

## Regroupement septembre 2021

### Programmation fonctionnelle

#### Epreuve 1

Dans les différents exercices, vous pouvez, si besoin est, écrire des sous-fonctions pour vous simplifier le travail.

1. À partir d'une liste associant des noms de ville, un nombre de gens à vacciner et un nombre de vaccins disponibles :

villes => ((ville1 population vaccins) (villeN population vaccins) ... (villeN population vaccins))

Faire une fonction qui construit une liste des villes ayant un nombre de vaccins inférieur à sa population ou ayant un nombre supérieur, en fonction d'un argument :

(vaccins villes 'insuffisant') => ramène la liste des villes n'ayant pas assez de vaccins.

(vaccins villes 'suffisant') => ramène la liste des villes ayant assez ou trop de vaccins

2. Faire une fonction qui calcule le nombre total de vaccins disponibles dans l'ensemble des villes

#### Epreuve 2

A partir d'un arbre (liste quelconque), supprimer les branches contenant (au premier niveau) l'atome nil, à l'aide d'une fonction chirurgicale récursive.

Ex. (a (b c) nil d) est une branche qui contient l'atome nil ; mais (a (b c) (nil) d) est une branche qui ne contient pas, au premier niveau, l'atome nil. On va donc supprimer la première branche, mais dans la deuxième, on supprimera la sous-branche (nil).

Astuce 1 : Utilisez une fonction auxiliaire qui teste si une branche contient l'atome nil.

Astuce 2 : Commencez par écrire une fonction non chirurgicale (qui construit sans modifier).