
Programmation fonctionnelle

Répétition 4

12 mars 2015

Correction exercice proposé

Exercice 1.

Écrire une fonction `filter` prenant deux arguments, une liste `l` d'objets et un prédicat unaire `p`, renvoyant la liste des éléments de `l` pour lesquels `p` est vrai.

Définir alors au moyen de la fonction précédente, la fonction `greater` prenant comme arguments une liste `l` de nombres entiers et un nombre entier `a`, et renvoyant la liste des éléments de `l` strictement supérieurs à `a`.

`(greater '(5 2 7 4 3 6) 4) ⇒ (5 7 6)`

Expression à évaluer

Exercice 2.

Quelle est la valeur de cette expression ?

```
((lambda (x) (list x (list (quote quote) x)))  
 (quote (lambda (x) (list x (list (quote quote) x)))))
```

Récursion profonde sur les listes

Exercice 3.

Définir la procédure `count-all` à deux arguments, un élément et une liste, et qui compte, en profondeur, le nombre de de fois que l'élément est contenu dans la liste.

`(count-all 1 '(0 (1 2 (3 4 (1))) (3 (2 1) 1) 1) 0 (1 2 (1 2 3)))) ⇒ 7`

Compositions de fonctions

Exercice 4.

Définir une fonction `compose-n` qui renvoie la fonction unaire donnée en argument `n` fois composée avec elle-même.

Variante :

Écrire une fonction `compose-fgf` qui prend comme argument une fonction f et renvoie une fonction qui prend comme argument une fonction g et qui renvoie la fonction $f \circ g \circ f$.

Variante 2 :

Écrire une fonction `compose-fgab` qui prend comme argument deux fonctions f et g , ainsi que deux entiers a et b et renvoie la fonction

$$\underbrace{f \circ \dots \circ f}_a \circ \underbrace{g \circ \dots \circ g}_b$$

Variante 3 :

Écrire une fonction `compose-fa` qui prend comme argument une fonction f et deux entiers a, b et renvoie la fonction

$$x \mapsto \underbrace{f \circ \dots \circ f}_a(b^x)$$

Arbres

Exercice 5.

Tout nœud interne d'un arbre arithmétique a exactement deux fils et est étiqueté par l'un des symboles `add`, `sub`, `mul` et `div`; toute feuille est étiqueté par un nombre entier.

Écrire la fonction `value` qui prend comme argument un arbre arithmétique et qui renvoie la valeur de l'expression arithmétique associée si l'évaluation de celle-ci n'implique aucune division par 0; sinon, `value` renvoie `#f`.

`(value '(mul (add (add 3 5) (sub 3 4)) (div 3 2.0)))` \Rightarrow 10.5

Somme

Exercice 6.

$$f(n) = \sum_{i=0}^{n-1} (((f(i) + 2) * (f(n - i - 1) + 3)) \bmod (n^2 + i + 5))$$

Exercices proposés

Exercice 7.

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

Exercice 8.

Un carré magique de dimension n est un tableau de taille $n \times n$ dans lequel chaque nombre de 1 à $n \times n$ apparaît une fois, et dont les sommes des lignes, des colonnes et des diagonales sont égales. Écrire un prédicat `magic?` qui vérifie si un carré de nombres est magique ou non. On représente un carré de nombres de dimension n par une liste de n listes contenant chacune n éléments.

```
(magic? '((4 9 2) (3 5 7) (8 1 6))) ⇒ #t
```

```
(magic? '((4 4 7) (5 3 7) (8 1 6))) ⇒ #f
```