adrien Koutsos
adrien.koutsos@lsv.fr
http://www.lsv.fr/~koutsos

1 Diagonalisations

Montrer par un argument diagonal (et donc sans réduction) que les langages suivants sont indécidables :

Exercice 1
$L_{\text{diag}} = \{w_i \mid w_i \notin \mathcal{L}(M_i)\}$

Exercice 2
$L_{\text{accepte}} = \{< M, w > \mid M \text{ accepte } w\}$

Exercice 3
$L_{\text{arret}} = \{< M, w > \mid M(w) \neq \bot\}$

2 Vers la réduction

Exercice 4
En supposant que le langage L_{arret} est indécidable (et seulement celui-ci), montrez que $L_{\emptyset} = \{< M > \mid \mathcal{L}(M) = \emptyset\}$ l'est également. Vous raisonnerez par l'absurde et supposerez l'existence d'une machine de Turing $M_{\emptyset}$ décidant $L_{\emptyset}$. Vous construirez ensuite une machine M décidant L_{arret} en utilisant $M_{\emptyset}$.

Exercice 5
En supposant que le langage L_{accepte} est indécidable (et seulement celui-ci), montrez que L_{arret} l'est également. Vous raisonnerez par l'absurde et supposerez l'existence d'une machine de Turing M_{arret} décidant L_{arret}. Vous construirez ensuite une machine M décidant L_{accepte} en utilisant M_{arret}.

3 Réductions

Les preuves des exercices 4 et 5 ont toutes un point commun expliqué ci-dessous. On dit que ce sont des preuves **par réduction**.

Rappel : Si on considère deux problèmes, $\mathfrak{A}$ et $\mathfrak{B}$, on dit que $\mathfrak{A}$ se réduit à $\mathfrak{B}$ (noté $\mathfrak{A} \preccurlyeq \mathfrak{B}$) si on peut exhiber une fonction **c**alculable f qui **pour toute** instance a de $\mathfrak{A}$, renvoie une instance $b = f(a)$ de $\mathfrak{B}$, telle que *a est une instance acceptante de* $\mathfrak{A}$ **si et seulement si** *b est une instance acceptante de* $\mathfrak{B}$. **Mais**, il n'est pas nécessaire que **toutes** les instances de $\mathfrak{B}$ soient dans l'image de f !

Si on a $\mathfrak{A} \preccurlyeq \mathfrak{B}$, alors :
— si $\mathfrak{A}$ est indécidable, alors $\mathfrak{B}$ est indécidable ;
— si $\mathfrak{B}$ est décidable, alors $\mathfrak{A}$ est décidable.

Exercice 6
Dire si les problèmes suivants sont décidable ou non. Si c'est le cas, donnez l'idée de la machine de Turing décidant le langage et sinon, faites une preuve **par réduction**.

1. **Donnée :** le code $< M >$ d'une machine de Turing
   Question : M s'arrête-t-elle sur le mot vide ?
2. **Donnée :** le code $< M >$ d'une machine de Turing
   Question : M s'arrête-t-elle sur au moins une donnée ?
3. **Donnée :** les codes $< M >$ et $< M' >$ de deux machines de Turing
   Question : $L(M) = L(M')$?
4. **Donnée :** le code $< M, w >$ d'une machine de Turing et d'un mot w et un entier n (en base 2)
   Question : M accepte-t-elle w après au plus n transitions ?
5. **Donnée :** le code $< M >$ d'une machine de Turing
   Question : M calcule en temps polynomial (i.e. M termine en temps polynomial) ?
6. **Donnée :** les codes $< M >$ d'une machine de Turing et le code $< M' >$ d'une machine de Turing qui s'arrête pour tout mot w en au plus $2.|w|$ transitions
   Question : Pour tout mot w, $M(w) = M'(w)$?

Exercice 7 (automates linéairement bornés)
Un *automate linéairement borné* est une machine de Turing qui lorsqu'elle lit un blanc, écrit un blanc et se déplace vers la gauche.

1. Montrer que le problème de l'arrêt des automates linéairement bornés est décidable.
2. Montrer que le problème de l'arrêt universel des automates linéairement bornés est indécidable :
   Donnée : $< M >$ où M est un automate linéairement borné
   Question : est ce que M s'arrête sur toute donnée ?

Exercice 8 (Fonctions calculables et langages récursifs)
Donner une fonction calculable dont l'image n'est pas récursive.

Exercice 9 (Systèmes de réécriture)
Un système de réécriture de mots sur l'alphabet Σ est donné par un ensemble fini de paires $(u, u') \in \Sigma^* \times \Sigma^*$. La relation de réécriture associée est $\rightarrow_R \subseteq \Sigma^* \times \Sigma^*$ définie par

$$w \rightarrow_R w' \text{ ssi } \exists (u, u') \in R.\ \exists w_1, w_2 \in \Sigma^*.\ w = w_1 u w_2 \wedge v = w_1 u' w_2$$

La relation de réduction $\rightarrow_R^*$ est la clôture réflexive transitive de la relation de réécriture. Par exemple, avec

$$R = \{(c, abaca), (c, bcbabab), (aca, d), (bcb, d), (ada, d), (bdb, d)\}$$

on a $c \rightarrow_R^* d$:

$$\overline{c} \rightarrow_R \underline{aba\overline{c}a} \rightarrow_R aba\underline{b\overline{c}babab}a \rightarrow_R abab\underline{ab\overline{aca}}bababa \rightarrow_R ababa\overline{b\underline{d}b}ababa$$

$$\rightarrow_R abab\overline{a\underline{d}a}baba \rightarrow_R aba\overline{b\underline{d}b}aba \rightarrow_R ab\overline{a\underline{d}a}ba \rightarrow_R a\overline{b\underline{d}b}a \rightarrow_R \overline{a\underline{d}a} \rightarrow_R \underline{d}$$

Montrer que le problème d'*accessibilité* est indécidable :
Donnée: un système de réécriture R et deux mots $u, v \in \Sigma^*$;
Question: est-ce que $u \rightarrow_R^* v$?