

# INFO0054 - Programmation fonctionnelle

## Répétition 7: Divers

Jean-Michel BEGON

10 Mars 2016

---

### Exercice 1.

Écrire une fonction `sublists` prenant comme argument une liste `l` et retournant la liste des sous-listes de `l`. (Une sous-liste de `l` est une liste de 0, 1 ou plusieurs éléments consécutifs de `l`.)

---

### Exercice 2.

Écrire la fonction `partitions` qui renvoie la liste des partitions d'un ensemble donné. On représente un ensemble par une liste sans répétitions.

```
(partitions '(a b c)) =>
'(((a) (b) (c)) ((a) (b c)) ((a b) (c)) ((b) (a c)) ((a b c)))
```

---

### Exercice 3.

Un arbre  $n$ -aire est un arbre dont chaque nœud admet un nombre quelconque de fils ; seules les feuilles sont étiquetées, l'étiquette étant un nombre naturel. La fonction  $f$  prend comme argument un arbre  $n$ -aire  $a$  et renvoie l'arbre  $n$ -aire  $f(a)$  obtenu en remplaçant chaque feuille de  $a$  étiquetée  $n > 0$  par un nœud interne dont les  $n$  fils sont des feuilles étiquetées  $n - 1$ . On demande de programmer la fonction  $f$ .

---

### Exercice 4.

Écrire une fonction `longest-inc` retournant la plus longue sous-liste croissante de la liste d'entiers donnée en argument.

---

### Exercice 5.

Écrire une fonction `nbsum` qui prend comme argument un nombre  $n$  et qui renvoie le nombre de façons d'écrire une somme égale à  $n$  (on comptera une seule fois les commutations).

---

### Exercice 6.

Écrire la fct `lagrange`, prenant comme argument un entier naturel  $n$  et renvoyant la liste de tous les quadruplets d'entiers naturels  $(a, b, c, d)$  tels que  $a^2 + b^2 + c^2 + d^2 = n$ .

```
(lagrange 13) ==>
((0 0 2 3) (0 0 3 2) (0 2 0 3) (0 2 3 0) (0 3 0 2) (0 3 2 0)
 (1 2 2 2) (2 0 0 3) (2 0 3 0) (2 1 2 2) (2 2 1 2) (2 2 2 1)
 (2 3 0 0) (3 0 0 2) (3 0 2 0) (3 2 0 0))
```

```
(lagrange 18)
((0 0 3 3) (0 1 1 4) (0 1 4 1) (0 3 0 3) (0 3 3 0) (0 4 1 1)
```

(1 0 1 4) (1 0 4 1) (1 1 0 4) (1 1 4 0) (1 2 2 3) (1 2 3 2)  
 (1 3 2 2) (1 4 0 1) (1 4 1 0) (2 1 2 3) (2 1 3 2) (2 2 1 3)  
 (2 2 3 1) (2 3 1 2) (2 3 2 1) (3 0 0 3) (3 0 3 0) (3 1 2 2)  
 (3 2 1 2) (3 2 2 1) (3 3 0 0) (4 0 1 1) (4 1 0 1) (4 1 1 0)

---

**Exercice 7.**

$$f(n) = \sum_{i=0}^{n-1} f(n-i-1)^{f(i)}$$

---

**Exercice 8.**

Ecrire la fonction `conway` qui calcule le *nième* terme de la suite de Conway :

1  
 1, 1  
 2, 1  
 1, 2, 1, 1  
 1, 1, 1, 2, 2, 1