

TP de Python numéro 3 - Annexe

Semaine du 7 novembre.

Utilisations de boucles imbriquées

Remarque. Il peut être nécessaire d'avoir recours à des boucles imbriquées pour certains algorithmes.

Exercice 1. 1. Écrire le code d'une fonction Python `ListeRangs` prenant en entrée une liste de nombres `L` et renvoyant en sortie la liste `G` de longueur `len(L)` donnée par la contrainte suivante : pour tout $i \in \llbracket 0, \text{len}(L) - 1 \rrbracket$, `G[i]` est le nombre d'éléments de `L` strictement inférieurs à `L[i]`. *On utilisera deux boucles imbriquées.*

2. (bonus) Reprendre la question précédente sans utiliser de boucles imbriquées.

Exercice 2. Écrire le code d'une fonction Python `Deux PlusProches` prenant en entrée une liste de nombres `L` de longueur au moins 2 et renvoyant en sortie les deux valeurs les plus proches présentes dans `L`.

Exercice 3. Écrire le code d'une fonction Python d'entête `def Trouve_sommes(L, but)` prenant en entrée une liste de nombres `L` et un nombre `but` et renvoyant en sortie la liste constituée des couples `(a, b)` d'éléments de `L` tels que `a+b=but`. *Chaque couple possible devra être présent une seule fois, et l'ordre ne sera pas pris en compte : la liste renvoyée ne devra pas contenir à la fois un couple `(a, b)` et le couple `(b, a)`.*