

1. Vitesse de recherche

Compléter le tableau suivant en chronométrant le temps mis par la machine pour exécuter la boucle suivante en fonction de n selon qu'on travaille sur une liste ($a = [i \text{ for } i \text{ in } \text{range}(n)]$) ou sur un ensemble ($a = \{i \text{ for } i \text{ in } \text{range}(n)\}$) :

<code>for i in range(10**6) :</code>		n = 10	n = 100	n = 1000
<code> if i in a :</code>	list			
<code> pass</code>	set			

Remarque : On pourra utiliser un chronomètre ou importer la fonction `time()` du module `time` qui permet de connaître le nombre de secondes écoulées depuis le 1^{er} janvier 1970 à 00h00.

2. maxlettre(mot)

Créer la fonction `maxlettre(mot)` qui renvoie l'ensemble des lettres les plus présentes dans le mot.

Exemples : `maxlettre("bonjour")` renvoie {"o"}
`maxlettre("didrit")` renvoie {"d", "i"}

3. Scrabble

Voici les valeurs des lettres au Scrabble :

- 1 point pour "a", "e", "i", "l", "n", "o", "r", "s", "t", "u"
- 2 points pour "d", "g", "m"
- 3 points pour "b", "c", "p"
- 4 points pour "f", "h", "v"
- 8 points pour "j", "q"
- 10 points pour "k", "w", "x", "y", "z"

Créer un dictionnaire donnant la valeur de chaque lettre.

Créer une fonction `score(mot)` calculant la valeur d'un mot.

Exemple : `score("bonjour")` renvoie 16

4. Statistiques.

Créer la fonction `statistiques(omega, n)` qui renvoie, sous forme d'un dictionnaire, les fréquences de chaque valeur dans la répétition de n tirages aléatoires sur un univers oméga donné.

Exemple : `statistiques(["pile", "face"], 100)`
 peut renvoyer {'face': 0.41, 'pile': 0.59}

5. Carré magique

Créer la fonction `magique(grille)` qui vérifie si une grille représente un carré magique, c'est-à-dire que les sommes des nombres de chaque ligne, de chaque colonne et des deux diagonales sont les mêmes. Exemple : grille = [[2, 7, 6], [9, 5, 1], [4, 3, 8]]

6. Polygônes.

Un dictionnaire nommé `points` contient les noms de points du plan. Créer la fonction `dessiner(polygone)` qui dessine un polygone dans un repère à l'aide des coordonnées de ses sommets.

Ne pas oublier d'écrire le nom des sommets sur le graphique.

Exemple : triangle = "ABC", rectangle = "DEFG"
`points = {"A" : (-3, -1), "B" : (1, 1), "C" : (-1, 4), "D" : (-1, -1), "E" : (3, 1), "F" : (4, -1), "G" : (0, -3)}`

7. Pixel Art.

Représenter le tableau suivant par une liste de listes.

0	0	0	1	0	0	1	0	0	0
0	0	1	1	1	1	1	1	0	0
0	1	1	0	1	1	0	1	1	0
1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	0	1
1	0	0	1	0	0	1	0	0	1
0	0	1	1	0	0	1	1	0	0

Créer la fonction :

`dessiner(grille, x, y, n)`

qui trace un dessin en Pixel Art

à partir d'une grille de pixels représentée par une liste de liste.

(0 pour les carrés blancs et 1 pour les carrés noirs).

On indiquera les coordonnées (x, y) du coin supérieur gauche et la taille n d'un pixel en paramètres.

Astuces : `t.shape("square")` pour avoir une forme carrée
`t.shapesize(n/25, n/25)` pour adapter sa taille
`t.stamp()` pour la dessiner aux coordonnées courantes