

Corrigé sujet 36 - Année : 2023

[Sujet 36 - 2022](#) ↴

1. Exercice 1



```

1 def couples_consecutifs(tab):
2     consecutifs = []
3     for i in range(len(tab)-1):
4         if tab[i+1]==tab[i]+1:
5             consecutifs.append((tab[i],tab[i+1]))
6     return consecutifs

```

Commentaires

- La condition `tab[i+1]==tab[i]+1` permet de tester que deux éléments consécutifs de la liste sont deux entiers qui se suivent.
- On peut utiliser les définitions de liste par compréhension :



```

def recherche(tab):
    return [(tab[i],tab[i+1]) for i in range(len(tab)-1) if
    tab[i+1]==tab[i]+1]

```

2. Exercice 2



```

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

```

```
11     if j-1 >=0 and M[i][j-1] == 1:  
12         propager(M, i, j-1, val)  
13     # l'element à droite fait partie de la composante  
14     if j+1 < len(M[i]) and M[i][j+1]:  
15         propager(M, i, j+1, val)
```

1. La propagation n'a lieu que si le pixel vaut 1 (sinon elle s'arrête)
2. On relance la propagation à partir de la case voisine si celle-ci est dans la grille (première partie de la condition) et aussi dans la même composante (seconde partie de la condition)