

---

---

# TP n° 7 : les listes chaînées

---

---

## 1. Manipulations de base des listes chaînées

### 1.1. Présentation

Voici une série d'instructions (les instructions ci-dessous s'enchaînent) :

```
1 l = creer_liste()
2 print(est_vide(l)) # Affiche True
3 l = ajout_en_tete(3, l) # La liste l contient l'élément 3
4 print(est_vide(l)) # Affiche False
5 l = ajout_en_tete(5, l) # La tête de la liste l contient l'élément 5,
   la queue contient l'élément 3
6 print(l) # Affiche [5, [3, []]]
7 l = ajout_en_tete(8, l) # La tête de la liste l contient l'élément 8,
   la queue contient les éléments 5 et 3
8 print(l) # Affiche [8, [5, [3, []]]]
9 t, l = supprime_en_tete(l) # La variable t vaut 8, la tête de la liste
   l contient l'élément 5, la queue contient l'élément 3
10 print(l) # Affiche [5, [3, []]]
11 print(t) # Affiche 8
12 l1 = creer_liste()
13 l2 = ajout_en_tete(8, ajout_en_tete(5, ajout_en_tete(3, l1)))
14 print(l2) # Affiche [8, [5, [3, []]]]
```

### 1.2. Exercice

Voici une série d'instructions (les instructions ci-dessous s'enchaînent), expliquez ce qui se passe à chacune des étapes.

```
1 l = creer_liste()
2 l = ajout_en_tete(10, l)
3 l = ajout_en_tete(9, l)
4 l = ajout_en_tete(7, l)
5 l1 = creer_liste()
6 l2 = ajout_en_tete(5, ajout_en_tete(3, ajout_en_tete(2, ajout_en_tete(
   1, ajout_en_tete(0, l1)))))
```

## 2. Implémentation en utilisant des fonctions

### 2.1. Implémentation du TAD liste chaînée

Ouvrir le programme *TP07\_listes\_fonctions* et remplacer toutes les instructions *pass* pour implémenter les instructions élémentaires du TAD.

Vérifier que les résultats des tests sont bien conformes aux prévisions de la partie 1.

### 2.2. Implémentation de fonctions supplémentaires

Ouvrir le programme *TP07\_listes\_fonctions\_supplementaires* et remplacer toutes les instructions *pass* pour implémenter les fonctions supplémentaires du TAD.

Attention, il faudra souvent recourir à la programmation récursive.

## 3. Implémentation en POO

### 3.1. Implémentation minimaliste avec une classe *Maillon*

Ouvrir le programme *TP07\_listes\_classe\_maillon*. Regarder le code et le tester pour en comprendre le fonctionnement.

### 3.2. Implémentation avec une classe *Maillon* et une classe *Liste\_chainee*

Ouvrir le programme *TP07\_listes\_classes*. On utilise cette fois deux classes : une classe *Maillon* et une classe *Liste\_chainee*. Compléter les méthodes proposées.

## Capacités exigibles

- Écrire plusieurs implémentations d'une même structure de données.