

# IngeSUP - TD 10 - Les structures de données 2

“Le langage structure tout de la relation inter-humaine.”

Jacques Lacan

## Exercice 10.1

Soit le dictionnaire :

```
d = {'nom': 'Dupuis', 'prenom': 'Jacque', 'age': 30}
```

1. Corriger l'erreur dans le prénom, la bonne valeur est 'Jacques'.

Entrée [ ]:

```
# VOTRE CODE ICI
```

2. Afficher la liste des clés du dictionnaire.

Entrée [ ]:

```
# VOTRE CODE ICI
```

3. Afficher la liste des valeurs du dictionnaire.

Entrée [ ]:

```
# VOTRE CODE ICI
```

4. Afficher la liste des paires clé/valeur du dictionnaire.

Entrée [ ]:

```
# VOTRE CODE ICI
```

5. Ecrire la phrase "Jacques Dupuis a 30 ans" en utilisant `print` sur les clés du dictionnaire `d`.

Entrée [ ]:

```
# VOTRE CODE ICI
```

## Exercice 10.2

On représente des matrices carrées sous la forme de listes de listes de nombres.

Entrée [ ]:

```
carre3 = [
    [2, 7, 3],
    [9, 5, 1],
    [4, 3, 8]
]

carre4 = [
    [4, 5, 11, 14],
    [15, 10, 8, 1],
    [6, 3, 13, 12],
    [9, 16, 2, 7]
]
```

1. Quelle est la valeur de la longueur de *carre4* ?

Réponse:

2. Supposons qu'on veuille accéder à la deuxième valeur de la deuxième ligne de *carre3*. Quelle instruction python doit-on taper ?

Entrée [ ]:

```
# VOTRE CODE ICI
```

3. Supposons qu'on veuille accéder à la troisième valeur de la première ligne de *carre3*. Quelle instruction python doit-on taper ?

Entrée [ ]:

```
# VOTRE CODE ICI
```

4. Quelle instruction permet de récupérer la valeur 3 de *carre4* ?

Entrée [ ]:

```
# VOTRE CODE ICI
```

5. On propose le code suivant :

Entrée [ ]:

```
def somme_ligne(carre, n):
    """
    carre est une liste de listes de nombres
    n est un nombre entier
    """
    somme = 0
    for nombre in carre[n]:
        somme = somme + nombre
    return somme
```

Que vaut *somme\_ligne(carre4, 2)* ? À quoi sert cette fonction ?

Réponse:

6. Définissez la fonction *lignes\_magiques* qui prend un carré en paramètre et qui vérifie que les sommes des nombres de chaque ligne sont égales.

Entrée [ ]:

```
# VOTRE CODE ICI
```

---

## Exercice 10.3

---

Les nombres complexes sont décrits par un couple de réels qui définit leur partie réelle et leur partie imaginaire.

On pourra définir un complexe par un tuple[float,float]. Ainsi le nombre complexe  $2 + 3i$  sera représenté par le tuple (2.0, 3.0), le nombre  $i$  par (0.0, 1.0) et un réel  $r$  par ( $r$ , 0.0).

1. Ecrivez les fonctions `partie_relle()` et `partie_imaginaire()` renvoyant respectivement la partie réelle et la partie imaginaire d'un nombre complexe `c` défini comme un tuple.

Entrée [ ]:

# VOTRE CODE ICI

2. Ecrivez la fonction `addition_complexe()` renvoyant la somme de deux nombres complexes passés en argument.

Entrée [ ]:

# VOTRE CODE ICI

On rappelle que le produit de deux nombres complexes  $(a + bi)$  et  $(c + di)$  est donné par

$$(a + bi) * (c + di) = (ac - bd) + (ad + bc)i$$

3. Ecrire une fonction `produit_complexe()` telle que `produit_complexe(c1, c2)` renvoie le produit des nombres complexes `c1` et `c2`.

Entrée [ ]:

# VOTRE CODE ICI

---

## Exercice 10.4

---

On définit un point par un tuple  $(x, y)$  et une liste de points par une liste de tuple **points**.

1. Ecrivez une fonction `distance()` qui détermine la distance entre tous les points distincts de la liste `points` entrée en paramètre. On affichera la distance maximale entre deux points distincts. Testez votre fonction avec `points=[(1,3),(0,1),(2,4)]`.

On rappelle que la distance  $d$  entre un point P1 de coordonnées  $(x_1, y_1)$  et un point P2 de coordonnées  $(x_2, y_2)$  s'obtient par :

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

On ne s'occupera pas de la redondance de certains calculs. (En effet la distance entre les points P1 et P2 est identique à la distance entre les points P2 et P1).

Entrée [ ]:

# VOTRE CODE ICI

---

## Exercice 10.5

---

1. Ecrivez une fonction nommée `compte_car()` qui accepte une chaîne de caractères et qui renvoie l'occurrence des caractères contenus dans la chaîne sous forme de dictionnaire.

Entrée [ ]:

```
def compte_car(chaine):  
    ...  
  
print(compte_car("afgzy,&!!djfijaijddjqokojdijiofjoeijf!"))    # TEST
```

2. Ecrivez une fonction nommée `compte_mots_ligne()` qui accepte une chaîne de caractères et qui renvoie l'occurrence des mots contenus dans la chaîne sous forme de dictionnaire. (Rappel: Les mots sont séparés par le caractère espace).

Entrée [ ]:

```
# VOTRE CODE ICI
```

À l'aide de la fonction précédente, il est maintenant possible de déterminer les mots les plus représentés (hors caractères spéciaux) dans un fichier composé de plusieurs lignes ou dans une variable.

3. Ecrivez une fonction `compte_mots_texte()` qui accepte une variable et retourne l'occurrence des mots contenus dans la variable sous forme de dictionnaire. Testez avec la variable `lafontaine_txt` entrée ci-dessous.

Entrée [ ]:

```
lafontaine_txt = """Rien ne sert de courir ; il faut partir à point.\nLe Lièvre et la Tortue en sont un témoignage.\nGageons, dit celle-ci, que vous n'atteindrez point\nSitôt que moi ce but. - Sitôt ? Etes-vous sage ?\nRepartit l'animal léger.\nMa commère, il vous faut purger\nAvec quatre grains d'ellébore.\n- Sage ou non, je parie encore.\nAinsi fut fait : et de tous deux\nOn mit près du but les enjeux :\nSavoir quoi, ce n'est pas l'affaire,\nNi de quel juge l'on convint.\nNotre Lièvre n'avait que quatre pas à faire ;\nJ'entends de ceux qu'il fait lorsque prêt d'être atteint\nIl s'éloigne des chiens, les renvoie aux Calendes,\nEt leur fait arpenter les landes.\nAyant, dis-je, du temps de reste pour brouter,\nPour dormir, et pour écouter\nD'où vient le vent, il laissa la Tortue\nAller son train de Sénateur.\nElle part, elle s'évertue ;\nElle se hâte avec lenteur.\nLui cependant méprise une telle victoire,\nTient la gageure à peu de gloire,\nCroit qu'il y va de son honneur\nDe partir tard. Il broute, il se repose,\nIl s'amuse à toute autre chose\nQu'à la gageure. A la fin quand il vit\nQue l'autre touchait presque au bout de la carrière,\nIl partit comme un trait ; mais les élans qu'il fit\nFurent vains : la Tortue arriva la première.\nEh bien ! lui cria-t-elle, avais-je pas raison ?\nDe quoi vous sert votre vitesse ?\nMoi, l'emporter ! et que serait-ce\nSi vous portiez une maison ?"""
```

Entrée [ ]:

```
# VOTRE CODE ICI
```

4. Déterminez l'occurrence des mots **Lièvre** et **Tortue** dans la variable `lafontaine_txt`.

Entrée [ ]:

```
# VOTRE CODE ICI
```

## Corrigé du TD 10

Vous pouvez retrouver le corrigé de ce TD [ici](#) ([Corrig%C3%A9s/Corrig%C3%A9\\_TD%2010.ipynb](#)).