

Fonctions comme données

Objectifs

- Comprendre l'intérêt de considérer les fonctions comme des données.

Exercice 1 Dans cet exercice, on souhaite produire une nouvelle liste à partir d'une liste existante. Après avoir expérimenté sur quelques cas particuliers, nous mettons en œuvre une solution plus générale.

On utilisera les fichiers `map.py` et `test_map.py`.

- 1.1.** Écrire une fonction qui, pour une liste d'entiers donnée, retourne une nouvelle liste qui contient tous les éléments de la première mis au carré.
- 1.2.** Écrire une fonction qui, pour une liste d'entiers donnée, retourne une nouvelle liste qui contient tous les éléments de la première divisés (division entière) par 2.
- 1.3.** Écrire une fonction qui, pour une liste d'entiers donnée, retourne une nouvelle liste qui contient vrai si l'élément correspondant de la première liste est pair, faux sinon.
- 1.4.** On constate que la structure du code des fonctions précédentes est la même. Proposer une fonction qui permet de produire facilement les résultats précédents. On l'appellera **map**.

Exercice 2 : Menus textuels

Nous avons eu l'occasion de définir plusieurs interfaces utilisateurs (IHM) qui s'appuient sur un menu textuel. Prenons du recul sur ce que nous avons fait.

- 2.1.** Comparer ces différentes IHM et indiquer ce qui pourrait être réutilisé entre ces applications pour faciliter l'écriture de la partie menu textuel de l'IHM.
- 2.2.** Définir un module *menus* (fichier `menus.py`) qui permet de définir un système de menus textuels réutilisables. On commencera par décrire ce qu'est un menu. On pourra partir d'un menu particulier. Bien sûr, on l'appliquera sur l'une des applications écrites. On complètera `devin_menu.py`.
- 2.3.** Il peut être pratique de sélectionner une entrée d'un menu avec un raccourci (une chaîne de caractères) qui est plus facile à mémoriser qu'un numéro. Ajouter cette fonctionnalité.