

# Algorithmes et Pensée Computationnelle

## Structure de données, Itération et Récursivité

Le but de cette séance est de s'exercer avec des structures de données qui seront utilisées lors des prochaines séances de cours/Travaux Pratiques. Les structures de données abordées lors de cette séance sont les tuples, les listes et les collections. Au terme de cette séance, l'étudiant sera capable de distinguer une structure de données immuable et non immuable, écrire un programme de façon simplifiée en utilisant les notions d'itération et de récursivité.

## 1 Les structures de données

### 1.1 Les tuples

Pour rappel, les tuples sont des listes d'éléments immuables, ce qui signifie que ces listes ne peuvent pas être modifiées. Les tuples sont utiles pour stocker des données que l'on va réutiliser plus tard.

En Python, pour créer un tuple, il suffit de définir une variable et de lui assigner des valeurs entre parenthèses et séparer les valeurs entre elles par des virgules :

#### >\_ Exemple

```
1 mon_tuple = (1,2,3,4)
```

**Question 1:** (🕒 5 minutes) **Manipulation d'un tuple** Créez un tuple nommé `mon_tuple` contenant les chiffres 1,2,3,4 et 5. Stockez le 4ème élément dans une variable `element_4`, puis affichez le contenu de cette variable.

#### 💡 Conseil

Pour accéder à un élément d'un tuple, ou d'une liste, vous pouvez utiliser l'indexation. Comme pour accéder aux caractères des chaînes de caractères, utilisez [ ].

#### >\_ Solution

```
1 mon_tuple = (1,2,3,4,5)
2 print(mon_tuple[3])
```

**Question 2:** (🕒 5 minutes) **Manipulation d'un tuple - 2** Créez un tuple nommé `mon_tuple` contenant les chiffres 1,2,3,4 et 5. Obtenez le nombre d'éléments contenus dans votre tuple et stockez le résultat dans une nouvelle variable nommée `taille_tuple`. Pour finir, affichez le contenu de `taille_tuple`.

#### 💡 Conseil

Pour calculer la taille d'un tuple, ou d'une liste, vous pouvez utiliser la fonction `len()`.

#### >\_ Solution

```
1 mon_tuple = (1,2,3,4,5)
2 taille_tuple = len(mon_tuple)
3 print(taille_tuple)
```