

## Représentation des données - Les tuples

### Manipulation de tuples

Un tuple est : Une séquence d'éléments non modifiables.

Pour créer un tuple on utilise des parenthèses

L'accès aux éléments se fait avec leurs indices

```
t=() # tuple vide

t=(4,) #initier un tuple avec une seule valeur

t=t+(2,5,7,8) # concaténation t vaut (4,2,5,7,8)

t[2] #affiche 5

t[2]=3 # Va provoquer une erreur car les tuples sont non modifiables
-----
TypeError                                 Traceback (most recent call last)
<ipython-input-19-8ae953d958b6> in <module>()
----> 1 t[2]=3

TypeError: 'tuple' object does not support item assignment
-----
t=(4,1,5,2,3,9,7,2,8)

t[2:5] # résultat retourné: (5, 2, 3)

t[:5] # résultat retourné: (4, 1, 5, 2, 3)

t[3:] # résultat retourné: (2, 3, 9, 7, 2, 8)

t[:3]+(10,)+t[4:] # résultat retourné: (4, 1, 5, 10, 3, 9, 7, 2, 8)

l=list(range(5)) # l est une liste qui vaut [0, 1, 2, 3, 4]

t=tuple(l) # tuple une fonction qui permet de convertir en tuple,
# t donc est un tuple qui vaut (0, 1, 2, 3, 4)

ch='lepython.org'

t=tuple(ch) # t est un tuple qui vaut ('l', 'e', 'p', 'y', 't', 'h', 'o', 'n', '.', 'o', 'r', 'g')

len(t) # résultat retourné : 12
```

NOM .....

---

*TP*

---

- En Mathématiques la distance entre deux points  $A(x_A, y_A)$  et  $B(x_B, y_B)$  dont on connaît les coordonnées dans un repère orthonormé est donnée par la relation :  $AB = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$

Écrire une fonction `dist_carree(A,B)` qui prend en paramètre deux tuples A et B contenant les coordonnées des points et qui renvoie le carré de la distance AB ( c-a-d:  $AB^2$ )

Recopier votre code ici :

```
def dist_carree(A,B):
    ''' spécifications
    '''

A=( , )
B=( , )
print(dist_carree(A,B))
```

Résultat : .....

- Les coordonnées du milieu de deux points  $A(x_A, y_A)$  et  $B(x_B, y_B)$  dont on connaît les coordonnées dans un repère orthonormé sont:  $\left(\frac{x_A + x_B}{2}, \frac{y_A + y_B}{2}\right)$

Écrire une fonction `milieu(A,B)` qui prend en paramètre deux tuples A et B contenant les coordonnées des points et qui renvoie les coordonnées de leur milieu sous forme d'un tuple.

Recopier votre code ici :

```
def milieu(A,B):
    ''' spécifications
    '''

A=( , )
B=( , )
print(milieu(A,B))
```

Résultat : .....

