Epreuve d'algorithmique — mathématiques appliquées

(Durée : 3 heures)

Il est recommandé de lire l'ensemble du sujet avant de commencer la rédaction. Les parties sont largement indépendantes, et de difficulté croissante. Il est donc souhaitable de répartir son temps équitablement entre elles.

Les questions concernant l'implémentation des algorithmes en Pascal ne seront prises en considération que si la description algorithmique de la méthode est correcte et suffisamment complète.

Introduction : pavage de figures planes

Il y a quelques années, John H. Conway, mathématicien anglais travaillant à l'Université de Princeton aux Etats-Unis, a étudié l'existence de pavages de figures du plan par des tuiles polygonales.

C'est par exemple un casse-tête mathématique classique que de montrer qu'un échiquier dans lequel on a enlevé 2 coins opposés diagonalement ne peut être pavé par des dominos (c'est une conséquence directe de la partie 3).

Les travaux de Conway ont été étendus et diffusés par William P. Thurston, médaille Field (1982). Ils ont été repris récemment en France dans un cadre algorithmique.

Le but de ce problème est de démontrer quelques résultats simples sur l'existence et la construction algorithmique de pavages des figures trapézoïdales par des dominos.

Notations

Dans tout ce problème, R désigne l'ensemble des nombres réels, et Z celui des entiers relatifs.

On considère le plan R^2 comme un ensemble de cases de côté 1 centrées sur les points de Z^2. Chaque case $[i-\frac{1}{2}, i+\frac{1}{2}] \times [j-\frac{1}{2}, j+\frac{1}{2}]$ du quadrillage est repérée par les coordonnées (i,j) de son *centre* dans Z^2. Les *bords* de la case sont les 4 segments

$$[(i-\tfrac{1}{2}, j-\tfrac{1}{2}), (i-\tfrac{1}{2}, j+\tfrac{1}{2})], \quad [(i-\tfrac{1}{2}, j-\tfrac{1}{2}), (i+\tfrac{1}{2}, j-\tfrac{1}{2})],$$
$$[(i+\tfrac{1}{2}, j-\tfrac{1}{2}), (i+\tfrac{1}{2}, j+\tfrac{1}{2})], \quad [(i-\tfrac{1}{2}, j+\tfrac{1}{2}), (i+\tfrac{1}{2}, j+\tfrac{1}{2})].$$

Un segment est donc adjacent à exactement *deux* cases.

Une *figure* F du plan est un ensemble *fini* de cases du quadrillage. La *région* R du plan R^2 correspondant à une figure F est l'union des carrés $[i-\frac{1}{2}, i+\frac{1}{2}] \times [j-\frac{1}{2}, j+\frac{1}{2}]$ tels que la case (i,j) soit dans F. La ligne L_j de la figure F est $\{i \mid (i,j) \in F\}$.

Hypothèse simplificatrice : Dans toute la suite, on ne considèrera que des figures dont toutes les cases ont des coordonnées stritcement positives.

Partie 1 : Convexité d'une figure

On considère une figure décrite par une matrice booléenne notée fig.

```
const N= 1024;      (* Par exemple... *)
type figure= array [1..N,1..N] of boolean;
var fig: figure;
```

telle que $fig[i,j] = true$ si la case (i,j) est dans la figure et $fig[i,j] = false$ sinon.

Question 1.1 : *Ecrire une procédure*

procedure figure-vide(fig: figure; var vide: boolean);

qui teste si la figure fig est vide.

On dit qu'une figure F est *convexe par lignes* si, pour tout j, la ligne L_j est soit vide, soit un intervalle $[min_j, max_j]$.

Question 1.2 : *Ecrire une procédure*

procedure figure-ligne-convexe(fig: figure; var ligne-convexe: boolean);

qui teste si la figure fig est convexe par lignes.

De façon similaire, on définit la notion de figure *convexe par colonnes*. On dit qu'une figure est *convexe* si elle est convexe par lignes et convexe par colonnes.

Question 1.3 : *Ecrire une procédure* procedure figure-convexe(fig: figure; var convexe: boolean);

qui teste si la figure fig est convexe.

On dit qu'une figure est un *trapèze* si elle est convexe et si de plus, pour tout i, j allant de 1 à N, (i, j) appartient à la figure implique que $(i, 1)$ appartient aussi à la figure.

Question 1.4 : *Montrer que F est un trapèze si et seulement si elle est convexe par lignes et $L_1 \supseteq L_2 \supseteq \ldots \supseteq L_N$. Soit $j_0 \geq 0$ maximal tel que $L_{j_0} \neq \emptyset$. En déduire que $(min_j)_{1 \leq j \leq j_0}$ est une suite croissante et que $(max_j)_{1 \leq j \leq j_0}$ est une suite décroissante.*

Question 1.5 : *Ecrire une procédure*

procedure figure-trapeze(fig: figure; var trapeze: boolean);

qui, avec un seul parcours du tableau fig, teste si la figure fig est un trapèze.

Partie 2 : Mot de contour d'une figure

On dit qu'un ensemble de cases est *connexe* si, pour toute paire de cases (i, j) et (k, ℓ) de cet ensemble, la propriété suivante est vraie. Il existe une suite de cases de l'ensemble, $(i_0 = i, j_0 = j)$, (i_1, j_1), ..., $(i_m = k, j_m = \ell)$ telle que :

$$\forall r \in \{1, \ldots, m\},\ (i_r = i_{r-1} \text{ et } |j_r - j_{r-1}| = 1) \text{ ou } (|i_r - i_{r-1}| = 1 \text{ et } j_r = j_{r-1}).$$

Question 2.1 : *Montrer qu'il existe des figures du plan telles que la région associée est connexe au sens topologique usuel, mais qui ne sont pas connexes au sens de la définition ci-dessus.*

On dit qu'une figure connexe F est *sans trou* si son complémentaire est un ensemble de cases qui est connexe au sens défini ci-dessus.

Soit F une figure. La *frontière* de F est l'ensemble des segments tels qu'exactement l'une des deux cases adjacentes à ce segment appartienne à F.

On définit le *mot de contour* d'une figure F finie connexe sans trou de la façon suivante : soit p le point de la région R correspondant à F d'abscisse minimale, et d'ordonnée minimale parmi les points de R d'abscisse minimale. On parcourt la suite des segments qui forment la frontière de F dans le sens direct à partir de p, c'est-à-dire en gardant toujours à sa gauche la case adjacente au segment parcouru appartenant à F, jusqu'à retrouver le point p.

Remarque importante. *On admettra que l'ensemble des segments de la frontière est ainsi parcouru sans répétition.*

Cette frontière est formée d'une suite de segments horizontaux et verticaux de longueur 1. À chaque segment horizontal, on associe la lettre d ou g selon qu'il est parcouru dans le sens des abscisses croissantes ou décroissantes. À chaque segment vertical, on associe la lettre h ou b selon qu'il est parcouru dans le sens des ordonnées croissantes ou décroissantes. À un parcours de frontière à partir de p est donc associée une suite de lettres de l'alphabet $\{d, g, h, b\}$: c'est le *mot de contour* de F.

Question 2.2 : *Quel est le mot de contour de la figure*

$$\{(1,1),(2,1),(3,1),(2,2),(3,2),(4,2),(2,3)\} \ ?$$

On définit le type de données

```
type contour= array[1..M] of char;
```

où M est une constante supposée assez grande pour nos besoins.

Question 2.3 : *Ecrire une procédure*

```
procedure mot-de-contour(fig: figure; var cont: contour; var long: integer);
```

qui, à partir d'une figure fig supposée connexe sans trou, détermine le nombre long de lettres de son mot de contour et remplit le tableau cont de telle façon que cont[i] soit la i-ième lettre du mot de contour pour tout i, $1 \leq i \leq long$. On ne demande pas de tester la connexité de la figure, ni l'absence de trou.

Hypothèse simplificatrice : *On supposera que la figure fig* ne touche pas les bords, *c'est-à-dire que $fig[1,i] = fig[i,1] = fig[N,i] = fig[i,N] = false$ pour tout i, $1 \leq i \leq N$. On ne testera pas cette hypothèse.*

Question 2.4 : *Montrer que le nombre de cases d'une figure connexe sans trou peut être calculé de façon simple à partir de son mot de contour.*

Question 2.5 : *Ecrire une procédure*

```
procedure surface-contour(cont: contour; var s: integer);
```

qui prenne pour donnée un mot de contour cont définissant une figure F et calcule la surface s de F (son nombre de cases).

Partie 3 : Pavage du trapèze par des dominos

Un *domino* est une figure formée de deux cases $\{(i,j),(i,j\pm1)\}$ (domino *vertical*) ou $\{(i,j),(i\pm1,j)\}$ (domino *horizontal*). Un *pavage* d'une figure F est un recouvrement de la figure par des dominos inclus dans F n'ayant aucune case commune deux à deux.

On suppose dans toute la suite que le plan est muni d'un *coloriage du plan en échiquier* comme suit : la case (i,j) est dite *blanche* si $i+j$ est pair, et *noire* sinon. Une figure F est dite *équilibrée* si elle contient le même nombre de cases noires que de cases blanches.

Question 3.1 : *Montrer que toute figure pavable par des dominos est équilibrée.*

Question 3.2 : *Montrer par un contre-exemple que la réciproque est fausse.*

Question 3.3 : *Montrer qu'un trapèze est pavable par des dominos si et seulement s'il est équilibré.*

Indication : *Si F a une "marche" assez large ($min_j - min_{j+1} \geq 2$ par exemple), poser un domino horizontal au bout de cette marche. Si aucune "marche" n'est assez large, essayer de poser un domino vertical au coin d'un marche suffisamment haute.*

On considère le type de données suivant :

```
type pavage= array [1..N,1..N] of integer;
var res: pavage;
```

Un pavage de F par des dominos $D_1, \ldots, D_n$, est représenté de la manière suivante : $res[i,j] = 0$ si la case (i,j) n'appartient pas à F, $res[i,j] = k$ si la case (i,j) est dans la figure F et est couverte par le domino D_k.

Question 3.4 : *Ecrire une procédure*

```
procedure pavage-de-trapeze(trapeze: figure, var pavable: boolean, var resultat: pavage);
```

qui teste si un trapèze est pavable par des dominos et en donne un pavage s'il en existe.

Partie 4 : Liens entre les différents pavages d'une même figure

Soit F une figure et c_0 une case donnée de F. Soient P_1 et P_2 deux pavages de F par des dominos. On définit la suite de cases $(c_i)_{i\geq 0}$ par récurrence de la manière suivante pour $i \geq 1$:

- si i est impair, alors c_i est l'unique case telle que $\{c_{i-1}, c_i\}$ soit un domino de P_1 ;
- si i est pair, alors c_i est l'unique case telle que $\{c_{i-1}, c_i\}$ soit un domino de P_2.

Question 4.1 : *Montrer que la suite $(c_i)_{i\geq 0}$ est périodique, et que sa période $k > 0$ est paire. A quelle condition est-elle égale à 2 ?*

Considérons un cycle de (c_i) de la forme $c_p, c_{p+1}, \ldots, c_{p+k} = c_p$ avec p pair. Soit P_2' l'ensemble obtenu à partir de P_2 en remplaçant les dominos suivants de P_2

$$\{c_{p+1}, c_{p+2}\}, \{c_{p+3}, c_{p+4}\}, \ldots, \{c_{p+k-1}, c_{p+k}\}$$

par les dominos suivants de P_1

$$\{c_p, c_{p+1}\}, \{c_{p+2}, c_{p+3}\}, \ldots, \{c_{p+k-2}, c_{p+k-1}\}.$$

On appelle *rotation* toute transformation de ce type.

Question 4.2 : *Montrer que P_2' est aussi un pavage de F. Donner une condition suffisante pour que ce nouveau pavage P_2' soit différent du pavage initial P_2. Est-elle nécessaire ? Montrer que P_2 peut être transformé en P_1 par une suite finie de rotations.*

Question 4.3 : *Ecrire une procédure*

```
procedure rotation(P1, P2: pavage ; i0, j0: integer; var resultat: pavage);
```

qui, étant donnés deux pavages P_1 et P_2 d'une même figure F, et une case (i_0, j_0) de F, construit le pavage P_2' obtenu en appliquant une rotation de P_2 à partir de la case $c_0 = (i_0, j_0)$.

Partie 5 : Dénombrement des pavages du carré

Soit x_n le nombre de pavages du carré $\{(i,j) \mid 1 \leq i \leq n,\ 1 \leq j \leq n\}$ par des dominos. Remarquer que $x_n = 0$ si n est impair. On suppose donc n pair.

Question 5.1 : *Montrer que*

$$2^{n^2/4} \leq x_n \leq 4^{n^2/2}.$$

Question 5.2 : *Montrer de plus que $x_n \leq 2^{n^2/2}$.*

Question 5.3 : *On suppose les cases noires du carré numérotées de 1 à $n^2/2$ et les cases blanches du carré numérotées de 1 à $n^2/2$. Montrer que*

$$x_n = \sum_{\sigma \in \mathcal{S}_{n^2/2}} |a_{1,\sigma(1)} \cdot a_{2,\sigma(2)} \cdots a_{n^2/2,\sigma(n^2/2)}|$$

avec

- $|a_{k,\ell}| = 1$ *si la case noire k est adjacente à la case blanche ℓ,*
- $a_{k,\ell} = 0$ *sinon,*
- $\mathcal{S}_{n^2/2}$ *est l'ensemble des permutations de* $\{1, \ldots, n^2/2\}$.

On définit $a_{k,\ell}$ comme suit :

- si les cases k et ℓ sont l'une au-dessus de l'autre sur une même verticale, c'est-à-dire (i,j) et $(i, j \pm 1)$, alors $a_{k,\ell} = 1$.
- si les cases k et ℓ sont adjacentes et sur une même horizontale, c'est-à-dire (i,j) et $(i \pm 1, j)$ alors $a_{k,\ell} = 1$ si j est pair et -1 si j est impair.

Question 5.4 : *Soit $A = (a_{k,\ell})_{1 \leq k,\ell \leq n^2/2}$. Montrer que $x_n = |\det A|$.*

Question 5.5 : *En déduire une procédure de calcul de x_n.*