

## TP 7 - Structures chaînées

**Exercice 1 (Liste)** Récupérer les fichiers `liste.h`, `liste.c` et `testeListe.c` et `Makefile` sur le Moodle. Ils contiennent les fonctions vues en cours du type liste et un début de `main` permettant de tester ces fonctions.

- Commencer par jouer un peu avec le programme (après l'avoir compilé), en ajoutant l'entier 4 en première position de la liste  $l_1$ , puis l'entier 27 en troisième position de  $l_1$ , l'entier 55 en deuxième position de  $l_1$  et enfin l'entier 9 en quatrième position de  $l_1$ .
- Compléter la fonction `main` avec les instructions à exécuter pour implanter le choix (V) du menu : restitution de la mémoire occupée par la liste désignée. Tester cette fonctionnalité en essayant de ré-insérer un élément après avoir (V)ider la liste.
- Implanter les 6 dernières fonctions non implantées de `list.h`.
- Pour chacune d'elles, implanter le choix de menu correspondant dans la fonction `main`.
- Compléter dans le fichier `testListe.c` la fonction `extrairePairs` qui construit la liste des entiers pairs d'une liste donnée.
- Implanter le choix de menu correspondant, qui extrait les entiers pairs d'une liste, supprime le contenu de l'autre liste et le remplace par la liste des entiers pairs.

**Exercice 2 (Vecteur)** En vous appuyant sur l'implémentation de la fonction `changeTaille` du TP précédent, implanter le module `vector` dont l'en-tête `vector.h` est fourni sur le Moodle. Pour vos tests utiliser le programme `testVector.c` fourni sur le Moodle dont l'exécution doit vous fournir le résultat suivant.

```

1 $ ./testVector
2 Quelle taille de vecteur voulez-vous ? 7
3 Vector of size 7 (capacity 7) : [ 0 1 2 3 4 5 6 ]
4 Ajout de 11 : Vector of size 8 (capacity 12) : [ 0 1 2 3 4 5 6 11 ]
5 Ajout de 22 : Vector of size 9 (capacity 12) : [ 0 1 2 3 4 5 6 11 22 ]
6 Ajout de 33 : Vector of size 10 (capacity 12) : [ 0 1 2 3 4 5 6 11 22 33 ]
7 Ajout de 44 : Vector of size 11 (capacity 12) : [ 0 1 2 3 4 5 6 11 22 33 44 ]
8 Ajout de 55 : Vector of size 12 (capacity 12) : [ 0 1 2 3 4 5 6 11 22 33 44 55 ]
9 Ajout de 66 : Vector of size 13 (capacity 17) : [ 0 1 2 3 4 5 6 11 22 33 44 55 66 ]
10 Ajout de 77 : Vector of size 14 (capacity 17) : [ 0 1 2 3 4 5 6 11 22 33 44 55 66 77 ]
11 Ajout de 88 : Vector of size 15 (capacity 17) : [ 0 1 2 3 4 5 6 11 22 33 44 55 66 77 88 ]
12 Ajout de 99 : Vector of size 16 (capacity 17) : [ 0 1 2 3 4 5 6 11 22 33 44 55 66 77 88 99 ]
13 Ajout de 110 : Vector of size 17 (capacity 17) : [ 0 1 2 3 4 5 6 11 22 33 44 55 66 77 88 99 110 ]
14 Ajout de 121 : Vector of size 18 (capacity 22) : [ 0 1 2 3 4 5 6 11 22 33 44 55 66 77 88 99 110 121 ]
15 Vector of size 0 (capacity 0) : [ ]
16
17 $ ./testVector
18 Quelle taille de vecteur voulez-vous ? 0
19 Vector of size 0 (capacity 0) : [ ]
20 Ajout de 11 : Vector of size 1 (capacity 5) : [ 11 ]
21 Ajout de 22 : Vector of size 2 (capacity 5) : [ 11 22 ]
22 Ajout de 33 : Vector of size 3 (capacity 5) : [ 11 22 33 ]
23 Ajout de 44 : Vector of size 4 (capacity 5) : [ 11 22 33 44 ]
24 Ajout de 55 : Vector of size 5 (capacity 5) : [ 11 22 33 44 55 ]
25 Ajout de 66 : Vector of size 6 (capacity 10) : [ 11 22 33 44 55 66 ]
26 Ajout de 77 : Vector of size 7 (capacity 10) : [ 11 22 33 44 55 66 77 ]
27 Ajout de 88 : Vector of size 8 (capacity 10) : [ 11 22 33 44 55 66 77 88 ]
28 Ajout de 99 : Vector of size 9 (capacity 10) : [ 11 22 33 44 55 66 77 88 99 ]
29 Ajout de 110 : Vector of size 10 (capacity 10) : [ 11 22 33 44 55 66 77 88 99 110 ]
30 Ajout de 121 : Vector of size 11 (capacity 15) : [ 11 22 33 44 55 66 77 88 99 110 121 ]
31 Vector of size 0 (capacity 0) : [ ]

```