

TP2 Dictionnaires

1 Dictionnaires

Exercice 1.1. *Le Scrabble est un jeu de lettres où l'on forme des mots et l'on marque des points en faisant la somme des valeurs de chaque lettre composant le mot.*

Dans la version française, les points sont les suivants :

- 1 points pour A, E, I, L, N, O, R, S, T et U;
- 2 points pour D, G et M;
- 3 points pour B, C et P;
- 4 points pour F, H et V;
- 8 points pour J et Q;
- 10 points pour K, W, X, Y, Z.

1. Donner la commande permettant de créer un dictionnaire dont les clés sont les lettres (type `str`) et les valeurs sont les points correspondant suivant le barème ci-dessus.
2. Écrire une fonction `score(mot)` qui prend en argument une chaîne de caractères `mot` (en majuscules sans accent) et qui renvoie la valeur de ce mot.

On utilisera bien sûr le dictionnaire défini dans la question précédente.

Exercice 1.2. *Le tiroir des chaussettes est sens dessus dessous : elles ne sont pas appariées et il en manque peut-être certaines pour faire des paires. On représente les chaussettes dans une liste de chaînes de caractères représentant leur couleur, par exemple :*

`tiroir = ['blanche', 'noire', 'noire', 'bleue', 'noire', 'blanche']`

*Écrire une fonction `paires(tiroir)` qui renvoie le nombre de paires de chaussettes de même couleur que l'on peut former.
qui renvoie le nombre de paires de chaussettes de même couleur que l'on peut former.*

Par exemple, avec la composition du tiroir donnée précédemment, le résultat sera 2.

En effet : une paire noire et une paire blanche, les deux restantes ne s'apparient pas.

2 Fonction de hachage

Exercice 2.1. Des chaînes vers les entiers

On propose ici un exemple simple de pré-traitement des chaînes de caractères avant hachage.

Nous allons nous servir du codage ASCII qui est donné sous Python par la fonction `ord(c)` qui renvoie le code ASCII sur 8 bits du caractère c .

1. Écrire une fonction **chaîneversentier(ch)** qui prend une chaîne de caractères en argument et renvoie la valeur :

$$\sum_{k=0}^{n-1} \text{ord}(ch[k])256^k.$$

si n désigne la longueur de la chaîne de caractères.

Remarque : on pourra penser à la méthode de Hörner pour en réduire la complexité.

Cette fonction renvoie des entiers de grande taille, par exemple :

chaîneentiers ('exemple de taille modérée')

renvoie :

639711425003567419095171883946955047862310670182554716698725.

2. On souhaite réserver une table de hachage de longueur m , la fonction de hachage étant $h : p \rightarrow p \% m$ (reste de la division euclidienne). Comment peut-on réécrire la fonction pour qu'elle réalise le pré-traitement et le hachage proposé à moindre coût ?

Exercice 2.2. Multiplication

Une méthode consiste à construire des fonctions de hachage de la façon suivante :

- on se donne un réel (ou un flottant) $\theta \in]0, 1 [$ et un entier $m > 0$.
- on définit $h : e \in \mathbb{N}^* \rightarrow \lfloor m \times ((e \times \theta) \bmod 1) \rfloor$ (où $x \bmod 1 = x - \lfloor x \rfloor$).

On se propose d'en explorer quelques propriétés.

1. Quelle sera la taille d'une table de hachage associée à une fonction multiplicative de cette forme ?
2. On décide de tester la répartition des clés dans la table en fonction de θ .
 - (a) Définir une fonction **hachage(m, theta, c)** qui prend en arguments m un entier strictement positif, $0 < \theta < 1$ un flottant, c entier et qui réalise le hachage de c .
 - (b) Écrire une fonction **testhachage(m, theta)** qui renvoie pour chaque entier i compris entre 0 et $m - 1$ le nombre de valeurs $c \in \llbracket 0, 10^5 \rrbracket$ telles que $h(c) = i$, soit :

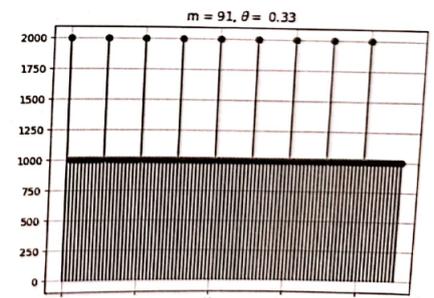
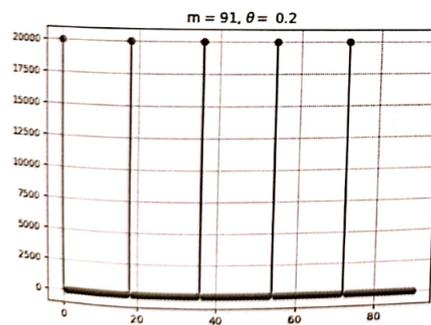
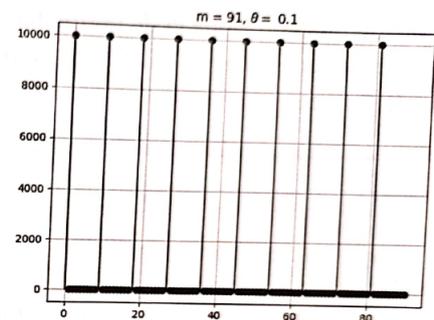
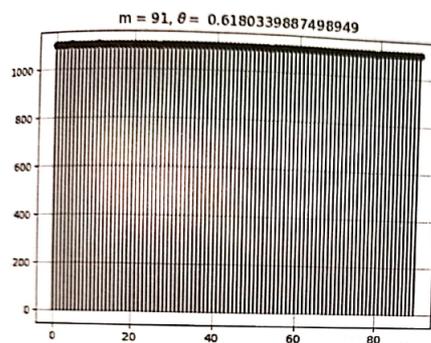
$$d_i = |\{c \in [0, 10^5] / h(m, \theta, c) = i\}|.$$

Ces valeurs seront stockées dans un dictionnaire que renverra la fonction.

- (c) Les quatre histogrammes ci-dessous représentent les répartitions obtenues (i en abscisse, et d_i en ordonnée) avec :

$$m = 91 \text{ et } \theta \in \left\{ \frac{\sqrt{5}-1}{2}, 0.1, 0.2, 0.33 \right\}.$$

Expliquer les résultats lorsque $\theta = 0.1, 0.2$.



- (d) Justifier que $c \rightarrow h(m, a/b, c)$ prend au plus $\max(m, b)$ valeurs. Est-ce cohérent avec ce que l'on observe ?
- (e) Conseil : Il pourra être intéressant d'observer la répartition des collisions lorsque $\theta = 0.33$ et pour différentes valeurs de m (plus petites, plus grandes que 100).