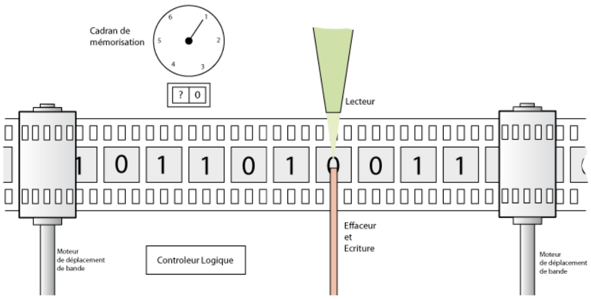
TD Machines de Turing et langages

Olivier Raynaud (raynaud@isima.fr)



Définition 1. *Un langage L est dit récursif* ***si et seulement si*** *il existe une machine de Turing M avec $L = L(M)$ et telle que :*
- *si $w \in L$ alors M accepte w (s'arrête) ;*
- *si $w \notin L$ M s'arrête dans un état non final.*

Question 1. *Montrer que si un langage L est récursif alors son complémentaire $\overline{L}$ est aussi récursif.*

Question 2. *Montrer que si un langage L et son complémentaire $\overline{L}$ sont récursivement énumérables alors L et $\overline{L}$ sont récursifs.*

Exercice 1 (Le langage Diagonal - L_d)**.**

Dans la suite on supposera qu'on dispose d'un codage pour les machines de Turing. Ainsi w_i codera pour la machine M_i.

Définition 2. *Le langage L_d (langage Diagonal) est l'ensemble des chaÓnes w_i telles que $w_i \notin L(M_i)$.*

Question 1. *Montrer que le langage L_d n'est pas récursivement énumérable.*

Exercice 2 (Le langage Universel - L_u).

Définition 3. *Le langage L_u (langage Universel) est l'ensemble des chaÓnes codant pour un couple (M, w) composé d'une machine de Turing M et d'une entrée w reconnue par la machine M.*

Question 1. *Montrer que le langage L_u est récursivement énumérable mais pas récursif.*

Question 2. *Montrer que l'ensemble $H(M)$ des paires (M, w) telles que la machine de Turing M s'arrête pour l'entrée w (en acceptant ou en refusant cette entrée) est récursivement énumérable mais pas récursif.*

Exercice 3 (Réductions de problèmes).

On réduit un problème P_1 en un problème P_2 en produisant une fonction f transformant les instances de P_1 en instances de P_2 de telle sorte que les réponses coÔncident.

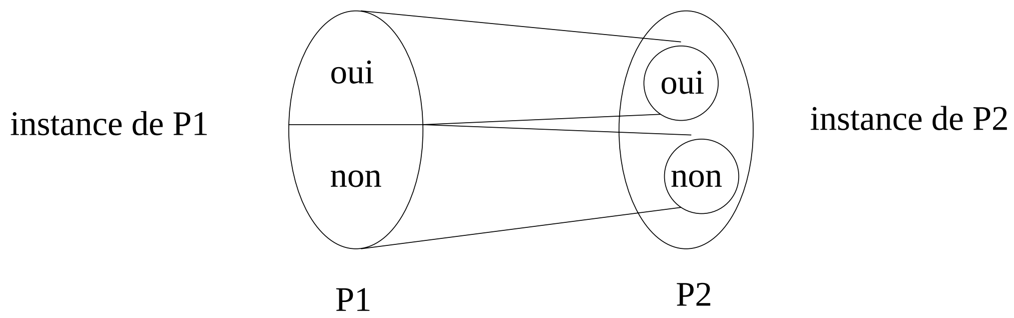


FIGURE 1 – Une réduction d'un problème P_1 en un problème P_2. Soit w_1 une instance de P_1, on doit avoir $P_1(w_1)$ est vrai **si et seulement si** $P_2(f(w_1))$ est vrai.

Question 1. *Montrer que s'il existe une réduction d'un problème P_1 vers un problème P_2 alors :*

- *si P_1 n'est pas décidable alors P_2 n'est pas décidable ;*
- *si P_1 n'est pas semi-décidable alors P_2 n'est pas semi-décidable.*

Exercice 4 (Le langage vide - L_e).

Définition 4. *Soit w_i le code d'une machine de Turing M_i :*

- *si $L(M_i) = \emptyset$ alors M_i n'accepte aucune entrée et $w_i \in L_e$;*
- *si $L(M_i) \neq \emptyset$ alors $w_i \in L_{ne}$*

Ainsi $L_e = \{w_i \mid L(M_i) = \emptyset\}$ et $L_{ne} = \{w_i \mid L(M_i) \neq \emptyset\}$. Notons que $\overline{L}_e = L_{ne}$.

Question 1. *Classer les langages L_e et L_{ne}. Lequel est le plus simple.*