

Travaux Dirigés n°3

Java Avancé

—M1 Apprentissage—

MyList

Structures de données, listes, type paramétré etc.

► Exercice 1. MyList

On souhaite dans cet exercice redéfinir une liste chaînée contenant des `String`.

1. Créer une classe `Cell` correspondant à une cellule de la liste chaînée. Elle stocke donc un `String` ainsi qu'une référence vers le maillon suivant de la liste.
2. Créer une classe `MyList` pour qu'un utilisateur puisse manipuler une liste chaînée sans manipuler les cellules. Elle contient une référence vers la première cellule de la liste.
3. Ajouter une méthode `add(String)` ajoutant une cellule contenant la `String` en paramètre en tête de liste.
4. Ajouter une méthode `size()` indiquant le nombre d'éléments de la liste.
5. Ajouter une méthode `toString()` affichant les éléments de la liste séparés par des virgules.
6. Ajouter une méthode `addLast(String)` ajoutant une nouvelle cellule à la fin de la liste.
7. Ajouter une méthode `add(String, int i)` ajoutant une nouvelle cellule après la cellule d'indice `i`.
8. Tester toutes ces méthodes dans un `main` (attention au cas premier élément etc).

```
1 MyList ml = new MyList();
2 ml.addLast("tatu");
3 ml.add("toto");
4 ml.add("titi");
5 ml.addLast("tutu");
```

9. Ajouter une méthode `String get(int index)` renvoyant le contenu de la cellule située à l'index `index`. Que faire si l'indice est invalide ?
10. Modifier `size()` pour que cette méthode s'effectue en temps constant.
11. Écrire une méthode `sumLetters()` renvoyant la somme des lettres contenues dans la liste.
12. Testez avec les tests `MyListTest1`

► **Exercice 2. MyList générique**

On souhaite maintenant que la liste puisse contenir n'importe quel type d'objets.

1. Changer votre implémentation pour que la liste chaînée contienne des `Object` plutôt que des `String`.
2. Ajouter une méthode `contains(Object)` indiquant si un objet est contenu dans la liste.
3. Ajouter un entier dans la liste (dans le code de votre main test).
4. Que faut-il changer pour la méthode `sumLetters()` ?
5. Générer `MyList` et `Cell` avec un type paramétré. Modifier le main en conséquence.

Exercise3. HashMap

Referring to your first TP, create a HashMap which collect circles and each circle can be referred by its center.

What is the result of following code?

```
public static void main(String [] args) {  
    Map<point, Circle> m = new HashMap<>();  
    point p1 = new point(100,150);  
    Circle c1 = new Circle(p1,100);  
    m.put(p1, c1);  
    System.out.println(m.containsKey(p1);  
    System.out.println(m.containsKey(new point(1,2)));
```

How to fix it?