

# Calculabilité

## TD n° 3

Louis Lemonnier  
lemonnier@lsv.fr

6 octobre 2022

## 1 Intro

### Exercice 1 : Diagonalisation

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

1.  $L_{\text{accepte}} = \{\langle M \rangle; w \mid M \text{ accepte } w\}$ ,
2.  $L_{\text{arret}} = \{\langle M \rangle; w \mid M \text{ termine avec l'entrée } w\}$ .

### Exercice 2 : Vers la réduction

En supposant que le langage  $L_{\text{arret}}$  est indécidable (et seulement celui-ci), montrez que  $L_{\emptyset} = \{\langle M \rangle \mid \mathcal{L}(M) = \emptyset\}$  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  $L_{\text{arret}}$  en utilisant  $M_{\emptyset}$ .

### Exercice 3 : Vers la réduction (2)

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

## 2 Les choses sérieuses

Les preuves des exercices 2 et 3 ont toutes un point commun expliqué ci-dessous. On dit que ce sont des preuves **par réduction**.

**Rappel (ou non) :** Si on considère deux problèmes,  $\mathfrak{A}$  et  $\mathfrak{B}$ , on dit que  $\mathfrak{A}$  se réduit à  $\mathfrak{B}$  (noté  $\mathfrak{A} \preceq \mathfrak{B}$ ) si on peut exhiber une fonction **calculable**  $f$  qui **pour toute** instance  $a$  de  $\mathfrak{A}$ , telle que  $a$  est une instance acceptante de  $\mathfrak{A}$  **si et seulement si** le calcul de  $f$  sur  $a$  accepte **ou**  $f(a)$  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  $\mathfrak{A} \preceq \mathfrak{B}$ , alors :

- si  $\mathfrak{A}$  est indécidable, alors  $\mathfrak{B}$  est indécidable ;
- si  $\mathfrak{B}$  est décidable, alors  $\mathfrak{A}$  est décidable.

**Exercice 4 : Est-ce décidable ?**

Dire si les problèmes suivants sont décidables 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  $\langle M \rangle$  d'une machine de Turing,  
**Question :**  $M$  s'arrête-t-elle sur le mot vide ?
2. **Donnée :** le code  $\langle M \rangle$  d'une machine de Turing,  
**Question :**  $M$  s'arrête-t-elle sur au moins une donnée ?
3. **Donnée :** les codes  $\langle M \rangle$  et  $\langle M' \rangle$  de deux machines de Turing,  
**Question :**  $L(M) = L(M')$  ?
4. **Donnée :** le code  $\langle M, w \rangle$  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  $\langle M \rangle$  d'une machine de Turing,  
**Question :**  $M$  calcule en temps polynomial (i.e.  $M$  termine en temps polynomial) ?
6. **Donnée :** les codes  $\langle M \rangle$  d'une machine de Turing et le code  $\langle M' \rangle$  d'une machine de Turing qui s'arrête pour tout mot  $w$  en au plus  $2 \cdot |w|$  transitions,  
**Question :** Pour tout mot  $w$ ,  $M(w) = M'(w)$  ?

**Exercice 5 : 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 :**  $\langle M \rangle$  où  $M$  est un automate linéairement borné,  
**Question :** Est-ce que  $M$  s'arrête sur toute donnée ?

**Exercice 6 : Systèmes de réécriture**

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

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

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  $c \rightarrow_R^* d$  :

$$\begin{aligned} \bar{c} &\rightarrow_R \underline{aba\bar{c}a} \rightarrow_R \underline{abab\bar{c}bababa} \rightarrow_R \underline{ababab\bar{c}abababa} \rightarrow_R \underline{ababab\bar{d}bababa} \\ &\rightarrow_R \underline{ababab\bar{d}ababa} \rightarrow_R \underline{abab\bar{d}baba} \rightarrow_R \underline{ab\bar{d}aba} \rightarrow_R \underline{ab\bar{d}ba} \rightarrow_R \underline{a\bar{d}a} \rightarrow_R \underline{d} \end{aligned}$$

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  $u \rightarrow_R^* v$  ?

**3 Cadeau****Exercice 7 : Fonctions calculables et langages récursifs**

Donner une fonction calculable dont l'image n'est pas récursive.