TD 02 – Haine, Paix

Exercice 1. *C'est un peu court*

1. Montrer que 3-SAT est NP-complet.
2. Montrer que CLIQUE est NP-complet.
3. Montrer que O/1-IPROG est NP-complet.

Exercice 2. NP-*dur, par bdur*

1. Soit MTND-TEMPS le problème de décider, étant données une machine de Turing M et une borne de temps t inférieure au nombre d'états de M, si M s'arrête en moins de t étapes lorsque son ruban d'entrée est initialement vide. Montrer que MTND-TEMPS est NP-complet.

Définition. Soit T un ensemble de tuiles carrées (de côté 1) dont chaque bord possède une couleur prise dans un ensemble C. Un **pavage du carré** $n \times n$ est une fonction de $\{1, \dots, n-1\}^2$ dans T. Il est dit **valide** si les bords communs de deux tuiles adjacentes ont la même couleur, et que le bord du carré est unicolore. Il est dit **non trivial** s'il contient au moins deux tuiles différentes.

2. Soit PFP le problème de décider, étant donnés un ensemble fini T de tuiles à couleurs dans un ensemble fini C et un entier n, s'il existe un pavage valide et non trivial du carré $n \times n$. Montrer que PFP est NP-complet.

Exercice 3. SAT-*solver*

✎ Montrer que si $\mathsf{P} = \mathsf{NP}$, il existe un algorithme polynomial qui, étant donnée une formule φ sous forme normale conjonctive, retourne un assignement valide pour φ s'il en existe un et 0 sinon.

Exercice 4. *L'une erre, l'autre non*

1. Montrer que si tout langage unaire[1] de NP est dans P, alors $\mathsf{EXP} = \mathsf{NEXP}$.
2. Si $\mathsf{EXP} \neq \mathsf{NEXP}$, que peut-on dire de P et NP ?

Exercice 5. *Théorème de Berman (1978)*

1. Donner un algorithme récursif pour résoudre SAT.
2. Soit S un langage unaire NP-complet. En utilisant une réduction polynomiale de SAT à S, améliorer l'algorithme précédent pour qu'il devienne polynomial.
3. Donc ?

1. Un langage est dit unaire s'il est inclus dans $\{1\}^*$.