

## Enonce

Soit une image binaire représentée dans un tableau à 2 dimensions. Les éléments `M[i][j]`, appelés pixels, sont égaux soit à `0` soit à `1`.

Une composante d'une image est un sous-ensemble de l'image constitué uniquement de `1` et de `0` qui sont côte à côte, soit horizontalement soit verticalement.

Par exemple, les composantes de



sont



On souhaite, à partir d'un pixel égal à `1` dans une image `M`, donner la valeur `val` à tous les pixels de la composante à laquelle appartient ce pixel.

La fonction `propager` prend pour paramètre une image `M` (représentée par une liste de listes), deux entiers `i` et `j` et une valeur entière `val`. Elle met à la valeur `val` tous les pixels de la composante du pixel `M[i][j]` s'il vaut `1` et ne fait rien s'il vaut `0`.

Par exemple, `propager(M, 2, 1, 3)` donne



Compléter le code récursif de la fonction `propager` donné ci-dessous :

```
 Script Python
```

```

1 def propager(M, i, j, val):
2     if M[i][j] == ...:
3         M[i][j] = val
4
5     # l'element en haut fait partie de la composante
6     if i-1 >= 0 and M[i-1][j] == ...:
7         propager(M, i-1, j, val)
8
9     # l'element en bas fait partie de la composante
10    if ... < len(M) and M[i+1][j] == 1:
11        propager(M, ..., j, val)
12
13    # l'element à gauche fait partie de la composante
14    if ... and M[i][j-1] == 1:
15        propager(M, ..., ..., val)
16
17    # l'element à droite fait partie de la composante
18    if ... and ...:
19        propager(..., ..., ..., ...)

```

Exemple :

```
 Script Python
```

```
>>> M = [[0, 0, 1, 0], [0, 1, 0, 1], [1, 1, 1, 0], [0, 1, 1, 0]]  
>>> propager(M, 2, 1, 3)  
>>> M  
[[0, 0, 1, 0], [0, 3, 0, 1], [3, 3, 3, 0], [0, 3, 3, 0]]
```