

Devoir surveillé n°9 – NSI
17/05/2024

Toutes les fonctions définies dans ce contrôle doivent l'être dans le langage Python et en respectant le paradigme de la programmation fonctionnelle.

Autrement dit : Il est interdit d'utiliser les instructions d'affectation, mais également les boucles `while` ou `for` (y compris en utilisant les définitions de liste par compréhension), sauf mention du contraire.

Exercice 1 (4 points)

Dans chaque cas ci-dessous, définir une fonction `carre(l)` qui prend en argument une liste `l` et qui retourne une liste dont les éléments sont les carrés de ceux de `l`, en respectant dans chaque cas la contrainte donnée :

1. La fonction doit être récursive et ne pas utiliser la fonction `map`.
2. La fonction doit utiliser la fonction `map`.

Exercice 2 (4 points)

Dans chaque cas ci-dessous, définir une fonction `fun(liste,n)` qui prend en arguments une liste d'entiers `liste` et un entier `n` et qui retourne la liste des quotients de la division des éléments de `liste` par `n` seulement dans le cas où les éléments ne sont pas divisibles par `n`.

Par exemple :

```
assert fun([2,5,7,11,13,15,17],5) == [0,1,2,2,3]
```

En effet, 5 et 15 sont divisibles par 5; on ne renvoie que les quotients des autres nombres par 5.

1. La fonction doit utiliser les fonctions `map` et `filter`.
2. La fonction doit utiliser une définition de liste par compréhension.
(le code n'est donc pas fonctionnel dans ce cas)

Exercice 3 (2 points)

Définir une fonction `nbpairs(l)` utilisant la fonction `reduce` qui calcule le nombre d'entiers pairs de la liste d'entiers `l` donnée en argument.