

Fonctions_recuratives

Exercice 1

Soit une chaine de caractères, écrire un algorithme récursif permettant de déterminer sa longueur

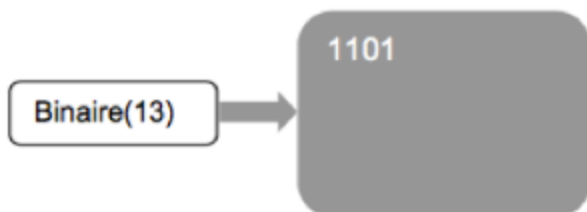
Exercice 2

Rendre récursive la fonction somme suivante :

```
def somme(L):  
    s=0  
    for val in L :  
        s+=val  
    return s
```

Exercice 3

Pour convertir un nombre entier positif N de la base décimale à la base binaire, il faut opérer par des divisions successives du nombre N par 2. Les restes des divisions constituent la représentation binaire.



Exercice 4

Un nombre N est pair si (N-1) est impair, et un nombre N est impair si (N-1) est pair.

Ecrire deux fonctions récursives mutuelles pair(N) et impair(N) permettant de savoir si un nombre N est pair et si un nombre N est impair.

Exercice 5

Soit un tableau X de N entiers, écrire une fonction récursive simple permettant de déterminer le maximum du tableau

Exercice 6

Un tableau X est trié par ordre croissant si $x(i) \leq x(i + 1), \forall i$,
écrire un algorithme récursif permettant de vérifier qu'un tableau X est trié ou non

Exercice 7

Un mot est un palindrome si on peut le lire dans les deux sens de gauche à droite et de droite à gauche. Exemple KAYAK est un palindrome. Ecrire une fonction récursive permettant de vérifier si un mot est palindrome.

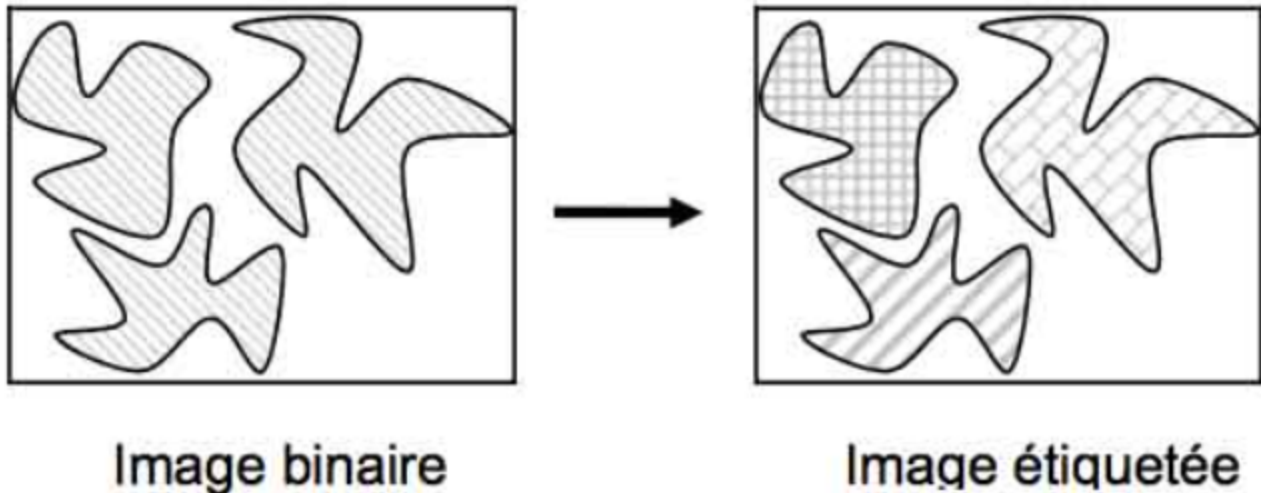
Exercice 8

Soit un tableau d'entiers contenant des valeurs 0 ou bien 1. On appelle composante connexe une suite contigue de nombres égaux à 1. On voudrait changer la valeur de chaque composante connexe de telle sorte que la première composante ait la valeur 2 la deuxième ait la valeur 3, la 3ème ait la valeur 4 et ainsi de suite. Réaliser deux fonctions :

1. La première fonction n'est pas récursive et a pour rôle de chercher la position d'un 1 dans un tableau.
2. La deuxième fonction est récursive. Elle reçoit la position d'un 1 dans une séquence et propage une valeur x à toutes les valeur 1 de la composante connexe.

Exercice 9

Soit une image binaire représentée dans une matrice à 2 dimensions. Les éléments $m[i][j]$ sont dits pixels et sont égaux soit à 0 soit à 1. Chaque groupement de pixels égaux à 1 et connectés entre eux forment une composante connexe (figure). L'objectif est de donner une valeur différente de 1 à chaque composante (2 puis 3 puis 4 etc.)



1. Ecrire une fonction récursive propager permettant de partir d'un point (i,j) situé à l'intérieur d'une composante connexe et de propager une étiquette T à tous les pixels situés à l'intérieur de la composante.
2. Ecrire une fonction etiqueter permettant d'affecter une étiquette différente à chaque composante connexe.