TD6 - Complexité
NL - Théorème de Ladner

Chargé de TD : Adrien Koutsos

Exercice 1 Montrer que les problèmes suivants sont **NL**-complets :

— **Input :** Un graphe dirigé $G = (V, E)$.
Question : G est-il fortement connexe ?

— **Input :** Un graphe dirigé $G = (V, E)$.
Question : G contient-il un cycle ?

Exercice 2 (PolyLog) On définie la classe de complexité :

$$\mathbf{PolyLog} = \bigcup_{k>0} \mathbf{SPACE}(log^k(n))$$

Montrer que $\mathbf{PolyLog} \neq \mathbf{P}$.

Exercice 3 (Théorème de Ladner) On souhaite montrer que si $\mathbf{P} \neq \mathbf{NP}$ alors il existe $L \in \mathbf{NP} \backslash \mathbf{P}$ tel que L ne soit pas **NP**-complet.

On considère un codage des machines de Turing dans les entiers tel que toute machine de Turing admette un nombre infinie de représentations : on notera M_i la machine de Turing codé par l'entier i. On définie récursivement la fonction $H : \mathbb{N} \mapsto \mathbb{N}$ et l'ensemble SAT_H comme suit :

$$SAT_H = \{\psi.0.1^{n^{H(n)}} \mid \psi \in SAT \wedge n = |\psi|\}$$

$$H(n) = \min \begin{cases} log(log(n)) \\ \min\limits_{i<log(log(n))} \{i \mid M_i \text{ decide } SAT_H \text{ en temps } i.x^i \text{ pour tout } |x| < log(log(n))\} \end{cases}$$

La définition ci-dessus n'est pas valide en 0 et en 1, on prendra donc $H(0) = H(1) = 0$. On va maintenant montrer les résultats suivants :

1. Montrez que H et SAT_H sont bien définies, que H est croissante et qu'elle se calcule en temps polynomial.
2. Montrez que si SAT_H est dans P alors H est bornée.
3. Réciproquement montrez que si H est bornée alors SAT_H est dans P.

On peux maintenant montrer les deux résultats suivants :

(i) $SAT_H \in \mathbf{NP}$.

(ii) $SAT_H \notin \mathbf{P}$.

On conclut la preuve du théorème en montrant que SAT_H n'est pas NP-complet.