

Programmation en Python

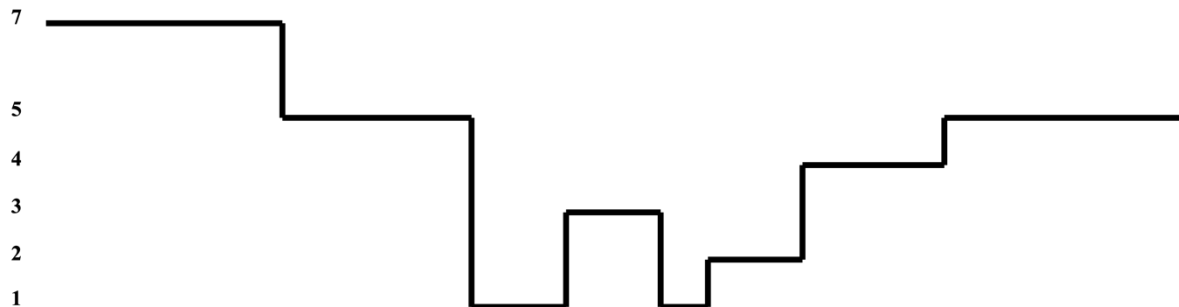
Exercice 1 :

On se donne une liste non vide d'entiers (on pourra l'assimiler à une suite d'altitudes). On appelle **plateau** toute sous-séquence d'éléments successifs ayant la même valeur (à la même altitude).

Exemple :

Tab = [7,7,7,7,7,5,5,5,5,1,1,3,3,1,2,2,4,4,4,5,5,5,5]

Tab est un tableau comportant 8 plateaux (cf figure ci-dessous). Le premier vaut 7 et a pour longueur 5 le cinquième vaut 1 et a pour longueur 1.



On considère une séquence d'entiers non vide représentée dans une liste.

1. Écrire une fonction python **premier_plateau** qui permet de trouver et de retourner la hauteur et la longueur du premier plateau de la séquence. Dans l'exemple ci-dessus, le résultat retourné est : **7 et 5**.
2. Écrire une fonction python **nb_plateaux** qui détermine le nombre de plateaux d'une liste de nombres (correspondant à une liste des altitudes) passée en argument.
3. Écrire une fonction python **plus_long** qui détermine la longueur du plus grand plateau d'une liste de nombres passée en argument.
4. Écrire une fonction python **nombre_max** qui permet de trouver et de retourner, pour une séquence donnée, la longueur maximale des plateaux et le nombre de plateaux de longueur maximale. Dans l'exemple ci-dessus, le résultat affiché est :
 - longueur maximale d'un plateau : 5
 - nombre de plateaux de longueur maximale : 2

On peut décrire une liste de valeurs (des altitudes) de manière compressée à l'aide de ses plateaux. La liste compressée est alors une liste de listes à deux éléments [valeur, longueur du plateau] qui permet de décrire la succession des plateaux. Dans le cas de **Tab**, la liste compressée correspondante est :

Tab_comp = [[7,5],[5,4],[1,2],[3,2],[1,1],[2,1],[4,3],[5,5]].

Cette liste compressée contient elle-même des listes : par exemple, **Tab_comp[0]** est la liste **[7,5]** et **Tab_comp[0][0]** donne **7**.

5. Écrire une fonction python **coder** qui reçoit en argument une liste de valeurs **Tab** et retourne la liste compressée correspondante.
6. Écrire une fonction python **décoder** qui reçoit en argument une liste **description**, sous la forme d'une liste de listes à deux éléments **[valeur,longueurdu_plateau]**, et retourne la liste de valeurs décrite par la liste compressée **description**.

Par exemple, **décoder (Tab_comp)** renverra la liste **Tab**.

Exercice 2 :

On considère n éléments rangés dans un tableau T ($n \geq 1$). Chaque élément ne peut prendre que l'une des trois valeurs : blanc, bleu ou rouge. On souhaite trier le tableau de sorte qu'on ait à gauche tous les éléments bleus (s'il y en a), puis au milieu tous les éléments blancs (s'il y en a), et à droite tous les éléments rouges (s'il y en a).

Écrire une fonction python **Trier** qui reçoit en argument le tableau **T** et retourne le tableau **T** trié de la manière suivante :

- à gauche tous les éléments bleus (s'il y en a) ;
- au milieu tous les éléments blancs (s'il y en a),
- à droite tous les éléments rouges (s'il y en a)

Remarque : le tableau T doit être parcouru une seule fois