

Programmation Fonctionnelle 1

TDMachine 3 - Listes

1) Construction et manipulation de listes :

- a) Construire la liste à 3 éléments contenant les entiers 10, 20 et 30 en utilisant d'une part :: et [] et d'autre part ; [] et [].
- b) Ecrire les fonctions head et tail, qui, étant donnée une liste non vide, retourne respectivement le 1^{er} élément de la liste et la liste privée de son premier élément.
- c) Composer les fonctions head et tail pour accéder, s'il existe, au 3ème élément de chacune des listes suivantes :

[1;2;3;4]	[(1,2);(3,4);(5,6)]
[]	[1;2]
[[1];[2;3;4];[]]	[[1,2];[3,4];[5,6]]

2) Donner le type, s'il existe, des expressions suivantes :

- a) []
- b) [1;2;true]
- c) [1;(2,true)]
- d) [1,2,3]
- e) [[1,2];[3,4]]
- f) [[1,2];[3,4,5]]
- g) [1;2;3]
- h) [(1,true,5.0);(2,false,6.4);(3,true,7.9)]
- i) ([1;2;3],[[];[true,false]])

3) Fonctions simples manipulant des listes :

- a) Ecrire la fonction consCpleDouble qui, étant donné un entier n, construit la liste des couples (i,2i) pour i = 1 à n.
- b) Ecrire la fonction consCpleF qui, étant donné une fonction f et un entier n, construit la liste des couples (i, f i) pour i = 1 à n.

4) Représentation d'un ensemble par une liste sans répétition :

On représente un ensemble par la liste sans répétition de ses éléments. Ainsi $\{1,2,3\}$ sera représenté par une liste contenant, une seule fois, les éléments 1, 2 et 3, dans un ordre quelconque.

- a) Ecrire la fonction `appartient` qui, étant donnés un élément et une liste, indique si l'élément appartient à la liste.
- b) Ecrire la fonction `consEns` qui, étant donnée une liste, construit l'ensemble des éléments de la liste.

A partir de maintenant, toutes les listes manipulées représenteront des ensembles.

- c) Ecrire la fonction `cardinal` qui, étant donnée un ensemble, calcule le nombre d'éléments de cet ensemble.
- d) Ecrire la fonction `union` qui, étant donnés deux ensembles, réalise leur union.
- e) Ecrire la fonction `inter` qui, étant donnés deux ensembles, réalise leur intersection.
- f) Ecrire la fonction `inclus` qui, étant donnés deux ensembles, indique si le premier est inclus dans le second.
- g) Ecrire la fonction `egalite` qui, étant donnés deux ensembles, indique si les deux ensembles sont égaux.
- h) Ecrire la fonction `prodCart` qui étant donnés deux ensembles réalise leur produit cartésien.

Rappel : Soient deux ensembles $A = \{a1,a2,...,an\}$ et $B = \{b1,b2,...,bk\}$

Le produit cartésien de A par B est l'ensemble

$\{(a1,b1),...,(a1,bk),(a2,b1),...,(a2,bk),...,(an,b1),...,(an,bk)\}$

- i) Ecrire la fonction `ensSousEns` qui, étant donné un ensemble, construit l'ensemble de tous ses sous ensembles.