TP 10. Dictionnaires.

🌈

Noé - MPSI 2

Exercice 1.

```python
def compter(S, v):
  occurences = 0
  for i in range(len(S)):
    if S[i] == v:
      occurences += 1
  return occurences

print(compter([1, 1, 3], 1))
print(compter("abaa", "b"))

def occurences(S):
  dict1 = {}
  for elt in S:
    dict1[elt] = compter(S, elt)
  return dict1
```

Exercice 2.

```python
def antecedent(D, v):
  ante = []
  for clef, valeur in D.items():
    if valeur == 5:
      ante.append(clef)
  return ante

D = {"A":5, "B":5, "C":6}
print(antecedent(D, 5))
```

Exercice 3.

```python
scrabble = {}

def scrabble_score(mot):
  occ = occurences(mot)
  score = 0
  for clef1, valeur1 in occ.items():
    for clef2, valeur2 in scrabble.items():
      if clef1 == clef2:
        score += valeur1 * valeur2
  return score
```

Exercice 4.

Question 1.

```python
from random import choice
from random import randint

listeTest = []
for i in range(100, 5001, 100):
  listeTest.append([choice([True, False]) for k in range(i)])
```

Question 2.

```python
liste_dic = []
for n in range(100, 50001, 100):
  dictio = {randint(0, 10000) : True for i in range(n)}
  liste_dic.append(dictio)

print(liste_dic)
```

Question 3.

```python
import time
import matplotlib.pyplot as plt

Temps = []
for n in range(100):
  dictionnaire = liste_dic[n]
  T0 = time.perf_counter()
  for k in range(0, 10000):
    k in dictionnaire
  T1 = time.perf_counter()
  Temps.append(T1 - T0)

n = [100*i for i in range(100)]

plt.plot(n, Temps)
plt.ylabel("Temps (s)")
plt.xlabel("n")
plt.title("k in D = f(n)")
plt.show()
```

Exercice 5.

Question 1.

```python
def minliste(i, L):
  indice = i
  min = float("inf")
  for j in range(i+1, len(L)):
    if abs(L[i] - L[j]) < min:
      indice = j
      min = abs(L[i] - L[j])
  return (indice, min)

def precondition(L):
  dictio = dict()
  for i in range(len(L)-1):
    dictio[i] = minliste(i, L)
  return dictio
```

Question 2.

On passe n fois dans la boucle for de `precondition`. Puis lorsque l'on appelle `minliste`, on parcourt $n - i$ fois la boucle for.

Donc c'est comme si on sommait sur un triangle lorsque l'on appelle `minliste` dans `precondition`.

Donc le nombre de passages total est :

$$n + \frac{n(n+1)}{2}$$

Question 3.

```python
def retour_minimal(L):
  dico = precondition(L)
  min = float("inf")
  clef_m = 0
  for clef, value in dico.items():
    print(value[1])
    if value[1] < min:
      print("Ok !")
      min = value[1]
      clef_m = clef
  return (clef_m, dico[clef_m][0], dico[clef_m][1])

L = [143, 272, 988, 602, 313, 740, 121, 146, 23, 222]
print(retour_minimal(L))
```

On obtient en sortie :

> 🌈
>
> [Out] : (0, 7, 3)

C'est bien ce que l'on attend.

Exercice 6.

```python
def coeffBin(n, k):
  if n == 0:
    return 0
  if k == 0:
    return n
  return coeffBin(n-1, k-1) + coeffBin(n-1, k)
```