

TD 1

Calculabilité

Luc Lapointe

luc.lapointe@ens-paris-saclay.fr
home.lmf.cnrs.fr/LucLapointe/

Exercice 1 - Machines de Turing

1. Construire une machine de Turing reconnaissant le langage des palindromes.
2. Construire une machine de Turing calculant $n + 1$, pour n donné en binaire avec le bit de poids fort à gauche. n peut contenir des 0 non significatifs.
3. Donner explicitement la table d'une machine de Turing qui, étant donné un mot $w \in \{0, 1\}^*$, accepte w si w contient au moins autant de 0 que de 1 et rejette sinon. On prendra soin de démontrer que la machine fait bien ce qu'elle est censée faire.

Exercice 2 - Questions existentielles

Parmi les trois fonctions suivantes, deux sont calculables, et le problème de la calculabilité de la troisième est un (célèbre) problème ouvert. Lesquelles ?

$$f_1(n) = \begin{cases} 0 & \text{si } P = NP \\ 1 & \text{sinon.} \end{cases}$$

$$f_2(n) = \begin{cases} 0 & \text{si les décimales de } \pi \text{ contiennent la sous-séquence } 1^n \\ 1 & \text{sinon.} \end{cases}$$

$$f_3(n) = \begin{cases} 0 & \text{si les décimales de } \pi \text{ contiennent} \\ & \text{une sous-séquence maximale de 1 de longueur } n \\ 1 & \text{sinon.} \end{cases}$$

Exercice 3 - Castors affairés

Pour $n \in \mathbb{N}$, soit \mathcal{E}_n l'ensemble des machines de Turing à ruban bi-infini, sur l'alphabet $\{1, B\}$, à n états (sans compter **accept** ni **reject**), et qui acceptent le mot vide.

Si $M \in \mathcal{E}_n$, on note $f(M)$ le nombre de 1 inscrits sur le ruban quand, sur la donnée ε , M s'arrête en acceptant. On considère la fonction

$$\text{SCORE}(n) = \max\{f(M) \mid M \in \mathcal{E}_n\}.$$

1. Calculer $\text{SCORE}(2)$.
2. Montrer que SCORE n'est pas calculable.
3. Montrer que $\text{SCORE}(3) \geq 6$.¹

Exercice 4 - Deux états

1. Montrer que toute fonction calculable est calculable par une machine de Turing sur l'alphabet $\{0, 1, B, \$\}$.
2. Montrer que toute fonction calculable est calculable par une machine de Turing à deux états (sans compter **accept** et **reject**).
3. Peut-on réaliser les deux simplifications à la fois ?

¹En fait, on a égalité.