

Tableaux

On utilisera les définitions suivantes :

```
#define TAILLE 100
typedef int vecteur[TAILLE];
```

Exercice 1

Une liste de n nombres entiers est rangée dans un tableau \mathbf{t} de taille `TAILLE`. Écrire une fonction `insérer` qui insère un élément donné \mathbf{x} dans le tableau \mathbf{t} à la position d'indice i . Si $i \geq n$, \mathbf{x} est inséré à la position d'indice n .

Exercice 2

Une liste de n nombres entiers est rangée dans un tableau \mathbf{t} . Écrire une fonction `supprimer` qui supprime l'élément d'indice i du tableau \mathbf{t} .

Exercice 3

On considère la suite définie par $u_0 = 1$ et la récurrence :

$$u_{n+1} = u_0 u_n + u_1 u_{n-1} + \cdots + u_n u_0 = \sum_{k=0}^n u_k u_{n-k}.$$

Calculer les quatre premiers termes de la suite. Écrire en langage C, la fonction `suite` qui remplit un tableau \mathbf{u} avec les $n + 1$ premiers termes u_0, u_1, \dots, u_n de la suite.

Exercice 4 Trouver le nombre d'éléments différents dans un tableau donné.

On utilisera les définitions suivantes :

```
#define LONGUEUR 100
typedef char chaine[LONGUEUR];
```

Exercice 5

Écrire les fonctions suivantes :

1. `int longueur(chaine source)` qui retourne le nombre de caractères de la chaîne `source`.
2. `void insert(chaine source, int pos, chaine destination)` qui insère `source` e position `pos` dans `destination`.
3. `void remove(chaine source, int debut, int n)` qui supprime `n` caractères à partir de `debut`.
4. `void replace(chaine source, chaine motif, chaine par)` qui remplace toutes les occurrences de `motif` dans `source` par `par`.

Exercice 6

Écrire une fonction qui vérifie si une chaîne de caractères est un palindrome.

Exercice 7

Écrire une fonction prenant en paramètres deux chaînes de caractères `tab1` et `tab2`, qui retourne 1 si chaque lettre de `tab2` apparaît dans `tab1` avec le même nombre d'occurrences, et 0 sinon.

Exercice 8

On définit ainsi le type `matrice` :

```
#define MAX 100
typedef double matrice[MAX][MAX];
```

1. Écrire une fonction
`void copier(matrice source, int l, int c, matrice destination)`
qui copie le contenu de `source` dans `destination`.
2. À l'aide des fonctions `copier()` et `multiplication()`, définir une fonction qui calcule la puissance `n`-ième d'une matrice (carrée).