

Exercice 1 : Listes chaînées - nouvelles fonctionnalités

Reprendre la classe *Liste* vue en cours et ajouter les méthodes ci-après :

1. Implémenter la méthode `__str__(self) → str` qui renvoie les valeurs de la liste séparées par un espace. Il sera nécessaire d'écrire une méthode supplémentaire *récursive* `afficher__rec(self, maillon : object) → str`.
2. Construire la méthode *impérative* `renverser(self) → None` qui inverse l'ordre de la liste.
3. Construire la méthode `concatener(self, l : object) → object` qui renvoie une nouvelle liste composée de la liste interne de la classe et *l*. Il sera nécessaire d'écrire une méthode supplémentaire *récursive* `concatener__rec(self, tete1 : object, tete2 : object) → object`. Attention les deux listes ne sont pas modifiées mais une nouvelle liste concaténée est renvoyée.
4. Estimer la complexité de la méthode *concatener*.

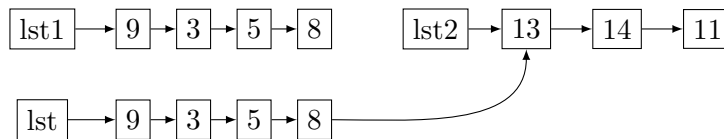


FIGURE 1 – Fonctionnement de `concatener__rec`

Exercice 2 :

1. En utilisant la classe *Liste* précédente, créer la liste triée 1,3,7,9,10.
2. Créer la méthode *récursive* `insérer(self, val : int) → None` qui insère l'élément *val* à la bonne place dans la liste triée.