

Rendre les copies avant le 12 mai 2004

## Le monde des rectangles

Dans cette série d'exercices, il est question d'ensembles de rectangles avec leurs côtés parallèles aux axes  $Ox$  et  $Oy$  (voir la figure 1). Dans une version simplifiée, on considère des ensembles de segments horizontaux sur l'axe  $Ox$ ; on peut aussi considérer une généralisation à plusieurs dimensions, en considérant des ensembles de parallélépipèdes.

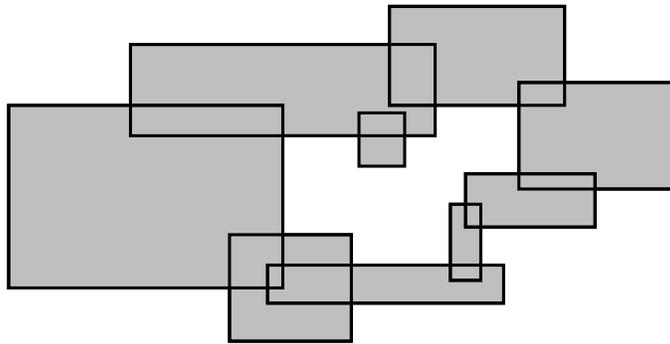


FIG. 1 – Un ensemble de rectangles sur le plan.

**Exercice 1.** Il y a  $n$  segments  $[a_i, b_i]$ ,  $i = 1, \dots, n$ , sur l'axe horizontal; les extrémités de certains segments peuvent coïncider. Proposer un algorithme pour calculer la mesure de leur réunion  $\bigcup_{i=1}^n [a_i, b_i]$ . La complexité doit être  $O(n \log n)$ .

L'algorithme ci-dessus est *statique* : l'ensemble des segments est connu dès le début. Pour construire des algorithmes *en ligne*, capables d'ajouter ou d'enlever des segments selon des requêtes et en ordre quelconque, on utilise une structure de données qui s'appelle *arbre de segments*. Bien que les segments peuvent nous parvenir au fur et à mesure, on suppose tout de même que l'ensemble des valeurs possibles de leurs extrémités est connu dès le début.

Supposons que  $N = 2^k$ , et les extrémités des segments traités appartiennent à une suite  $x_0, x_1, \dots, x_N$  triée en ordre croissant. Un exemple d'un arbre de segments, pour  $N = 16$ , est donné sur la figure 2. Chaque nœud, à part deux indices (ou deux coordonnées si cela est plus pratique), contient aussi une information supplémentaire qui dépend du problème traité. Le segment  $[x_5, x_{12}]$ , par exemple, est alors représenté comme union des segments  $[x_5, x_6]$ ,  $[x_6, x_8]$  et  $[x_8, x_{12}]$ . Il faut bien noter le fait que, une fois le segment  $[x_8, x_{12}]$  utilisé, on n'utilise plus ses plus petits sous-segments. En termes généraux, un segment  $[x_i, x_j]$  se représente comme union des segments qui se caractérisent par les trois propriétés suivantes : (a) ils font partie de  $[x_i, x_j]$  ; (b) ils correspondent aux nœuds de l'arbre ; (c) ils sont les plus grands possibles satisfaisant (a) et (b).

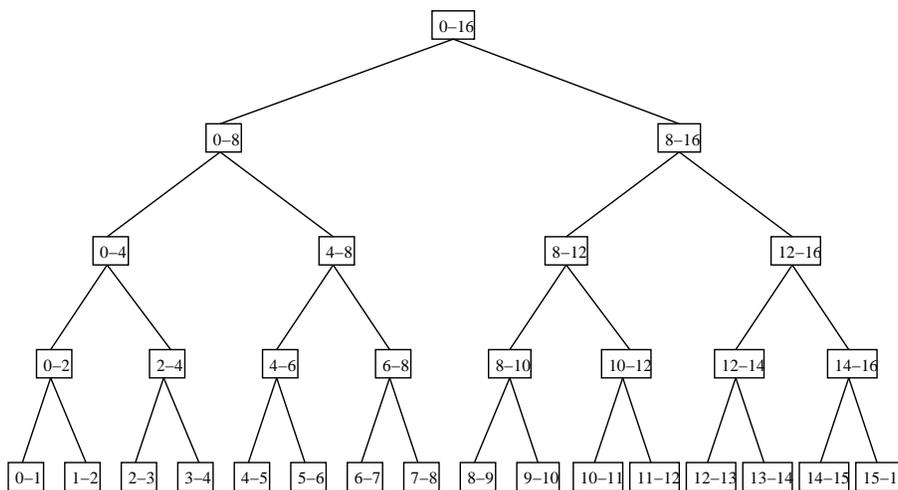


FIG. 2 – L'arbre de segments pour traiter les segments dont les extrémités appartiennent à un ensemble donné de points  $x_0, x_1, \dots, x_{16}$ . Chaque nœud, à part deux indices (ou deux coordonnées), contient aussi une information supplémentaire qui dépend du problème traité.

**Exercice 2.** Un ensemble de segments dont les extrémités appartiennent à l'ensemble donné  $x_0, \dots, x_N$ , est représenté sous forme d'un arbre de segments. Proposer un algorithme qui, pour un  $x$  donné, répond à la question combien de segments de l'ensemble couvrent ce point. Complexité  $O(\log N)$ .

Préciser quelle information supplémentaire, à part les deux coordonnées, faut-il stocker dans les nœuds de l'arbre.

**Exercice 3.** Un ensemble de segments change dynamiquement : les segments  $y$  sont ajoutés ou d'y sont enlevés au fur et à mesure. Proposer un algorithme de complexité  $O(\log N)$  pour *recalculer* la mesure de la réunion des segments après chaque opération d'insertion ou de suppression.

Remarque. Dans cet exercice, on demande aussi une *preuve* que la complexité de l'algorithme en question est vraiment  $O(\log N)$ . Cette preuve n'est pas évidente car l'insertion ou la suppression d'un segment ne se réduisent pas à un parcours d'une seule branche de l'arbre.

**Exercice 4.** Les données du problème sont  $n$  rectangles sur le plan dont les côtés sont parallèles aux axes (voir de nouveau la figure 1). L'objectif est de trouver l'aire de leur réunion (c'est-à-dire, l'aire de la région hachurée sur la figure). Proposer un algorithme de complexité  $O(n \log n)$  pour résoudre ce problème.

**Indications.** Utiliser la méthode de balayage par une ligne verticale se déplaçant de gauche à droite. Selon l'axe horizontal, utiliser un analogue de l'algorithme de l'exercice 1 et, pour chaque position de la ligne de balayage utiliser, selon l'axe vertical, l'algorithme de l'exercice 3.

**Exercice 5.** Proposer un algorithme de complexité  $O(n^2 \log n)$  pour calculer le volume d'une réunion de  $n$  parallélépipèdes donnés de dimension 3, dont les côtés sont parallèles aux axes  $Ox$ ,  $Oy$  et  $Oz$ .

Remarque. Pour ce problème-là, il existe aussi un algorithme plus performant, de complexité  $O(n^2)$ . On ne vous demande pas de chercher celui-ci.