

### Exercice 58 : Moyenne des éléments d'une liste

Écrire une fonction qui calcule la moyenne des éléments d'une liste de réels donnée.

### Exercice 59 : Somme des éléments de deux listes

- a) Écrire une fonction qui construit la liste somme de deux listes de  $n$  éléments chacune. On suppose dans un premier temps que les listes sont de même longueur, dans une version améliorée, si une des deux listes est plus petite, on recopiera telle quelle la fin de la liste la plus longue à la fin de la liste résultat.

Exemple : soient les listes  $L1 = [12,5,18,21]$  et  $L2 = [2,8,10,50,62,100]$   
On veut construire la liste  $L3 = [14,13,28,71,62,100]$

- b) Modifier la fonction précédente pour construire la liste *en compréhension*. On suppose dans ce cas que les deux listes sont de même taille.

Indication : on peut utiliser la fonction `zip`

### Exercice 60 : Décalage à droite

Écrire un programme qui effectue un décalage circulaire vers la droite des éléments d'une liste sans utiliser de tableau intermédiaire.

On demande de modifier la liste avec le décalage, pas seulement de l'imprimer.

Exemple :  $L : 1\ 6\ 3\ 0\ 8$  Devient :  $8\ 1\ 6\ 3\ 0$

### Exercice 61 : Méthodes de tri

Écrire deux fonctions qui effectuent le tri par ordre croissant d'une liste d'entiers en appliquant respectivement chacune des deux méthodes décrites ci-dessous.

#### 1) la méthode du tri par sélection

Le principe de cette méthode est de sélectionner à chaque étape, le minimum des éléments de la liste et à l'échanger avec l'élément de la première place. On fera ensuite de même avec l'élément à placer à la deuxième place puis à la troisième etc.

#### 2) la méthode du tri à bulle

Le principe de cette méthode est de faire monter, à chaque étape, le maximum des éléments de la suite par une suite de permutations : à l'étape  $i$ , on parcourt la table de gauche à droite et on échange l'élément courant avec son suivant s'ils ne sont pas classés dans l'ordre croissant. A la fin du parcours, le maximum est bien placé, et on recommence sur la liste privée de son dernier élément et ainsi de suite jusqu'à ce qu'il n'y ait plus qu'un seul élément à classer.

### Exercice 62 : Insertion dans une liste triée

On dispose d'une liste d'entiers rangés par ordre croissant. On souhaite écrire un programme qui insère un élément à sa place dans ce tableau, c'est à dire en respectant l'ordre croissant.

- 1) Écrire une fonction `place` qui recherche la place de l'élément à insérer :
- Par une recherche séquentielle
  - Par une recherche dichotomique

Écrire le programme qui, une fois sa place connue, insère l'élément dans la liste.

### Exercice 63 : Période d'un nombre rationnel

Un nombre rationnel est donné par son numérateur et son dénominateur (entiers). Tout nombre rationnel admet un développement décimal périodique.

Exemples : la période de  $74/14$  est : 285714      La période de  $73/6$  est 6

$$\begin{array}{r|l} 74 & 14 \\ 40 & \hline 120 & 5,285714 \\ 80 & \\ 100 & \\ 20 & \\ 60 & \\ 4 & \end{array} \qquad \begin{array}{r|l} 73 & 6 \\ 13 & \hline 10 & 12,16 \\ 40 & \\ 4 & \end{array}$$

On souhaite un programme qui donne ce développement pour un nombre rationnel donné :

- 1) Écrire une fonction `pgcd` en mettant en œuvre l'algorithme d'Euclide :
- Soit  $a$  et  $b$  deux entiers naturels, si  $b$  est non nul, on peut effectuer la division euclidienne de  $a$  par  $b$ . Il existe un couple unique d'entiers  $(q,r)$  tels que :
- $a = bq + r$  et  $0 \leq r < b$
- si  $r = 0$  alors  $b$  divise  $a$  et  $\text{PGCD}(a,b) = b$
- si  $r > 0$  alors tout diviseur commun à  $a$  et  $b$  est diviseur de  $r$  et réciproquement tout diviseur commun à  $b$  et  $r$  est diviseur de  $a$ . On a donc  $\text{PGCD}(a,b) = \text{PGCD}(b,r)$ .
- 2) Rendre la fraction irréductible en divisant le numérateur et le dénominateur par leur PGCD.
- 3) Écrire la fonction qui calcule la période d'un nombre rationnel donné par son numérateur et son dénominateur.
- Pour connaître la période, il suffit d'arrêter les divisions des restes successifs par les quotients successifs lorsqu'on retrouve un reste déjà obtenu.