

Objectif

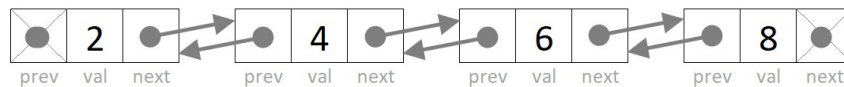
- Utiliser les types abstraits de données : pile, file, liste.
- Implémenter une **liste doublement chaînée**.
- Mettre en œuvre le concept d'**interface**, séparation interface / implémentation.
- Créer une classe utilisant un **type générique**.

Énoncé du problème

On vous demande de fournir une classe qui stocke une liste d'éléments (*d'un type générique*) et fournissant des méthodes permettant d'ajouter/de retirer des éléments en début et/ou en fin de liste.

Votre travail :

- Adaptez l'application `TestListeDblCh` afin d'utiliser `DList` comme étant de type générique: possibilité de créer une `DList` contenant à choix des entiers, des Strings, ou des Articles.
- Vous ne devriez pas avoir besoin de toucher aux méthodes `Traiter`.
- Créez la classe `VotreDList` qui implémente les 5 méthodes utilisées dans `TestListeDblCh` : **`addFirst`** & **`addLast`** qui ajoutent un élément en début/fin de liste, **`removeFirst`** & **`removeLast`** qui retournent (*et ôtent*) le premier/dernier élément de la liste, ainsi que **`isEmpty`** indiquant si la liste est vide.
- `VotreDList` conservera les éléments dans une liste doublement chaînée :



Exemples de sorties :

- En utilisant la méthode `remplirDEntiers` :

```
4 3 2 1 6 7 8 9
9 8 7 6 1 2 3 4
```

- En utilisant la méthode `remplirDeString` :

```
Dan Cloe Bob Ana Ryan Sam Tara Ugo
Ugo Tara Sam Ryan Ana Bob Cloe Dan
```