

Programmation fonctionnelle - TD - séance 2

Exercice 1 (Listes)

1. Ecrire une fonction qui compte le nombre d'éléments d'une liste
2. Ecrire une fonction qui compte le nombre d'éléments pairs d'une liste d'entiers. Ecrire le type de la fonction.
3. Ecrire la fonction `somme` qui fait la somme des nombres d'une liste de flottants. Ecrire le type de la fonction.
4. Ecrire une fonction `min` qui retourne le plus petit entier d'une liste d'entiers. Peut on l'utiliser avec une liste de chaînes de caractères ? Ecrire le type de la fonction.
5. Ecrire la fonction `intervalle` : `intervalle n m` retourne la liste des entiers compris entre `n` et `m` inclus. Si `n` est strictement supérieur à `m`, le résultat calculé est la liste vide. Ecrire le type de la fonction
Généraliser cette fonction en introduisant un pas (supposé positif non nul). Ecrire le type de la fonction
6. Ecrire une fonction `pairs` qui extrait d'une liste d'entiers la liste des entiers pairs.
7. Ecrire les fonctions `appartient` et `place` : `appartient` teste l'appartenance d'un élément à une liste et `place` donne la position d'un élément dans une liste (0 si pas dans la liste, $n > 0$ si l'élément est le n ème). Ecrire le type - le plus général - de la fonction
8. Démontrer mathématiquement que les fonctions `longueur` et `pairs` vérifient la propriété suivante : la longueur de la liste des entiers pairs d'une liste (calculée par `pairs l`) a une longueur inférieure ou égale à celle de `l`. Vous commencerez par écrire formellement la formule à démontrer puis vous la démontrerez en utilisant le principe d'induction structurale sur les listes. Refaire ensuite la démonstration en faisant une récurrence sur la longueur des listes.

Exercice 2 (Des ensembles)

On représente un ensemble par une liste d'éléments sans doublons (la liste ne contient pas deux fois le même élément).

1. Ecrire une fonction qui ajoute un élément à un ensemble représenté par une liste sans doublons. Ecrire le type - le plus général - de cette fonction.
2. Ecrire une fonction `list2set` qui transforme une liste en un ensemble, i.e qu'elle élimine les doublons d'une liste. Ecrire le type de la fonction `list2set`.
3. Ecrire la fonction qui calcule l'intersection de deux ensembles représentés par des listes sans doublons.
4. Ecrire la fonction qui calcule l'union de deux ensembles représentés par des listes sans doublons.

Exercice 3 (Rationnels)

Le rationnel $\frac{p}{q}$ sera représenté par le couple (p, q) de type `int*int`.

1. Ecrire la fonction `inverse_ratio` qui calcule l'inverse d'un nombre rationnel.
2. Ecrire la fonction qui réalise l'addition de 2 nombres rationnels. On ne cherchera pas à simplifier le rationnel résultat.
3. Ecrire une fonction qui inverse tous les rationnels d'une liste.
4. Ecrire une fonction fait la somme de tous les rationnels d'une liste.