Option Informatique en Spé MP et MP*

Le problème du loueur de skis

Devoir à rendre après les vacances de la Toussaint

Résumé

La première partie demande la rédaction de quelques fonctions en Caml.

La deuxième partie propose le dénombrement, puis la génération d'objets combinatoires classiques : injections de $[\![1, p]\!]$ dans $[\![1, n]\!]$, applications strictement croissantes de $[\![1, p]\!]$ dans $[\![1, n]\!]$.

La troisième partie vous amène à résoudre un problème d'optimisation combinatoire, au moyen d'une technique connue sous le nom de *programmation dynamique*.

Enfin, la quatrième partie replace le problème du loueur de skis dans le cadre général des matrices de Monge.

En cas de problème dans l'énoncé, vous pouvez me contacter par téléphone (06 83 09 73 20) ou par mail (`bruno@enix.org`). Avec un peu de chance, il y aura des informations sur la page Web de l'option informatique :

`http://bruno.maitresdumonde.com/optinfo/Spe-MP/dmds2001/mongeski`

$\boxed{\textit{Veuillez rédiger chaque partie sur une copie séparée.}}$

Table des matières

1 Boîte à outils Caml

► L'objectif de cette partie est la construction d'une *boîte à outils*, c'est-à-dire une panoplie de fonctions simples, auxquelles vous pourrez faire appel dans les autres parties (même si vous ne les avez pas rédigées).

► L'emploi de références ou de structures de données mutables est interdit. Le recours à des boucles devra se faire avec parcimonie et à bon escient. Pensez fonctionnel, rédigez fonctionnel, soyez fonctionnel !

Question 1 • Rédigez en Caml une fonction :

```
intervalle : int -> int -> int list
```

spécifiée comme suit : `intervalle p q` construit l'intervalle discret $[\![p, q]\!]$. Par exemple, `intervalle 3 7` rendra la liste `[3;4;5;6;7]`.

Question 2 • Rédigez en Caml une fonction :

```
ajoute_devant : int -> int list list -> int list list
```

spécifiée comme suit : `ajoute_devant x l` ajoute x en tête de chacune des listes membres de la liste de listes ℓ. Par exemple, `ajoute_devant 7 [[2;7];[];[7;1;3]]` rendra la liste `[[7;2;7];[7];[7;7;1;3]]`.

Question 3 • Rédigez en Caml une fonction :

```
somme_bloc : int vect vect -> int -> int -> int -> int -> int
```

spécifiée comme suit : `somme_bloc m l k l' k'` calcule la somme des coefficients de la matrice m dont l'indice de ligne est compris entre l et l' (inclus) et l'indice de colonne entre k et k' (inclus).

2 Injections, injections croissantes

► Soient n et p deux naturels vérifiant $1 \leqslant p \leqslant n$. Notons $\mathrm{a}(n,p)$ le nombre d'injections de $[\![1, p]\!]$ dans $[\![1, n]\!]$, et $\mathrm{b}(n,p)$ le nombre d'applications strictement croissantes de $[\![1, p]\!]$ dans $[\![1, n]\!]$.

Question 4 • Donnez (preuves à l'appui) les expressions respectives de $\mathrm{a}(n,p)$ et $\mathrm{b}(n,p)$.

Question 5 • L'étudiant Jean-Maurice Malencontreux propose de calculer $\mathrm{b}(n,p)$ au moyen des fonctions suivantes :

```
let produit = it_list (prefix *) 1;;
let factorielle n = produit (intervalle 1 n);;
let b n p = (factorielle n)/(factorielle p)/(factorielle (n-p));;
```

Hélas, lorsqu'il a voulu calculer $\mathrm{b}(13,1)$ il a obtenu un résultat nul. Expliquez, commentez ...

Question 6 • Rédigez en Caml une fonction :

```
b : int -> int -> int
```

spécifiée comme suit : `b n p` calcule $\mathrm{b}(n,p)$. Vous tiendrez compte des obligations suivantes : le nombre d'opérations (multiplications, divisions) ne doit pas être supérieur à $2\min(p, n-p)$; toutes les divisions effectuées doivent ≪tomber juste≫ ; aucun résultat intermédiaire ne doit être supérieur à $p\,\mathrm{b}(n,p)$.

Question 7 • Rédigez en Caml une fonction :

```
injections : int -> int -> int list list
```

spécifiée comme suit : `injections n p` construit la liste des injections de $[\![1, p]\!]$ dans $[\![1, n]\!]$; une injection f sera représentée par la liste $\big(f(j)\big)_{1 \leqslant j \leqslant p}$. L'ordre dans lequel les injections sont énumérées est indifférent.

Question 8 • Rédigez en Caml une fonction :

```
injections_croissantes : int -> int -> int list list
```

spécifiée comme suit : `injections_croissantes n p` construit la liste des injections croissantes de $[\![1, p]\!]$ dans $[\![1, n]\!]$, le résultat étant présenté comme à la question précédente.

3 Le problème du loueur de skis

Like the ski resort full of girls hunting for husbands and husbands hunting for girls, the situation is not as symmetrical as it might seem.

Alan Lindsay Mackay

▶ Un loueur de skis dispose de n paires de skis ; p clients se présentent dans sa boutique, avec $p \leqslant n$. Chaque skieur sera d'autant plus content que la longueur de la paire de ski qui lui est affectée est proche de sa propre taille. Notant s_i la longueur de la i-ième paire de skis, et c_j la taille du j-ième client, le problème du loueur est donc de trouver une affectation (c'est-à-dire une injection f de $[\![1, p]\!]$ dans $[\![1, n]\!]$) qui minimise la quantité

$$\|f\| = \sum_{1 \leqslant j \leqslant p} |c_j - s_{f(j)}|$$

Nous noterons $\gamma(c, s)$ la valeur minimale de $\|f\|$; une affectation f vérifiant $\|f\| = \gamma(c, s)$ sera dite *optimale*.

▶ Dans les estimations de coût, vous ne compterez que les opérations élémentaires (addition, soustraction, comparaison).

Question 9 • La méthode de la force brute consiste à examiner toutes les affectations possibles. Donnez une estimation du coût de cette méthode dans chacun des trois cas suivants :

- $p = \mathcal{O}(1)$;
- $p = n - \mathcal{O}(1)$;
- $n = 2p$.

Question 10 • Dans cette question, nous supposons $p = 1$: un seul client s'est présenté (au grand dam du loueur). Décrivez un algorithme réalisant une affectation optimale, dans chacun des deux cas suivants :

- les skis sont rangés dans un ordre quelconque ;
- les skis sont rangés par ordre croissant.

Question 11 • Dans cette question, nous supposons $p = n = 2$, $c_1 \leqslant c_2$ et $s_1 \leqslant s_2$. Deux affectations sont possibles : $f : j \in [\![1, 2]\!] \mapsto j$ et $g : j \in [\![1, 2]\!] \mapsto 3 - j$. Montrez que $\|f\| \leqslant \|g\|$.

▶ Supposons désormais que les skis et les clients sont rangés par ordre croissant : $c_j \leqslant c_{j+1}$ pour $1 \leqslant j < p$ et $s_k \leqslant s_{k+1}$ pour $1 \leqslant k < n$. Pour $1 \leqslant j \leqslant p$, notons $c[1..j]$ la liste des j plus petits clients ; définissons de même $s[1..k]$ pour $1 \leqslant k \leqslant n$. Enfin, notons $\gamma_{j,k} = \gamma\big(c[1..j], s[1..k]\big)$.

Question 12 • Supposons $n = p$; exhibez une affectation optimale. Quel est le coût de son obtention ?

Question 13 • Revenons maintenant au cas général $n \geqslant p$. Justifiez la relation suivante, pour $2 \leqslant p < n$:

$$\gamma(c, s) = \min\Big(\gamma_{p,n-1}, |c_p - s_n| + \gamma_{p-1,n-1}\Big)$$

Question 14 • En déduire un algorithme calculant $\gamma(c, s)$ pour un coût $\mathcal{O}(n^2)$. Quelle est la taille de l'espace mémoire nécessaire pour la mise en œuvre de cet algorithme ?

Question 15 • Rédigez en Caml une fonction :

```
gamma : int list -> int list -> int
```

spécifiée comme suit : `gamma c s` calcule $\gamma(c, s)$.

Question 16 • Comment le loueur peut-il déduire, des calculs précédents, une affectation optimale ?

Question 17 • Rédigez en Caml une fonction :

```
affectation : int list -> int list -> int list
```

spécifiée comme suit : `affectation c s` construit une affectation optimale f, qui sera représentée par la liste $\big(f(j)\big)_{1 \leqslant j \leqslant p}$.

4 Matrices de Monge

► Notons $\mathcal{M}_{n,p}(E)$ l'ensemble des matrices à n lignes et p colonnes, à coefficients dans E. Nous dirons que $m \in \mathcal{M}_{n,p}(\mathbb{R})$ possède la propriété de MONGE si $m_{i,j} + m_{\ell,k} \leqslant m_{i,k} + m_{\ell,j}$ dès que $1 \leqslant i \leqslant \ell \leqslant n$ et $1 \leqslant j \leqslant k \leqslant p$.

Question 18 • Soient $(c_i)_{1\leqslant i\leqslant n}$ et $(s_j)_{1\leqslant j\leqslant p}$ deux listes de réels, classées par ordre croissant : $s_i \leqslant s_{i+1}$ pour $i \in [\![1, n-1]\!]$ et $c_j \leqslant c_{j+1}$ pour $j \in [\![1, p-1]\!]$. Montrez que la matrice m à n lignes et p colonnes définie par $m_{i,j} = |s_i - c_j|$ possède la propriété de MONGE.

Question 19 • Montrez que $m \in \mathcal{M}_{n,p}(\mathbb{R})$ possède la propriété de MONGE ssi elle vérifie la propriété suivante : $m_{i,j} + m_{i+1,j+1} \leqslant m_{i,j+1} + m_{i+1,j}$ quels que soient $i \in [\![1, n-1]\!]$ et $j \in [\![1, p-1]\!]$.

Question 20 • Rédigez en Caml une fonction :

```
est_de_monge : int vect vect -> bool
```

spécifiée comme suit : `est_de_monge m` détermine si la matrice m possède la propriété de MONGE.

► Soit $d \in \mathcal{M}_{n,p}(\mathbb{R}_+)$. Définissons une matrice $\widehat{d} \in \mathcal{M}_{n,p}(\mathbb{R})$ par $\widehat{d}_{i,j} = \sum_{i\leqslant \ell\leqslant n} \sum_{1\leqslant k\leqslant j} d_{\ell,k}$ quels que soient $i \in [\![1, n]\!]$ et $j \in [\![1, p]\!]$.

Question 21 • Calculez $\widehat{d}$ lorsque $d = \begin{pmatrix} 1 & 4 & 2 \\ 3 & 1 & 6 \end{pmatrix}$.

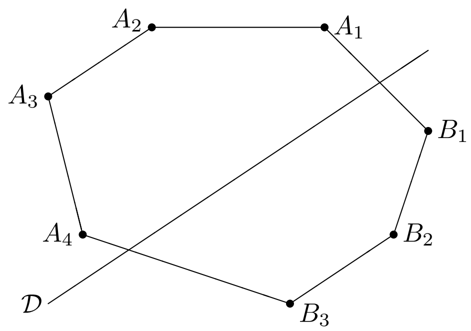
Question 22 • Rédigez en Caml une fonction :

```
chapeau : int vect vect -> int vect vect
```

spécifiée comme suit : `chapeau d` construit la matrice $\widehat{d}$.

Question 23 • Montrez que si d est à coefficients positifs ou nuls, alors $\widehat{d}$ possède la propriété de MONGE.

► Soient $\mathcal{P}$ un polygone convexe à n sommets, et $\mathcal{D}$ une droite ne passant par aucun sommet de $\mathcal{P}$, mais coupant celui-ci en deux. Numérotons les sommets situés d'un côté de $\mathcal{D}$ de A_1 à A_p dans le sens direct, et ceux situés de l'autre côté de $\mathcal{D}$ de B_1 à B_{n-p} dans le sens rétrograde.



Question 24 • Montrez que la matrice Δ à p lignes et $n-p$ colonnes définie par $\Delta_{i,j} = d(A_i, B_j)$ possède la propriété de MONGE.

► Soit $m \in \mathcal{M}_{n,p}(\mathbb{R})$. Pour $i \in [\![1, n]\!]$, définissons γ_i comme le plus petit indice j tel que $m_{i,j} = \min_{1\leqslant k\leqslant p} m_{i,k}$. Ainsi, le coefficient m_{i,γ_i} est le plus petit de la ligne i, et, si le minimum est atteint par plusieurs coefficients, c'est celui qui est le plus à gauche.

Question 25 • Montrez que la détermination de γ_i requiert au moins $p-1$ comparaisons.

Question 26 • Supposons que m possède la propriété de MONGE. Montrez que la suite $(\gamma_i)_{1\leqslant i\leqslant n}$ est croissante.

Question 27 • Supposons que m possède la propriété de MONGE. Montrez que l'on peut déterminer la suite $(\gamma_i)_{1\leqslant i\leqslant n}$ en effectuant $\mathcal{O}(p \lg n)$ comparaisons.

FIN

[mongeski] *Version du 20 février 2008*