Modèles de calcul - TD5

Exercice 1 Exemples de langages R et RE

1. Montrer que $L_1 := \{w \mid w \text{ est un palindrome}\}$ est décidable.
2. Montrer que $L_2 := \{\langle M, w\rangle \mid M \text{ s'arrête sur } w\}$ est récursivement énumérable.

Solution de l'exercice 1

1. Déjà vu en TD. Écrire un algo, dire qu'on peut le transformer en MT (sans décrire une vraie MT). Dire que ça s'arrête toujours avec la bonne réponse.
2. Écrire un programme qui lance M sur w, et accepte ensuite. Si $\langle M, w\rangle \in L_2$, la MT associée à ce programme accepte. Sinon, la MT boucle. C'est bien la def de RE.

Exercice 2 Opérations sur les langages

Soient L_1 et L_2 deux langages décidables sur l'alphabet Σ.

1. Montrer que $L_1 \cap L_2$, $L_1 \cup L_2$ et $\bar{L_1}$ sont décidables.
2. On suppose maintenant que L_1 et L_2 sont récursivement énumérables. Que peut-on dire de $L_1 \cap L_2$, $L_1 \cup L_2$ et $\bar{L_1}$?

Solution de l'exercice 2

1. On fait tourner les MT l'une après l'autre (ça termine toujours), et on fait les opérations logiques.

2. Pour $\cup$ et $\cap$, c'est dans RE : faire tourner les MT en parallèle pas à pas. Par contre le complémentaire peut ne pas être dans RE : prendre un langage récursivement énumérable et non décidable (on a vu en cours que ça existait, et on va remettre une couche à l'exercice suivant), son complémentaire n'est pas récursivement énumérable.

Exercice 3 $RE \cap co-RE$

Montrer que si L et $\bar{L}$ sont récursivement énumérables, alors L est décidable.

Solution de l'exercice 3

Faire tourner les deux MT en parallèle, chacune sur une bande, s'arrêter dès qu'une s'arrête.

Exercice 4 Énumérateurs

Un *énumérateur* est une MT déterministe qui liste des mots de Σ^* séparés par un symbole spécial \$.

1. Montrer qu'un langage L est décidable si et seulement s'il existe un énumérateur qui écrit les mots de L par taille croissante (et aucun mot n'appartenant pas à L).
2. Montrer que L est récursivement énumérable si et seulement s'il existe un énumérateur M qui n'écrit que des mots de L et tel que tout mot de L sera écrit à un moment de l'exécution de M.

Solution de l'exercice 4

1. Si L est décidable, on teste les mots par ordre de taille croissant, et on écrit seulement ceux qui sont bon. Dans l'autre sens, si on a un énumérateur, il suffit d'énumérer les mots jusqu'à, ou bien trouver le mot que l'on cherche, et sortir oui, ou bien commencer à énumérer des mots plus grands, et alors on répond non.

2. Si on a un énumérateur, c'est facile de construire une MT qui accepte les mots du langage : sur l'entrée w, on simule l'énumérateur jusqu'à obtenir w. Si on y arrive on accepte, sinon, on continue la simulation. C'est une MT qui accepte exactement les mots de L mais qui ne s'arrête pas si $w \notin L$.

   Dans l'autre sens, c'est un peu plus technique. Soit M une MT pour L. On va tester pour $i = 1, 2, \ldots$, pour chaque mot w de longueur au plus i, si M termine sur w en i étapes. Si c'est le cas, et que la machine répond oui, on l'écrit. Ainsi tout mot, de longueur ℓ sur lequel la machine s'arrête après k pas, avec un accept, sera listé lors de l'itération $\max(\ell, k)$. Et aucun autre mot ne sera listé.