

Info201 - TD3

DICTIONNAIRES

- On accède à une case *existante* d'un dictionnaire avec `D[k]` où `k` est la *clé*.
- Lorsqu'on n'est pas sûr que la case existe, il faut plutôt utiliser la méthode `D.get(k, e)`: si la case `k` existe, le résultat sera le même que pour `D[k]`, et sinon, le résultat sera `e`.
- On peut obtenir la liste des clés d'un dictionnaire avec `D.keys()`.
- On peut parcourir les cases d'un dictionnaire avec

```
for k in D.keys():
    ...
    ... D[k] ...
    ...
```

Exercice 1. On considère un dictionnaire où les cases ont des valeurs numériques, comme

```
>>> D = {"une case": 17,
        "une deuxième case": -3,
        "une autre case": 0,
        "ceci n'est pas une case": 66}
```

1. Écrivez une fonction qui calcule la somme de toutes les cases.
2. Écrivez une fonction qui calcule le maximum de toutes les cases.
3. Écrivez une fonction qui renvoie une clé contenant une case de valeur minimale.
4. Écrivez une fonction qui renvoie `True` ou `False` suivant que le nombre `x` apparaît ou non dans un dictionnaire.

Exercice 2. On considère une liste de dictionnaire, où chacun des dictionnaires contient 4 cases :

- une case "nom" et une case "prénom",
- une case "année de naissance",
- une case "no_etudiant"

1. Écrivez une fonction qui calcule la moyenne des âges d'une telle liste.
2. Écrivez une fonction qui renvoie le n° d'étudiant d'un étudiant d'âge minimal.

Exercice 3. Écrivez une fonction qui permet de faire la *réunion* de deux dictionnaires. Si D1 et D2 sont deux dictionnaires, `union(D1, D2)` sera un dictionnaire qui contient les cases de D1 et les cases de D2.

Lorsqu'une clé apparaît à la fois dans D1 et dans D2, la valeur de la case correspondante dans `union(D1, D2)` sera la valeur de la case de D1.

Exercice 4. Écrivez une fonction qui permet de faire l'*intersection* de deux dictionnaires numériques. Si D1 et D2 sont deux dictionnaires, `inter(D1, D2)` sera un dictionnaire qui contient les clés qui sont à la fois dans D1 et dans D2.

La valeur d'une telle case sera le produit des deux cases correspondantes dans D et D2.

CHAINES

Exercice 5. Écrivez une fonction `est_palindrome` qui vérifie si une chaîne est un palindrome *sans prendre la ponctuation, les majuscules ou les espaces en compte*. Par exemple, votre fonction devra répondre True sur les chaînes "Was it a car or a cat I saw?", "Le sel..." ou "Isä, älä myy myymälääsi."

Exercice 6. Écrivez une procédure `multiplications(n)` qui affiche la table de multiplication jusqu'à $n*n$. L'affichage devra ressembler à

```
>>> multiplication(10)
1  2  3  4  5  6  7  8  9
2  4  6  8 10 12 14 16 18
3  6  9 12 15 18 21 24 27
4  8 12 16 20 24 28 32 36
5 10 15 20 25 30 35 40 45
6 12 18 24 30 36 42 48 54
7 14 21 28 35 42 49 56 63
8 16 24 32 40 48 56 64 72
9 18 27 36 45 54 63 72 81
```

(Notez l'alignement des colonnes...)

Exercice 7. Écrivez une procédure `puissances(n)` qui affiche la table des puissances jusqu'à $n**n$. L'affichage devra ressembler à

```
>>> puissances(7)
1  1  1  1  1  1  1
2  4  8 16 32 64 128
3  9 27 81 243 729 2187
4 16 64 256 1024 4096 16384
5 25 125 625 3125 15625 78125
```

```
6 36 216 1296 7776 46656 279936
7 49 343 2401 16807 117649 823543
```

(Notez l'alignement des colonnes...)

FICHIERS

- Pour ouvrir un fichier en lecture, on utilise `F = open(nom_fichier, mode="r")`,
- Pour ouvrir un fichier en écriture, on utilise `F = open(nom_fichier, mode="w")`.
- Pour fermer un fichier, on utilise `F.close()`.
- Pour récupérer une nouvelle ligne dans un fichier (ouvert en lecture), on utilise la méthode `l = F.readline()`. Attention, s'il n'y a plus de ligne dans le fichier, le résultat sera la chaîne vide `""`. Dans tous les autres cas, le résultat contiendra le caractère `"\n"` pour le retour charriot.
- Pour écrire une chaîne dans un fichier (ouvert en écriture), on utilise la méthode `F.write(s)`. Attention, cette méthode n'ajoute pas de retour charriot implicitement.

Exercice 8. Écrivez une procédure qui affiche la ligne la plus longue d'un fichier dont le nom est passé en argument.

Exercice 9. Écrivez une procédure qui ajoute les numéros de ligne dans un fichier en créant un nouveau fichier avec l'extension `".no"`.

Si le fichier `test.txt` contient

```
    Lorem ipsum dolor sit amet, consectetur
    adipiscing elit, sed do eiusmod tempor
    incididunt ut labore et dolore magna aliqua. Ut
    enim ad minim veniam, quis nostrud exercitation
    ullamco laboris nisi ut aliquip ex ea commodo
    consequat.
```

alors un appel à `no_lignes("test.txt")` créera un nouveau fichier `test.txt.no` dont le contenu sera

```
001 - Lorem ipsum dolor sit amet, consectetur
002 - adipiscing elit, sed do eiusmod tempor
003 - incididunt ut labore et dolore magna aliqua. Ut
004 - enim ad minim veniam, quis nostrud exercitation
005 - ullamco laboris nisi ut aliquip ex ea commodo
006 - consequat.
```

Exercice 10. Écrivez une fonction qui prend un nom de fichier et un caractère en argument et qui compte le nombre d'occurrence de ce caractère à l'intérieur du fichier.

Exercice 11. Écrivez une fonction qui prend un nom de fichier et qui renvoie un *dictionnaire* où les clés sont les caractères apparaissant dans le fichier et les valeurs sont le nombre d'occurrences de ces caractères dans le fichier.

Par exemple, sur le fichier `test.txt` précédent, le résultat sera:

```
{
  'l': 10, ',': 3, 'r': 9, 'u': 13, 'n': 12, ' ': 30,
  'd': 11, 'g': 2, 'x': 2, 'p': 4, 'q': 4, '\n': 6,
  't': 15, 'i': 24, 'v': 1, 'o': 18, 'c': 8, 'm': 13,
  'e': 18, 's': 11, 'a': 15, '.': 2, 'b': 2
}
```

La case 'l' contient 10 car le fichier contient effectivement 10 occurrences du caractère l, en minuscule ou en majuscule.

Exercice 12. On dispose d'un fichier contenant des mots par ordre alphabétique avec leur définition. Chaque mot défini est seul sur une ligne et les lignes suivantes sont la définition correspondante. La fin de la définition est notée par un symbole % seul sur une ligne. Par exemple :

ABAQUE

- 1- Terme d'architecture. Tailloir, partie supérieure du chapiteau des colonnes, sur laquelle porte l'architrave.
- 2- Terme d'antiquité. Tableau couvert de poussière, sur lequel on traçait des nombres et on enseignait le calcul ; et aussi sorte de carré long, évidé, qui était muni de boules passées dans des fils tendus et qui servait à compter.

%

ABBAYE

- 1- Monastère d'hommes ou de filles.
- 2- Le bénéfice attaché au titre d'abbé.
- 3- Les bâtiments du monastère

%

ABEILLE

Insecte qui produit le miel et la cire, et qui appartient au genre des insectes hyménoptères.

%

...

Écrivez une procédure `affiche_definition(nom_fichier, mot)` qui affiche la définition d'un mot donné en argument en la cherchant dans le fichier.