

**TD 3**  
**Programmation C L2.1**  
**Passage par adresse, tableaux**

---

**1. Passage par adresse**

**(a) Valeurs minimum et maximum d'une liste de n valeurs entières**

- Écrire une fonction `MinMax` qui transmet à la partie appelante le minimum et le maximum de `n` entiers lus au clavier, où `n` est un paramètre de la fonction.
- Écrire un programme permettant de lire un entier positif `n` et d'afficher le minimum et le maximum de `n` valeurs lues au clavier.

**(b) Simplification de deux nombres représentant une fraction**

- Ecrire une fonction `Simplifie` qui reçoit deux valeurs en paramètre représentant le numérateur et le dénominateur d'une fraction et transforme ces valeurs en numérateur et dénominateur d'une fraction équivalente simplifiée. Par exemple, si les variables données initialement ont pour valeur 3 et 12, leur valeur après exécution de la procédure sera 1 et 4 ( $\frac{3}{12} = \frac{1}{4}$ ). On supposera dans ce cas que la procédure reçoit une fraction valide (c'est-à-dire, que le dénominateur n'est pas nul).
- Ecrire une fonction qui réalise le même travail que la fonction `Simplifie` mais qui indique une erreur grâce à la valeur retournée si le dénominateur de la première fraction est nul (1 si le déroulement est correct, 0 en cas d'erreur).

On pourra utiliser sans l'écrire une fonction `int pgcd(int a, int b)` qui renvoie le pgcd de 2 entiers `a` et `b`, `b`  $\neq$  0.

**2. Tableaux** Dans toute la suite, on considère que la taille des tableaux, transmise aux fonction est valide.

**(a) Affichage**

Ecrire une fonction qui affiche, 10 par ligne, les éléments d'un tableau d'entiers.

**(b) Somme, moyenne**

Ecrire une fonction qui reçoit un tableau, sa taille et renvoie la somme de ses valeurs. Ecrire une fonction qui reçoit un tableau, sa taille et renvoie la moyenne de ses valeurs.

**(c) miroirTranche**

Écrire une fonction qui reçoit un tableau, sa taille et deux indices `min` et `max`. Si les indices `min` et `max` sont valides, la fonctions renverse les éléments situés entre les deux indices `min` compris et `max` exclu.

**(d) Décalage à droite**

Écrire une fonction `void decaleD(int t, int taille)` qui reçoit un tableau sa taille et effectue un décalage circulaire d'une case vers la droite des éléments de `t`

(e) **Décalage à droite de k**

- *version naive* En utilisant la fonction précédente, écrire une fonction `void decaleDk(int t, int taille, unsigned int k)` qui reçoit un tableau, sa taille et un entier positif `k` et effectue un décalage circulaire de `k` cases vers la droite.

Combien la fonction effectue-t-elle d'affectations pour un `k` donné.

- *version efficace* Dessiner un tableau contenant les entiers de 0 à 9. Effectuer le miroir de la partie entre les indices 0 et 4 et le miroir de la partie entre les indices 4 à 10.

Effectuer enfin le miroir de la totalité du tableau obtenu.

Écrire une version du décalage de `k` éléments basée sur cette méthode.

Combien la fonction effectue-t-elle d'affectations pour un `k` donné.

(f) **Saisie**

Écrire une fonction qui reçoit un tableau d'entiers et sa taille et le remplit avec des valeurs lues au clavier.

(g) **Copie**

Écrire une fonction qui reçoit 2 tableaux d'entiers de même taille et copie le premier dans le second.