SI 4^{e}année **Année 2019-2020**
Complexité et Algorithmique

Non-déterminisme, réductions, problèmes NP-complets

Feuille de travaux dirigés n^{o}2

8 octobre 2018

1. On s'intéresse au problème suivant :

Nom : Somme de Sous-Ensembles
Instance : un ensemble fini E, une taille $s(e) \in \mathbb{N}$ pour chaque $e \in E$ et une capacité $C \in \mathbb{N}$.
Question : existe-t-il un sous-ensemble $E' \subseteq E$ tel que :

$$\sum_{e \in E'} s(e) = C$$

Montrer que le problème SOMME DE SOUS-ENSEMBLES est dans NP.

2. Montrer que le problème 3DM est dans NP.

3. On s'intéresse au problème du nombre composé :

Nom : nombre composé
Instance : N un entier en base k
Question : N est-il composé ?

1. Montrer que **nombre composé** $\in \mathsf{NP}$.
2. Que peut-on dire lorsque $k = 1$?
3. Est-il possible d'améliorer la complexité de la machine de Turing de la question précédente ?

Pour les deux premières questions, vous justifierez vos réponses en décrivant les machines de Turing correspondantes.

4. On rappelle les énoncés de quelques problèmes :

Nom : Chaîneham
Instance : Un graphe fini $G = (V, E)$ représenté sous forme de listes d'adjacence.
Question : Le graphe admet-il une chaîne Hamiltonienne (i.e. qui passe une et une seule fois par tous les sommets) ?

Nom : Cycleham
Instance : Un graphe fini $G = (V, E)$ représenté sous forme de listes d'adjacence.
Question : Le graphe admet-il un cycle Hamiltonien (i.e. qui passe une et une seule fois par tous les sommets) ?

Nom : Cheminham
Instance : Un graphe orienté fini $G = (V, E)$ représenté sous forme de listes d'adjacence.
Question : Le graphe admet-il un chemin Hamiltonienne (i.e. qui passe une et une seule fois par tous les sommets) ?

Nom : Circuitham
Instance : Un graphe orienté fini $G = (V, E)$ représenté sous forme de listes d'adjacence.
Question : Le graphe admet-il un circuit Hamiltonien (i.e. qui passe une et une seule fois par tous les sommets) ?

Construire les réductions suivantes :

1. **Cheminham** $\propto$ **Circuitham**
2. **Cycleham** $\propto$ **Circuitham**
3. **Chaîneham** $\propto$ **Cheminham**
4. **Cycleham** $\propto$ **Chaîneham**
5. **Circuitham** $\propto$ **Cheminham**
6. **Circuitham** $\propto$ **Cycleham**
7. **Cheminham** $\propto$ **Chaîneham**

5. Les problèmes de satisfiabilité. On s'intéresse aux problèmes suivants :

Nom : k-SAT
Instance : Une formule booléenne ϕ sous forme normale conjonctive composée de clauses de degré au plus k.
Question : ϕ est-elle satisfiable ?

Nom : Xk-SAT
Instance : Une formule booléenne ϕ sous forme normale conjonctive composée de clauses de degré exactement k.
Question : ϕ est-elle satisfiable ?

1. Montrer que 2-**SAT**$\propto$**X**2-**SAT**
2. Généralisation : montrer que k-**SAT**$\propto$**X**k-**SAT**
3. Montrer que k-**SAT**$\propto(k+1)$-**SAT**
4. Montrer que **X**$(k+1)$-**SAT** $\propto$ k-**SAT** si $k > 3$
   Indication : On peut transformer une clause $C_i = l_{i,1} \vee l_{i,2} \vee \ldots \vee l_{i,k+1}$ en la conjonction de deux clauses $C'_i = l_{i,1} \vee l_{i,2} \vee \ldots \vee l_{i,t} \vee y_i$ et $C''_i = l_{i,t+1} \vee l_{i,t+2} \vee \ldots \vee l_{i,k+1} \vee \neg y_i$, où y_i est une nouvelle variable qui ne sert nullepart ailleurs. Expliquez comment cette technique de *glue* peut servir. Pour quelles valeurs de k peut-on utiliser cette technique ?
5. En utilisant la méthode de *glue* on peut aussi réduire une formule de **SAT** à une donnée de 3-**SAT**, conclure la NP-complétude de 3-**SAT** et **X**-3-**SAT**, ainsi que celles de k-**SAT** et **X**-k-**SAT** pour $k \geq 3$.
6. Montrer que **X**2-**SAT**$\in$ P.

Indication : pour toute clause, on construira un graphe dont les sommets sont les variables et la négation des variables et tel que pour chaque clause $l_i \vee l_j$ on a une implication $\neg l_i \to l_j$ et $\neg l_j \to l_i$.
On montrera que ϕ est une antilogie [1] si et seulement si il existe un circuit dans le graphe qui contient à la fois x_i et sa négation.

1. A l'opposée d'une *tautologie*, une formule vraie pour toute valeur de vérité de ses variables, une *antilogie* est une formule qui est fausse quelle que soit la valeur de vérité de ses variables, donc une formule dont la négation est une tautologie.