TD 9

Langages Formels, Calculabilité et Complexité

Exercice 1: Existence de fonctions à sens unique

Supposons qu'il existe :

- une bijection f des entiers sur n bits vers les entiers sur n bits, pour tout n (*i.e.* sur une entrée x de n bits, $f(x)$ est un entier sur n bits tel que $f(x) \neq f(y)$ quand $x \neq y$).
- la fonction se calcule en temps polynomial (étant donné un entier x sur n bits, une machine de Turing retourne $f(x)$ en au plus $O(n^k)$ pas de calculs pour une constante k fixée.
- la fonction inverse de f, f^{-1} ne peut pas se calculer en temps polynomial. (On dit que c'est une fonction à sens unique.)

1. Montrer que si une telle bijection existe, alors $P \neq NP$. (*Idée: Montrer que le langage* $L = \{(x, y) : f^{-1}(x) < y\}$ *appartient à* $NP \setminus P$.)
2. Montrer de plus, que si elle existe, alors $NP \cap \mathrm{co} - NP \neq P$.

Exercice 2:

Montrer que si $P = NP$, alors tout langage de NP sauf les langages triviaux $\emptyset$ et A^* sont NP-complet. Pourquoi doit-on exclure ces langages triviaux?

Exercice 3:

Une formule booléenne est en *forme normale disjonctive* (DNF) si elle est une disjonction de clauses et chaque clause est une conjonction de littéraux. Le problème DNF-satisfiabilité DNFSat est le suivant :
INSTANCE: une formule booléenne $\phi \in DNF$.
QUESTION: Est-ce que ϕ est satisfiable ?

1. Montrer que DNFSat $\in P$.
2. Il est bien connu que, pour toute formule booléenne ϕ, on peut construire une formule booléenne ϕ dans DNF qui est équivalente à ϕ. Pourquoi ce fait avec (1) n'implique pas que SAT est dans P ?

   Pour tout entier naturel fixé k, on définit SAT_k de la manière suivante :
   INSTANCE: une formule booléenne $\phi \in CNF$ avec k clauses.
   QUESTION: Est-ce que ϕ est satisfiable ?
3. Montrer que $\mathrm{Sat}_k \in P$.
4. Considérer l'algorithme suivant pour 3-Sat: sur l'entrée ϕ, compter le nombre de clauses k de ϕ, utiliser ensuite l'algorithme en temps polynomial pour Sat_k qu'on a conçu à la question précédente. Pourquoi il n'implique pas que 3-Sat est dans P ?

5. Montrer que si vous avez un algorithme qui décide SAT, alors vous avez un algorithme qui trouve une solution.
6. Le problème du voyageur de commerce est le suivant: étant donné un graphe orienté avec un poids sur chaque arc et une valeur k, est-il possible de faire un tour (visiter chaque sommet au moins une fois) tel que le poids du voyage soit inférieur à k ? Comment trouver une solution à ce problème si vous savez le décider ?

Exercice 4: Coloriage

Le problème de COLORIAGE est le suivant :

INSTANCE: un graphe non-orienté $G = (V, E)$, et un entier positif k

QUESTION: Est-il possible de colorier les sommets de G avec au plus k couleurs différentes de telle manière que les sommets de toutes arêtes aient des couleurs différentes ?

Considérer la réduction suivantes des instances de SAT vers les instances de COLORIAGE. Soit ϕ une formule arbitraire en CNF, avec des variables $x_1, x_2, \ldots, x_n$ et les clauses $C_1, C_2, \ldots, C_r$. Étant donné ϕ, construire un graphe $G = (V, E)$ avec

$$\begin{aligned}
V &= \{v_0, \ldots, v_n\} \cup \{x_1, \ldots, x_n\} \cup \{\bar{x}_1, \ldots, \bar{x}_n\} \cup \{C_1, \ldots, C_r\};\\
E &= \{\{v_i, x_j\}, \{v_i, \bar{x}_j\} : 1 \leq i, j \leq n \text{ et } i \neq j\}\\
&\quad \cup \{\{v_i, v_j\} : 0 \leq i < j \leq n\}\\
&\quad \cup \{\{v_0, C_k\} : 1 \leq k \leq r\}\\
&\quad \cup \{\{x_i, \bar{x}_i\} : 1 \leq i \leq n\}\\
&\quad \cup \{\{x_i, C_k\} : x_i \text{ n'est pas un litteral de } C_k\}\\
&\quad \cup \{\{\bar{x}_i, C_k\} : \neg x_i \text{ n'est pas un litteral de } C_k\}
\end{aligned}$$

1. Dessiner le graphe G pour la formule
$$\phi = (\bar{x}_1 \vee x_2) \wedge (x_1 \vee x_3) \wedge (x_1 \vee \bar{x}_2 \vee \bar{x}_3)$$
2. Montrer que la réduction précédente s'exécute en temps polynomial de SAT vers COLORIAGE.
3. En déduire que COLORIAGE est NP-complet.

Exercice 5: Emploi du temps

Le problème suivant, EMPLOIDUTEMPS intervient lors de la préparation du concours d'entrée à l'École.

INSTANCE: Un ensemble P d'examens, un ensemble S de slots de temps, un ensemble $C = \{c_1, \ldots, c_k\}$ de candidats et pour chaque candidat c_i, un ensemble $P_i \subseteq P$ d'examens qu'il est supposé passer.

QUESTION: Est-ce qu'il existe une assignation des examens dans les slots de temps qui évite les collisions ? (Une collision apparaît quand un candidat doit passer deux examens en même temps.)

Montrer une réduction entre COLORIAGE et EMPLOIDUTEMPS pour montrer que ce dernier problème est NP-complet.