

TP Algorithmique programmation

TP 10 : Liste Chaînée

Une liste chaînée est une structure de données dans laquelle les éléments sont rangés linéairement. Chaque élément est lié à son successeur et il n'est donc pas possible d'accéder directement à un élément quelconque de la liste :



Le but de ce TP est d'implémenter cette structure de données.

Exercice 1. Écrire une classe `Paire` représentant les maillons d'une liste chaînée. Les attributs seront d'une part la valeur du maillon (appelé la *car*) et une référence vers le maillon suivant (appelé le *cdr*) d'autre part.

La classe comportera des accesseurs et des modificateurs pour chaque attribut ainsi qu'une méthode `Paire rechercher(Object)` et `toString()`.

Écrire une classe `Liste` représentant les listes chaînées et comportant des méthodes `void ajouter(Object)` (ajout en tête de la liste), `boolean rechercher(Object)` et `toString`. Vous pourrez tester vos classes à l'aide du programme de test fourni sur la page web des TP.

Exercice 2. On souhaite ajouter des méthodes permettant le parcours des listes chaînées. Écrire une classe `ItListe` héritant de `Liste` et possédant les méthodes suivantes :

- `void allerAuDebut()` initialise le parcours
- `void avancer()` avancer d'un maillon
- `Object elementCourant()` retourne le car du maillon courant
- `boolean finDeListe()` teste si on est à la fin de la liste
- `void modifier(Object)` remplace le car du maillon courant par la valeur en paramètre
- `boolean rechercher(Object)` le maillon courant devient celui ayant le paramètre en car (s'il est présent dans la liste).
- `toString` met, en particulier, en valeur le maillon courant

Vous pourrez tester vos classes à l'aide du programme de test fourni sur la page web des TP.