ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Sections d'Informatique et de Systèmes de Communication

Série d'exercices 13 19 Dec. 2011

1. La taille de l'input

Pour chacun des problèmes suivants, donner la taille de l'input en fonction de n :

a) Etant donné un nombre $n \in \mathbb{N}$, trouver sa décomposition en produit de nombres premiers.

b) Etant donné une matrice binaire $n \times n$, calculer l'inverse de cette matrice, s'il existe.

c) Etant donné n entiers non-négatifs plus petits que 1000, trouver le plus grand.

d) Etant donné un graphe G avec des poids entre -100 et 100 (donné par sa matrice d'adjacence) trouver un arbre couvrant minimal.

2. Réduction polynomiale

Rappelons d'abord qu'une *clique* dans un graphe $G(V, E)$ est un sous-ensemble $S \subseteq V$ tel que $S \times S \subseteq E$, c'est à dire un ensemble de sommets qui sont tous connectés entre eux.

Considérons les problèmes de décision suivants :

Problème : CLIQUE
Input : Un graphe non orienté G, et un entier k.
Output : Vrai si G contient une clique de taille $\geq k$, Faux sinon.

Problème : HALF-CLIQUE
Input : Un graphe non orienté G, ayant n sommets (où n est pair).
Output : Vrai si G contient une clique de taille $\geq n/2$, Faux sinon.

Nous voulons montrer que

$$\text{CLIQUE} \leq_P \text{HALF-CLIQUE}.$$

Ainsi, il s'agit donc de construire une fonction f qui transforme un input (G, k) de CLIQUE en un input G' de HALF-CLIQUE.

a) Soit G un graphe, et soit $f_1(G)$ le graphe obtenu en ajoutant à G un sommet connecté à tous les autres sommets. Montrer que G contient une clique de taille $\geq k$ si et seulement si $f_1(G)$ contient une clique de taille $\geq k + 1$.

b) Généraliser la construction ci-dessus. Plus précisément, pour tout $a \in \mathbb{N}$, montrer qu'étant donné un graphe G, on peut construire un graphe $f_a(G)$ tel que G contient une clique de taille $\geq k$ si et seulement si $f_a(G)$ contient une clique de taille $\geq k + a$.

c) Soit m le nombre de sommets dans G. Supposons que $k \leq m/2$. Trouver a en fonction de m et k tel que G contient une clique de taille $\geq k$ si et seulement si $f_a(G)$ contient une clique de taille $\geq n_a/2$, où n_a dénote le nombre de sommets dans $f_a(G)$.

d) Déduire que CLIQUE $\leq_P$ HALF-CLIQUE. (*Indice :* Distinguer les deux cas $k \leq m/2$ et $k > m/2$).

3. SAT

On rappelle qu'une formule CNF (*Conjunctive normal form*) sur les variables $x_1, \ldots, x_n$ s'écrit

$$F = C_1 \wedge \ldots \wedge C_m,$$

où chaque clause C_j est de la forme $\lambda_1 \vee \ldots \vee \lambda_{k_j}$ (et on a pour chaque i : $\lambda_i = x_i$ ou $\overline{x_i}$). Les λ_i sont appelés des *littéraux*. Une *attribution* aux variables $x_1, \ldots, x_n$ consiste à donner la valeur TRUE ou FALSE à chacune des variables x_i.

Pour une attribution donnée, la formule F sera donc soit vraie (l'attribution est alors dite *satisfaisante*), soit fausse. Une formule F est dite *satisfaisable* s'il existe une attribution aux variables $x_1, \ldots, x_n$ telle que F est vraie.

a) Considérons la formule suivante sur les variables x_1, x_2, x_3 :

$$F_1 = (x_1 \vee \overline{x_2}) \wedge (x_2 \vee x_3 \vee \overline{x_1}) \wedge (\overline{x_3} \vee \overline{x_1})$$

L'attribution $x_1 =$ TRUE, $x_2 =$ FALSE, $x_3 =$ TRUE est elle satisfaisante ?

b) Trouver une attribution satisfaisante pour F_1.

c) La formule suivante, sur les variables x_1 et x_2 est elle satisfaisable ?

$$F_2 = (x_1 \vee x_2) \wedge (x_1 \vee \overline{x_2}) \wedge (\overline{x_1} \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2})$$

d) On considère le problème suivant : étant donné une formule CNF F sur n variables, avec m clauses contenant $k_1, \ldots, k_m$ littéraux, determiner si cette formule est satisfaisable (C'est le problème SAT). Quelle est la taille de l'input pour ce problème ?

e) Montrer que ce problème est dans NP.

f) Montrer que si P=NP, alors il existe un algorithme polynomial qui, étant donné une formule CNF F, retourne une attribution satisfaisante aux variables $x_1, \ldots, x_n$, si une telle attribution existe (donc il ne retourne pas juste TRUE ou FALSE, mais il retourne l'attribution elle même si elle existe).