

Travaux pratiques n° 1

Exercice 1

Ecrire une fonction récursive binaire qui affiche les bits de la décomposition binaire d'un entier positif n . Exemple: le nombre 13 vaut 1101 en binaire. Etudier le cas où $n=0$

Exercice 2

Ecrire une fonction `search` qui recherche une valeur particulière dans un tableau d'entiers. La fonction prendra en paramètre le tableau et la valeur à rechercher. Elle retournera l'indice de la valeur recherchée si elle existe sinon -1. Inclure cette fonction dans un programme qui remplit initialement le tableau avec des valeurs pseudo-aléatoires (utiliser la fonction `random()`) et saisit la valeur à rechercher.

Exercice 3

Ecrire une fonction `inverse` qui renverse un tableau d'entiers. Cette fonction prendra en paramètre le tableau et sa taille. Après l'appel à la fonction `inverse`, le tableau contient les mêmes valeurs mais dans l'ordre inverse. Si le tableau initial contient les valeurs {1, 3, 12,4}. Il devra ensuite contenir {4,12, 3,1}.

Exercice 4

Ecrire une fonction `matrice_uni` créant et affichant une matrice unitaire ($M_{i,j}=1$ si $i=j$ et 0 sinon).

Exercice 5

a) Ecrire une fonction `matrice_trans` qui prend en arguments 2 matrices A , B et qui calcule la transposée de A : $B=A^t$.

La matrice transposée sera mémorisée dans une deuxième matrice B qui sera ensuite affichée.

b) Proposer une fonction `matrice_transbis` qui prend comme arguments la matrice A . A est transposée par permutation de ses éléments. Le résultat est mémorisé dans A .

Exercice 6

Ecrire une fonction trace qui trace une pyramide de caractères. La fonction prend comme arguments le caractère à afficher ainsi que le nombre n de niveaux de la pyramide.

trace('^',4) affiche

```
^
^^
^^^
^^^^
^^^^^^
```

trace('^',2) affiche

```
^
^^
```

Exercice 7

Ecrire une fonction qui permet de remplir une matrice de N lignes et N colonnes à l'aide des caractères '1', '2', '3' tel que pour N=12 :

```
1
1 2
1 2 3
1 2 3 1
1 2 3 1 2
1 2 3 1 2 3
1 2 3 1 2 3 1
1 2 3 1 2 3 1 2
1 2 3 1 2 3 1 2 3
1 2 3 1 2 3 1 2 3 1
1 2 3 1 2 3 1 2 3 1 2
1 2 3 1 2 3 1 2 3 1 2 3
```