Épreuve d'informatique de l'X - 2005 - MP/PC

(Durée : 2 heures)

Lignes d'horizon

On attachera une grande importance à la concision, à la clarté, et à la précision de la rédaction. On précisera en tête de copie le langage de programmation utilisé.

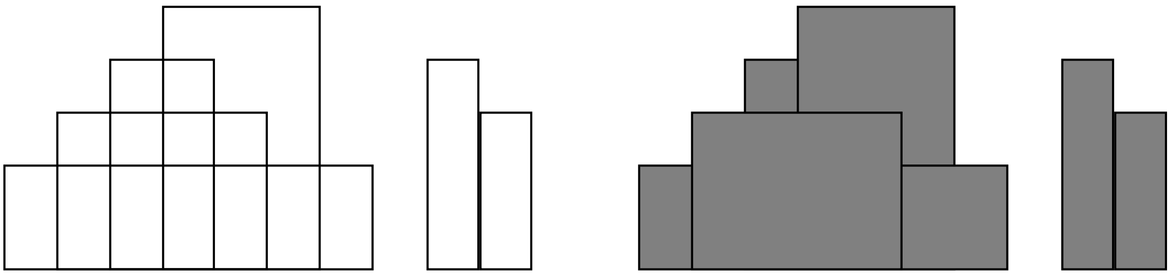
Le temps d'exécution $T(f)$ d'une fonction f est le nombre d'opérations élémentaires (addition, soustraction, multiplication, division, affectation, etc.) nécessaire au calcul de f. Lorsque ce temps d'exécution dépend d'un paramètre n, il sera noté $T_n(f)$. On dit que la fonction f s'exécute :

— en temps linéaire en n, s'il existe $K > 0$ tel que pour tout n, $T_n(f) \leqslant Kn$;

— en temps quadratique en n, s'il existe $K > 0$ tel que pour tout n, $T_n(f) \leqslant Kn^2$.

I. Détermination de la ligne d'horizon

On cherche à calculer la ligne d'horizon engendrée par n beaux bâtiments modernes ($n > 0$), assimilés à des parallélépipèdes verticaux. Pour simplifier, on se place dans l'espace à deux dimensions ; nos bâtiments sont de simples rectangles verticaux ; la ligne d'horizon est représentée par une suite de segments horizontaux, comme indiqué sur la figure suivante :



Dans trois tableaux globaux à valeurs entières g, d et h de longueur n, on range respectivement l'abscisse de gauche, l'abscisse de droite et la hauteur du bâtiment i vérifiant $0 < h[i] < M$ et $0 \leqslant g[i] < d[i] < N$ pour tout i ($0 \leqslant i < n$) où M et N sont deux constantes globales (comme d'habitude, les abscisses croissent de la gauche vers la droite, et les ordonnées du bas vers le haut).

Dans l'exemple ci-dessus, les tableaux g, d et h valent respectivement $\langle 2,3,0,1,8,9\rangle$, $\langle 4,6,7,5,9,10\rangle$ et $\langle 4,5,2,3,4,3\rangle$.

On définit une matrice $E = (e_{i,j})$ ($0 \leqslant i < M$, $0 \leqslant j < N$) de pixels (*picture elements*) dans laquelle on dessine les bâtiments ; chaque élément $e_{i,j}$ vaut 0 ou 1 ; plus exactement $e_{i,j} = 1$ si et seulement si le point de coordonnées réelles $(j + 0,5,\ i + 0,5)$ est à l'intérieur d'un bâtiment.

Question 1. Écrire une fonction `remplir()` qui initialise la matrice E.

La ligne d'horizon est représentée par un tableau *hor* de longueur ℓ contenant une succession d'abscisses et de hauteurs : $hor[2k]$ et $hor[2k+2]$ déterminent les abscisses de début et de fin d'un bout de ligne d'horizon à hauteur $hor[2k+1]$ ($0 \leqslant k < \ell/2 - 1$). Ainsi, dans l'exemple ci-dessus, la ligne d'horizon peut être représentée par le tableau $\langle 0,2,1,3,2,4,3,5,6,2,7,0,8,4,9,3,10,0,N\rangle$ avec $N = 11$. La « sentinelle » N pourra ne pas être le dernier élément du tableau *hor* ; on autorisera donc les lignes d'horizon contenues dans les seuls premiers éléments d'un tableau.

Dans un premier temps (questions 2 et 3), on se contente d'une ligne d'horizon où tous les segments horizontaux sont de longueur 1. Avec cette convention, on prend, pour représenter la ligne d'horizon correspondant à la figure donnée en exemple, $\langle 0,2,1,3,2,4,3,5,4,5,5,5,6,2,7,0,8,4,9,3,10,0,N\rangle$. Alors la longueur ℓ du tableau vérifie $\ell = 2N + 1$ et, pour $0 \leqslant k \leqslant N$, $hor[2k] = k$.

Question 2. Écrire une fonction `horizon1()` qui calcule, à partir de la matrice E, la ligne d'horizon en temps linéaire par rapport au produit MN.

On peut réduire le temps d'exécution de cette fonction en écrivant une fonction qui longe la ligne d'horizon dans la matrice E. Le contour de l'horizon est la ligne continue formée d'une succession de lignes horizontales et de verticales qui borde l'ensemble supérieur des bâtiments. La longueur de ce contour est la somme des longueurs des segments verticaux et horizontaux qui le composent.

Question 3. Écrire une fonction `horizon2()` qui calcule, à partir de la matrice E, la ligne d'horizon en temps linéaire par rapport à la longueur L du contour de l'horizon.

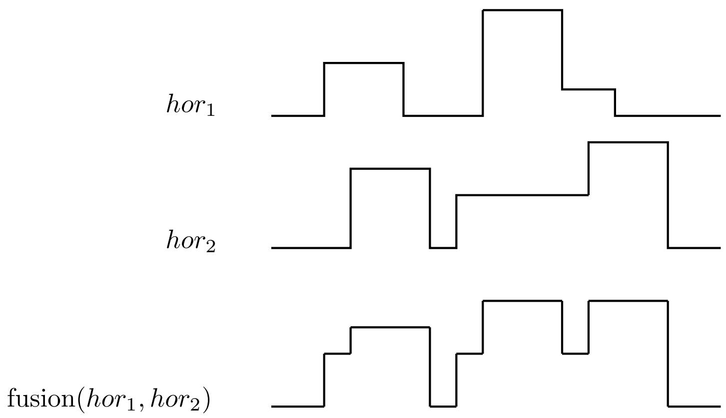
II. Transformations de lignes d'horizon

Un tableau `hor`, de longueur ℓ, représentant une ligne d'horizon, est en forme canonique si, pour tout k, elle vérifie

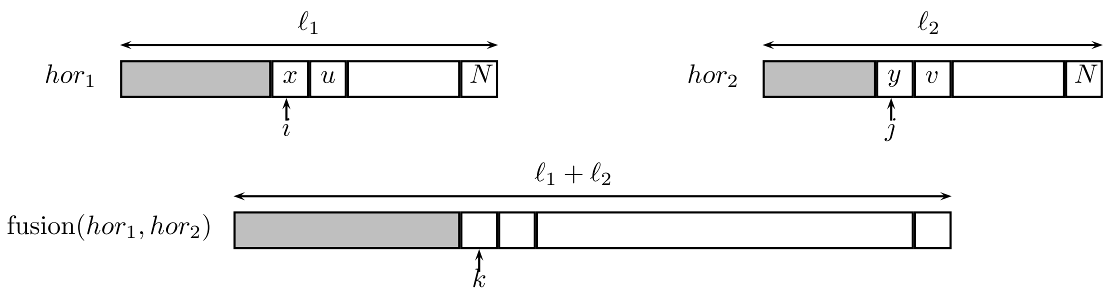
$$hor[2k] < hor[2k+2] \ (0 \leqslant k < (\ell-1)/2) \quad \text{et} \quad hor[2k+1] \neq hor[2k+3] \ (0 \leqslant k < (\ell-3)/2).$$

Question 4. Écrire une fonction `canonique(`hor`)` qui retourne un tableau représentant hor en forme canonique, en temps linéaire par rapport à la longueur ℓ de hor.

On veut à présent fusionner deux lignes d'horizon comme indiqué sur la figure suivante :



Les deux lignes d'horizon sont représentées par les tableaux hor_1 et hor_2 de longueurs ℓ_1 et ℓ_2. On peut effectuer cette fusion très facilement en examinant les deux tableaux de la gauche vers la droite, en maintenant la hauteur u' du dernier bâtiment rencontré dans hor_1 et la hauteur v' du dernier bâtiment rencontré dans hor_2, comme indiqué sur la figure suivante :



Question 5. Écrire la fonction `fusion (`hor_1`, `hor_2`)` qui retourne, en temps linéaire par rapport à $\ell 1 + \ell 2$, une ligne d'horizon fusionnant les lignes d'horizon hor_1 et hor_2 de longueur ℓ_1 et ℓ_2.