

ISEN école d'ingénieurs TP Allocation Mémoire

Exercice 1 :

Créer deux copies d'une chaîne de caractères passée en argument (il n'y aura donc pas d'interaction à faire avec l'utilisateur dans ce programme).

Vous convertirez ensuite la première copie en minuscules et la seconde en majuscules.

Finalement, votre programme affichera la chaîne originale et les deux copies.

Exemple :

```
➤ ./exo1 "Hello World"
Hello World
hello world
HELLO WORLD
```

Exercice 2 :

À partir de valeurs entières entrées par l'utilisateur en argument, créer une liste de valeurs paires, une liste de valeur impaires et les afficher.

Exemple :

```
➤ ./exo2 1 2 3 4 5 6 7
1 3 5 7
2 4 6
```

Notes :

- Vous supposerez que les arguments sont du bon type
- Réaliser le programme en deux étapes :
 1. Une première version où les tableaux contenant les deux listes seront dimensionnés « au plus grand », c'est-à-dire à la taille correspondant au nombre de valeurs passées au programme
Dans l'exemple ci-dessus, la taille serait donc de 7
 2. Une deuxième version où les tableaux précédents, une fois créés seront redimensionnés au plus juste.
Dans l'exemple précédent, la dimension du tableau des valeurs paires serait donc de 3 et celle des valeurs impaires de 4

Exercice 3 :

À partir d’une phrase donnée en argument, créer dynamiquement un tableau de chaînes de caractères où chaque entrée correspondra à un mot. Votre programme affichera ensuite chaque mot.

Exemple :

```
➤ ./exo3 "Welcome to the new World"
Welcome
to
the
new
World
```

Notes :

- Dans une première version, vous supposerez que les mots ne sont séparés que par un seul espace
- Dans une seconde version, avant de traiter une phrase, vous commencerez par supprimer tout espace en trop