

Travaux dirigés n° 5

Karelle JULLIAN

Lycée Kléber
MPSI, Option Info

2017/2018

Exercice 1 (Carrés magiques)

Un carré magique est une matrice de taille $N * N$ contenant tous les nombres de 1 à N^2 telle que les sommes des nombres de chaque ligne, chaque colonne, chaque diagonale soient chacune égale à une constante. Exemple :

4	9	2
3	5	7
8	1	6

- 1 Quelle est la constante d'un carré magique d'ordre N ?
- 2 Définir le type `Matrice` qui représentera un tableau à deux dimensions dont les indices commencent à 1.
- 3 Ecrire une fonction prenant en argument une matrice m d'ordre n et retournant vrai ou faux si la matrice est un carré magique.
- 4 Un algorithme de construction est le suivant :
 - placer les nombres suivant leur ordre croissant : 1 puis 2, puis 3, ...

- commencer par la case juste au-dessous de la case du milieu ;
- se déplacer en diagonale vers le bas et vers la droite quand c'est possible ;
- remplir comme si chaque ligne et chaque colonne était un tore ;
- si une case est déjà remplie, on avance encore d'une ligne et on recule d'une colonne.

Voici le début de la construction du carré quand $N = 3$:

4		2
3		
	1	

Représenter le carré magique d'ordre 5 ainsi construit.

- 5 Ecrire le programme qui remplit une matrice m de taille n suivant l'algorithme.

Exercice 2 (Hörner.)

Un polynôme est représenté par la liste de ses coefficients, en commençant par le plus haut degré :

```
type polynome == float list
```

Ecrire la fonction

```
horner : float list -> float -> float
```

*qui à un polynôme p et à un réel a fait correspondre $p(a)$.
Donner le coût en nombre de multiplications et d'additions.*

Exercice 3 (Médian de deux tableaux.)

- ① *On se donne deux tableaux triés de longueur $n = 2^k$. Ecrire une fonction*

```
median : 'a vect -> 'a vect -> 'a
```

utilisant la méthode Diviser Pour Régner qui renvoie le médian de ces deux tableaux.

- ② *Modifier cette fonction pour qu'elle accepte des tableaux de longueur quelconque. Quel est son coût ?*

Exercice 4 (Fibonacci.)

La suite de Fibonacci est définie par $F_0 = F_1 = 1$ et la relation $F_{n+1} = F_n + F_{n-1}$, $n \geq 1$.

- 1 Ecrire une fonction

`fibonacci : int -> int`

qui, à un entier n fait correspondre F_n . Coût en additions ?
(ne pas faire un calcul naïf, tâcher de minimiser le coût)

- 2 Montrer la relation

$$F_{n+p} = F_n F_p + F_{n-1} F_{p-1}.$$

En déduire un calcul par la méthode Diviser Pour Régner de F_n . Quel est le coût en multiplications et en additions ?

Exercice 5 (liste de billets.)

Disposant d'une liste d'entiers 1 et d'un entier s , on cherche une sous-liste m dont les éléments aient pour somme l'entier s .

L'algorithme est le suivant

- si la liste est vide et si la somme est nulle la solution est la liste vide, sinon pas de solution
- si la liste n'est pas vide $a : : q$, deux cas possibles :
 - * on tente une solution en gardant a et en cherchant une sous-liste de q dont les éléments aient pour somme $s-a$
 - * si cela ne marche pas, on "revient en arrière" et on tente une solution en ne choisissant que des éléments de q .

Exercice 6 (puissance.)

Ecrire la fonction

`puissance : int -> int -> int`

définie par $\text{puissance } x \ n = x^n$. Cette fonction devra utiliser la méthode Diviser Pour Régner, d'abord sous forme récursive, puis sous forme itérative. On donnera une preuve.

Exercice 7 (plus grand élément par dichotomie.)

Ecrire une fonction récursive travaillant par dichotomie

`max_dich : 'a vect -> int -> int -> 'a`

qui, à un tableau t d'entiers et deux entiers i et j fait correspondre le plus grand élément du tableau t situé entre i et j .

(La méthode est aussi bien adaptée à la recherche du deuxième plus grand élément).

Exercice 8

Une stratégie est dite admissible lorsque les cordes ne se coupent pas.

On admet que cette notion de dépend pas de la position des points sur le cercle (du moment qu'ils sont ordonnés).

① *On donne deux stratégies*

$s1$	2	1	6	5	4	3	8	7
$s2$	2	1	7	5	4	8	3	6

Représentez-les, et précisez si elles sont admissibles.

② *Dans le cas $n = 3$, donnez le nombre de stratégies possibles, admissibles ou pas.*

- 3 Représentez parmi elles celles qui sont admissibles.
- 4 Déterminez le nombre de stratégies de taille n , admissibles ou non
- 5

Exercice 9 ()

Dans un jeu de 32 cartes, on attribue la valeur

- 0 aux cartes de 7 à 10 et à l'as ;
 - 1 au valet ;
 - 2 à la dame ;
 - 3 au roi.
- 1 Définir le type carte
 - 2 Ecrire une fonction `valeur : carte list -> int` qui à une main, représentée par une liste, fait correspondre sa valeur. Cette fonction sera récursive.

Exercice 10 ()

On représente un ensemble d'entiers par une liste sans doublons.

- ❶ *Définir le type ensemble.*
- ❷ *Ecrire une fonction `insertion_ens` de type `int -> ensemble -> ensemble` telle que `insertion_ens v E` renvoie un ensemble constitué des mêmes éléments que `E` ainsi que de l'élément `v` si celui-ci n'y figurait pas déjà. Cette fonction devra être récursive ou faire appel à des fonctions auxiliaires récursives.*
- ❸ *Donner une estimation du nombre d'appels récursifs effectués en fonction du nombre d'éléments de `E`.*
- ❹ *Ecrire une fonction `elimination_ens` de type `int -> ensemble -> ensemble` telle que `elimination_ens v E` renvoie une ensemble contenant les mêmes entiers que `E` sauf l'entier `v` s'il y figurait.*
- ❺ *Donner des exemples de valeurs de `v` et de `E` pour lesquelles le nombre d'appels récursifs à `elimination_ens` correspond au meilleur et au pire des cas.*

Exercice 11 ()

Dans une facture doivent figurer les informations suivantes

- *nom_client ;*
- *date_facture : jour*mois*année ;*
- *livré : booléen ;*
- *règlement : indique le mode de règlement*
 - si le client règle à crédit : la valeur attribuée sera credit*
 - si le client règle à comptant : la valeur attribuée sera comptant*
- suivi d'une mention Chèque ou Espèce ou Carte_bancaire ;*
- *liste_achats : désigne la liste des achats effectués.*

Chaque ligne d'achat mentionne :

- *code de l'article : code entier.*
- *quantité.*

Définir les types mention, mode_
règlement, ligne_achat et facture, puis un exemple_client.

Exercice 12 ()

Typez et évaluez les phrases suivantes (sans machine), écrites dans une même session CAML :

```
1- #let a = ref (5-3*2);;  
2- #let b= let x =3 and y=4,2 in  
    !a + (snd y)*x, !a*x+ fst y;;  
3- #let h =function  x -> function y -> (y*(x y));;  
4- # a:= h (fun x -> x+x) 5;;  
5- #a;;
```

Exercice 13 (même chose avec)

```
1- #let alpha = function a -> function b  
    -> function c ->
```

```

let rec gamma = function
  [] -> []
  | (x::R) -> if x=a then b::(gamma R) else x::(gamma R)
in gamma c;;
2- #let delta = alpha 5 8;;
3- #delta ([1;5;8;9;5;4;1;3;5;9;4]);;
4- # let rec omega a = function [] -> 0
    | x::R -> (a*x)+ (omega (a+1) R);;
5-#let epsilon = alpha 9 1 [1;5;8;9;5;8;1;1;5;9]
    in omega 0 (alpha 8 1 (delta epsilon));;

```

Exercice 14 ()

*On définit un type nombre de manière à regrouper des valeurs entières, réelles et complexes (eux-mêmes représentés sous la forme float*float).*

- ① Construire le type nombre.
- ② Définir une fonction carre de type nombre → nombre et qui renvoie le carré du nombre pris en argument.

Exercice 15 ()

Un centre de formation propose des formations donnant lieu à des stages.

Une formation est représentée en CAML par une paire dont le premier élément est son numéro de code formation (entier), et dont le deuxième élément est un enregistrement défini par :

- *un intitulé de la formation (25 caractères),*
 - *un nombre de séances de formation,*
 - *un niveau de progression : Initiation ou Perfectionnement,*
 - *le nombre maximum de stagiaires,*
 - *le tarif de formation (nombre réel).*
- ① *Définir les types enreg_formation, niveau_de_progression et formation.*
 - ② *La journée de travail est découpée en deux sessions : la session du matin et celle de l'après-midi.*

Chaque séance de stage mobilise une session complète, soit le matin , soit l'après-midi. La session est simplement représentée par un littéral : MATIN ou APRES_MIDI.

Une séance de stage est définie par :

- *la date de la séance,*
- *la session,*
- *le code salle de formation affectée à la séance (les salles ont un code formé par la lettre correspondant au bâtiment A ou B, suivie d'un simple numéro de salle).*

Un stage correspondant à une formation programmée dans un calendrier, il est défini par :

- *le code du stage : numéro du stage dans l'année,*
- *le code de la formation correspondante,*
- *le nom du formateur (libellé de 25 caractères),*
- *la liste des séances.*

Donner, en CAML, une définition des types de données date, code_salle, session_type et séance.

Exercice 16 ()

On considère les éléments d'un ensemble fini A comme étant les ceux d'une liste CAML.

Ecrire une fonction parties qui retourne une liste correspondant à l'ensemble $P(A)$ des parties de A .

```
syntaxe = parties liste
```

```
typage = parties : 'a list -> 'a list list
```

```
exemple :
```

```
#parties [1; 2; 3; 4] ;;
```

```
- : int list list = [[1; 2; 3; 4]; [1; 2; 3];
```

```
[1; 2; 4]; [1; 2]; [1; 3; 4]; [1; 3]; [1; 4];
```

```
[1]; [2; 3; 4]; [2; 3]; [2; 4]; [2]; [3; 4];
```

```
[3]; [4]; []]
```