

INFO0054 - Programmation fonctionnelle

Répétition 4: Récursion sur les nombres et sur les listes

Jean-Michel BEGON

06 Mars 2018

Représentation en mémoire

Exercice 1.

Représenter le résultat en mémoire des opérations suivantes :

```
'(2)                                '((3 4 5))
(cons 1 2)                          (list '(1 2) '(3 4 5))
(list 1 2)                          (cons '(1 2) '(3 4 5))
'(3 4 5)                            (append '(1 2) '(3 4 5))
```

Listes

Exercice 2.

Définir la fonction `suffix` à deux arguments `l1` et `l2` dont les valeurs sont des listes et qui retourne `#t` si `l1` est suffixe de `l2` et `#f` sinon.

```
(suffix '(1 2) '(3 x 1 2)) ⇒ #t
(suffix '(1 2) '(3 x 2)) ⇒ #f
```

Exercice 3.

Écrire une fonction `frequency` prenant en argument une liste d'atomes `ls` et renvoyant une table d'apparition de chacun des atomes dans la liste `ls`. Cette table sera représentée par une liste de paires pointées, dont le `car` est un atome et le `cdr` le nombre d'occurrences de cet atome. Tous les atomes de `ls` apparaissent une et une seule fois dans la table.

```
(frequency '(a b c b a b d a c b b))
⇒ ((a . 3) (b . 5) (c . 2) (d . 1))
```

Polynômes

Exercice 4.

Écrire une fonction qui à tout polynôme $P(x)$ et naturel n associe la dérivée n -ième du polynôme $\frac{d^n P(x)}{d^n x}$.

On représentera un polynôme par la liste de ses coefficients, par ordre croissant des degrés. $\Rightarrow a + bx + cx^2 + \dots$ sera représenté par la liste `(a b c ...)`.

Exercice 5.

Écrire une fonction qui à deux polynômes P et Q associe le polynôme composé $P \circ Q$.
On représentera un polynôme par la liste de ses coefficients, par ordre croissant des degrés.
 $\Rightarrow a + bx + cx^2 + \dots$ sera représenté par la liste (a b c ...).

Les diviseurs

Exercice 6.

Ecrire une fonction `div-ls` qui retourne la liste des diviseurs d'un entier strictement positif.

Exercice 7.

Un entier naturel est *parfait* s'il est égal à la somme de ses diviseurs positifs propres. Ecrire un prédicat `perfect?` déterminant si son argument est un nombre parfait.

Le nombre 28 est parfait parce que $28 = 1 + 2 + 4 + 7 + 14$;
le nombre 32 n'est pas parfait parce que $32 \neq 1 + 2 + 4 + 8 + 16$.

(`perfect? 28`) \Rightarrow #t

(`perfect? 32`) \Rightarrow #f

Exercice 8.

Écrire un prédicat `prime?` qui détermine si un entier strictement positif est premier ou non.

Exercices proposés

Exercice 9.

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

Exercice 10.

Implémenter le quicksort, spécifier les fonctions auxiliaires et discuter sa complexité.

Exercice 11.

Implémenter le tri par fusion, spécifier les fonctions auxiliaires et discuter sa complexité.