

TD N°1 - INTRODUCTION À LA COMPLEXITÉ

Exercice 1

- Montrez en détaillant toutes les étapes que $8n + \sqrt{n} + 16 \in \Theta(n)$

Exercice 2

- Pour chacune des formules suivantes

$$42n^2 + 100n + 2000$$

$$1000 + n^3$$

$$\ln(n) + 3n$$

$$200n + n \ln(n)$$

Montrez auxquels de ces ensembles elle appartient:

$$O(n^3) \quad \Theta(n^3)$$

$$O(n) \quad \Theta(n)$$

$$O(n \log n) \quad \Theta(n \log n)$$

$$O(n^2) \quad \Theta(n^2)$$

Exercice 3

- Etant donné deux fonctions f et g sur $\mathbb{N}$, montrez que:

$$\max\{f, g\} \in \Theta(f + g)$$

$$f + g \in \Theta(\max\{f, g\})$$

$$\text{si } f \in \Theta(g), \text{ alors } g \in \Theta(f)$$

$$\text{si } f \in O(g) \text{ et } g \in O(h), \text{ alors } f \in O(h)$$

Exercice 4

- Soit l'algorithme suivant:

Algorithm. *Dummy*(tableau de n bits T)

$s=0$

Pour $i=1$ à n faire

Si $T[i]=1$ alors

Pour $j=1$ à i faire

$s=s+1$

Fin Pour

Fin Pour

Si s est pair, renvoyer YES sinon NO

Pour une instance de taille n , quelle est la complexité en temps de cet algorithme dans le meilleur des cas, et dans le pire des cas ? Quelle est sa complexité en espace mémoire ?

Exercice 5

. Les propositions suivantes sont-elles vraies ou fausses ?

- (1) Si je montre qu'un algorithme a une complexité en temps de $O(n^2)$, est-il possible que sur certaines instances, sa complexité ne soit que de $O(n)$?
- (2) Si je montre qu'un algorithme a une complexité en temps de $O(n^2)$, est-il possible que sur *toutes* ses instances, sa complexité ne soit que de $O(n)$?
- (3) Si je montre qu'un algorithme a une complexité en temps de $\Theta(n^2)$, est-il possible que sur certaines instances, sa complexité ne soit que de $O(n)$?
- (4) Si je montre qu'un algorithme a une complexité en temps de $\Theta(n^2)$, est-il possible que sur *toutes* ses instances, sa complexité ne soit que de $O(n)$?
- (5) Tout problème appartenant à $DTIME(n^3)$ appartient aussi à $DTIME(2^n)$
- (6) Tout problème appartenant à $DTIME(n)$ appartient aussi à $DSPACE(n^2)$
- (7) Il y a des problèmes dont la complexité en temps est inférieure à $\Theta(2^n)$, mais qui ne sont pas dans $PTIME$?

Exercice 6

. Soit le problème de décision suivant: Etant donné un ensemble de n entiers $E \subset \{1, \dots, m\}$, y-a-t-il parmi cet ensemble un nombre égal à la somme de tous les autres ? (Formellement: $\exists x \in E, x = \sum_{y \in E \setminus \{x\}} y$).

- (1) Donner une instance YES, et une instance NO de ce problème.
- (2) Proposez deux codages pour ce problème. Quelle est la taille d'une instance (en bits), en fonction de m et/ou en fonction de $|E|$? Quel est le meilleur codage ?
- (3) Proposez (sans les détails) un algorithme et montrez-en la complexité (pour l'un des codages au choix)
- (4) Ce problème (associé à ces codages) est-il dans $PTIME$?

Exercice 7

. Considérons le problème de décision suivant: "un graphe est-il planaire

?”. Soit n , le nombre de sommets du graphe. Supposons qu’on ait un algorithme en $O(n^k)$ qui résolve ce problème.

- (1) Le problème est-il dans PTIME ?
indice: cela dépend du codage.
- (2) Peut-on en déduire une conclusion générale pour l’ensemble des problèmes de décisions pour lesquels on connaît un algorithme en $O(m^k)$, m étant la “taille de l’objet traité” ?

Exercice 8

. Considérez le programme suivant:

Algorithm. *PRIM(entier m)*

Pour $i = 1 \dots m - 1$ faire

Si m modulo $i = 0$ Alors Renvoyer YES

Fin Pour

Renvoyer NO

- (1) Quel problème de décision résout cet algorithme ?
- (2) Quelle est la complexité de cet algorithme, en fonction de la valeur de m ?
- (3) Quelle est la taille d’une instance ?
- (4) Quelle est la complexité de cet algorithme, en fonction de la taille de l’instance ?
- (5) Que peut-on conclure quant au problème de décision ? Appartient-il à PTIME ou EXPTIME ?

Exercice 9

. Soit le problème de décision suivant: Etant donné un ensemble d’entiers relatifs $E \subset \mathbb{Z}$, y-a-t-il parmi cet ensemble un sous-ensemble non-vide dont la somme est nulle ? (Formellement: $\exists X \subseteq E, X \neq \emptyset, \sum_{x \in X} x = 0$).

- (1) Donner une instance YES, et une instance NO de ce problème.
- (2) Proposez (sans les détails) un algorithme et montrez-en la complexité
- (3) Ce problème est-il dans PTIME ? est-il dans PSPACE ?