

ALG. Listes Chaînées. Exercices

Pour les exercices suivants, on pourra s'appuyer sur la classe *Cellule* du cours avec les méthodes `__init__(self)`, `__str__(self)` et d'autres si souhaitées. On utilisera Jupyter pour les programmes.

Exercice 1 : une liste chaînée d'entiers naturels

- 1/ **Ecrire** une fonction *listN(n)* qui reçoit en argument un entier naturel *n* et renvoie la liste chaînée contenant les entiers 1, 2, ..., 3, 4, *n*.
- 2/ **Faire** un jeu de tests.

Exercice 2 : égalité de deux listes chaînées

- 1/ **Ecrire** une fonction *equal(lst1, lst2)* qui reçoit deux listes chaînées en argument et qui renvoie *True* si elles sont identiques, c'est-à-dire qu'elles contiennent les mêmes éléments dans le même ordre.
- 2/ **Faire** un jeu de tests.

Aide : considérer le cas de listes vides

Exercice 3 : concaténation de deux listes

On considère deux listes chaînées *lst1* et *lst2*. **Concaténer** les listes *lst1* et *lst2*, c'est créer une liste *lst3* contenant tous les éléments de *lst1* puis ceux de *lst2*.

- 1/ **Ecrire** une fonction *concat(lst1, lst2)* qui renvoie une nouvelle liste *lst3* créée par concaténation des listes *lst1* et *lst2*.
- 2/ **Faire** un jeu de tests.
- 3/ **Tester** le petit programme.

```
# Création d'une Liste chaînée
lst2 = Cellule(2, Cellule(3, None))
lst1 = Cellule(1, lst2)
lst3 = Cellule(4, Cellule(5, None))
print(lst1) # Attendu : 1 , 2 , 3
print(lst2) # Attendu : 2 , 3
lst4 = concat(lst1, lst3)
print(lst4) # Attendu : 1 , 2 , 3 , 4 , 5
print(lst1) # Attendu : 1 , 2 , 3 (Surprise !!)
print(lst2) # Attendu : 2 , 3 (Surprise !!)
```

Justifier les valeurs de *lst1* et *lst2* à la fin du script.