TD 9 – Complexité en temps

1 Problèmes de factorisation

On considère les trois problèmes suivants :

— FINDSMALLESTFACTOR Étant donné $N \in \mathbb{N}_{>0}$ renvoie le plus petit $k \in \mathbb{N}$ avec $k \geq 2$ tel que k divise N.

— FINDGREATESTFACTOR Étant donné $N \in \mathbb{N}_{>0}$ renvoie le plus grand $k \in \mathbb{N}$ avec $k < N$ tel que k divise N.

— HASFACTOR Étant donné $(N, M) \in (\mathbb{N}_{>0})^2$ décide s'il existe un facteur non trivial de N plus petit que M.

Exercice 1. Montrer que si l'on sait résoudre un de ces problèmes en temps polynomial, alors on sait résoudre les trois. Attention, le temps polynomial se réfère à la taille de l'entrée qui est logarithmique en les nombres représentés.

Pour la suite, vous pouvez supposer que tester la primalité d'un nombre est dans P.

Exercice 2. Montrer que HASFACTOR est dans NP.

Exercice 3. Montrer que HASFACTOR est dans coNP.

2 Coloriabilité

Étant donné un graphe non-orienté $G = (V, E)$, un *k-coloriage* de G, pour $k \in \mathbb{N}_{>0}$, est une application $c\colon V \to [\![1, k]\!]$ associant une couleur (un nombre) de 1 à k à chaque sommet de G, de telle manière que deux sommets x et y adjacents (c'est-à-dire, tels que $\{x, y\} \in E$) n'aient jamais la même couleur ; lorsqu'un tel k-coloriage existe, on dit que G est *k-coloriable*.

Exercice 4. Montrer que le langage $3\mathrm{COL} = \{\langle G\rangle \mid G$ graphe non orienté 3-coloriable$\}$ est dans NP.

Exercice 5. Montrer que le langage $2\mathrm{COL} = \{\langle G\rangle \mid G$ graphe non orienté 2-coloriable$\}$ est dans P.

3 Propriétés de clôture

Exercice 6. Montrer que P est close par intersection, union, complémentation et étoile.

4 Quelques propriétés de classes

Étant donné une fonction $t\colon \mathbb{N} \to \mathbb{N}$, $\mathsf{DTIME}(t(n))$ est la classe des langages reconnus par une MT $\mathcal{M}$ telle qu'il existe une constante $\alpha \in \mathbb{R}$ pour laquelle, sur toute entrée x, $\mathcal{M}(x)$ fonctionne en temps au plus $\alpha \cdot t(|x|)$.

On peut alors définir $\mathsf{P} = \bigcup_{k\in\mathbb{N}} \mathsf{DTIME}(n^k)$ et $\mathsf{EXP} = \bigcup_{k\in\mathbb{N}} \mathsf{DTIME}(2^{n^k})$.

De même, étant donné une fonction $t\colon \mathbb{N} \to \mathbb{N}$, $\mathsf{NTIME}(t(n))$ est la classe des langages reconnus par une MTN $\mathcal{M}$ telle qu'il existe une constante $\alpha \in \mathbb{R}$ pour laquelle, sur toute entrée x, $\mathcal{M}(x)$ fonctionne en temps au plus $\alpha \cdot t(|x|)$, quels que soient les choix qu'elle fait lors de son exécution.

On définit alors, de manière similaire, $\mathsf{NP} = \bigcup_{k\in\mathbb{N}} \mathsf{NTIME}(n^k)$ et $\mathsf{NEXP} = \bigcup_{k\in\mathbb{N}} \mathsf{NTIME}(2^{n^k})$.

Rappelons en outre que, pour toute classe C de langages, la classe coC est donnée par $\mathsf{coC} = \{\bar{L} \mid L \in \mathsf{C}\}$.

Exercice 7. Montrer que si $\mathsf{P} = \mathsf{NP}$, alors $\mathsf{NP} = \mathsf{coNP}$.

Exercice 8. Montrer que si $\mathsf{P} = \mathsf{NP}$, alors $\mathsf{EXP} = \mathsf{NEXP}$.

Exercice 9. Montrer que si tout langage unaire de NP est dans P, alors $\mathsf{EXP} = \mathsf{NEXP}$.