

Outils et algorithmes pour la protection de l'information (I231)
Session 2 (2020-2021) - Partie 1 (13pts)

La précision et la clarté de votre rédaction sont *fondamentales*.
 Chaque réponse *doit* être justifiée, les programmes *doivent* être commentés et les algorithmes *expliqués*. Le barème est indicatif.
 Les documents sont interdits. Durée : 2h00

RAPPELS ET NOTATIONS

On appelle *fonction caractéristique* d'une partie A d'un ensemble X l'application $\mathbf{1}_A : X \rightarrow \{0, 1\}$ définie par

$$\mathbf{1}_A(x) := \begin{cases} 1 & \text{si } x \in A. \\ 0 & \text{si } x \notin A. \end{cases} \quad (1)$$

et le *vecteur caractéristique* de A le n -uplet binaire

$$\chi(A) := (\mathbf{1}_A(x_1), \mathbf{1}_A(x_2), \dots, \mathbf{1}_A(x_n)). \quad (2)$$

On rappelle que les *opérateurs bit-à-bit* sont l'extension des opérateurs de la logique booléenne aux n -uplets binaires en agissant sur chacun des n bits des opérandes. Quand ces opérateurs sont définis sur des entiers naturels, ils agissent sur les chiffres de leurs écritures binaires. On dispose de l'opérateur unaire de *négation* \neg et des opérateurs binaires de *disjonction* $+$, de *conjonction* \cdot , de *disjonction exclusive* \oplus .

On rappelle qu'une *partition* d'un ensemble X est une famille de parties non-vides et deux-à-deux disjointes de X dont la réunion est égale à X .

EXAMEN

EXERCICE 1. [4pts] On se fixe l'alphabet $\Sigma := \{0, 1\}$.

- (1) Démontrez que l'opérateur bit-à-bit \oplus vu comme une fonction de $\mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ est Turing-calculable.
- (2) Calculez les fonctions de complexité en temps $T(n)$ et en espace $E(n)$ de votre algorithme.

NB. Les opérandes en entrée sont de même taille n (en complétant le plus court par des 0 à gauche si nécessaire) et sont séparés par une case vide. Expliquez votre algorithme en français, ou en langage pseudo-algorithmique

au préalable et commentez impérativement les blocs d'instructions de votre machine de Turing.

EXERCICE 2. [9pts] On considère les problèmes de décision suivants :

Problème XC [COUVERTURE EXACTE]

Instance : Un ensemble fini X et $S \subseteq \mathcal{P}(X)$.

Question : S contient-il une partition P de X ?

Problème SSE [SOMME DE SOUS-ENSEMBLES]

Instance : Un ensemble fini E , une pondération $w : E \rightarrow \mathbb{N}^*$ et une capacité $W \in \mathbb{N}$.

Question : Existe-il une partie $A \subseteq E$ telle que $\sum_{x \in A} w(x) = W$?

On pose $X = \{1, 2, \dots, n\}$ et $S = \{S_1, S_2, \dots, S_m\}$.

- (1) Trouvez une instance positive (non-triviale) pour chacun des problèmes.
- (2) Démontrez que le problème SSE est dans la classe NP.

On transforme une instance (X, S) du problème XC en instance (E, w, W) du problème SSE : on pose $b := n + 1$ et $E := \{1, 2, \dots, m\}$, et on définit la pondération $w : E \rightarrow \mathbb{N}^*$ et la capacité W par

$$w(j) := \sum_{i=1}^n \mathbf{1}_{S_j}(i) b^{i-1}, \quad W := \frac{b^n - 1}{b - 1}.$$

- (3) Démontrez que si $P \subseteq S$ est une partition de X , alors il existe une partie A de E de poids total W .
- (4) Démontrez que s'il existe une partie A de E de poids total W alors il existe une partition $P \subseteq S$ de X .
- (5) Démontrez que cette transformation est polynomiale. Que faut-il supposer sur le problème XC pour en conclure que SSE est NP-complet ?

Indication : pour vous aider à comprendre la transformation, dressez à partir de l'instance positive de XC que vous avez fournie en (1), la table à m lignes et 4 colonnes dont la j -ème ligne est :

j	S_j	$\chi(S_j)$	$w(j)$
-----	-------	-------------	--------

et en mettant en évidence les bits non-nuls des vecteurs caractéristiques $\chi(S_j)$ sur les lignes j associées à la partition P de X .