TD3 : Listes chaînées

1 Maillons

On veut créer une classe Maillon qui contiendra un entier et la référence du maillon suivant. Écrire des méthodes **dynamiques** pour :

1. retourner la valeur contenue dans un maillon,
2. retourner le maillon suivant,
3. modifier la valeur contenue dans le maillon,
4. relier deux maillons.

2 Listes d'entiers

En utilisant la classe Maillon, on veut maintenant écrire une classe Liste pour contenir une suite de maillons reliés les uns aux autres. Cette classe va contenir deux références de maillons, une sur le premier maillon de la liste et une sur le maillon courant. On veut pouvoir se déplacer dans la liste mais aussi la modifier. Écrire des méthodes **dynamiques** pour :

1. tester si la liste est vide,
2. aller au début ou à la fin de la liste (deux méthodes distinctes),
3. savoir si on est au début ou à la fin de la liste (deux méthodes distinctes),
4. conaître la référence du maillon précédent,
5. avancer ou reculer dans la liste (deux méthodes distinctes),
6. insérer un maillon avant ou après le maillon courant (deux méthodes distinctes),
7. supprimer le maillon courant,
8. afficher une liste.

3 Listes doublement chaînées et listes génériques

On va maintenant garder dans chaque maillon non seulement une référence vers le maillon suivant, mais aussi une vers le maillon précédent pour simplifier les opérations. Quelles méthodes doivent être modifiées pour prendre en compte ce changement.

Comment faire pour pouvoir insérer n'importe quel type d'objet dans la liste et plus seulement des entiers ?