TD04 - Fonctions constructibles en temps

Exercice 1. *Définition d'une fonction constructible en temps*

Une fonction f de $\mathbb{N}$ dans $\mathbb{N}$ sera dite constructible en temps si il existe une machine de Turing M qui sur toute entrée x de longeur n s'arrête au bout de $f(n)$ étapes exactement.

1. Soient f_1 et f_2 deux fonctions de $\mathbb{N}$ dans $\mathbb{N}$ telles que f_2 et $f_1 + f_2$ sont constructibles en temps. Supposons en outre que pour presque tout n, $f_1(n) \geq \epsilon.f_2(n) + (1 + \epsilon)n$. Montrer qu'alors f_1 est constructible en temps.

2. Soit f une fonction de $\mathbb{N}$ dans $\mathbb{N}$ telle que $f(n) \geq n$ et telle qu'il existe un nombre réel positif ϵ pour lequel $f(n) \geq (1 + \epsilon)n$ presque partout.
Monter que les conditions suivantes sont équivalentes :
- f est constructible en temps.
- il existe une machine de Turing M qui sur toute entrée x de longueur n retourne $1^{f(n)}$ en temps $O(n)$.

Exercice 2. *Langages unaires*

Un langage est dit unaire si il est inclus dans 1^*.

1. Donner un algorithme récursif pour résoudre SAT.

2. Soit S un langage unaire NP-dur. En réduisant SAT à S, améliorer l'algorithme précédent pour qu'il devienne polynomial.

3. Qu'en concluez-vous ?

Exercice 3. *Langages creux*

Un langage L est dit creux lorsqu'il existe un polynôme p tel que, pour tout n, $L \cap \Sigma^n$ est de cardinal au plus $p(n)$.

1. Soit L un langage creux. Que pouvez-vous dire du cardinal de $L \cap \Sigma^{\leq n}$?

2. Nous allons montrer que si il existe un langage L creux et NP-dur, alors P = NP. Soit donc un tel langage L et soit X dans NP :

$$x \in X \text{ ssi } \exists w \in \Sigma^{p(|x|)}, \langle x, w\rangle \in A$$

avec p un polynôme et $A \in$ P. On veut montrer que X est décidable en temps polynomial.
Soit $G(A) = \{\langle x, w\rangle, \text{ tels que } \exists y \in \Sigma^{p(|x|)}, y \geq w \text{ et } \langle x, y\rangle \in A\}$.
Montrer que $G(A)$ est dans NP.

3. En utilisant une réduction de $G(A)$ à L, montrer que l'on peut décider X en temps polynomial (conseil : on pourra déterminer un algo polynomial qui, sur l'entrée x, trouve le plus grand w tel que $\langle x, w\rangle \in A$ lorsqu'il existe).