

# Correction

NSI - 2021 Sujet Zéro (21-sujet-zero)

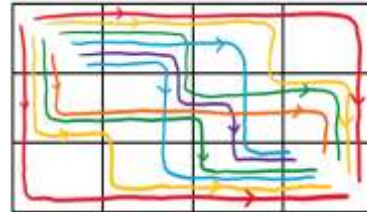
## Exercice 2 - Chemin de somme maximale

1.1. Pour atteindre la case (2, 3) en partant de (0, 0), il y a nécessairement 2 déplacements vers le bas.

1.2. Chaque chemin a 2 cases vers le bas, 3 vers la droite, sans oublier la case de départ, cela mène à une longueur de 6 cases.

2. Les chemins possibles sont :

Chemin	Somme
$(0, 0) \rightarrow (0, 1) \rightarrow (0, 2) \rightarrow (0, 3) \rightarrow (1, 3) \rightarrow (2, 3)$	11
$(0, 0) \rightarrow (0, 1) \rightarrow (0, 2) \rightarrow (1, 2) \rightarrow (1, 3) \rightarrow (2, 3)$	10
$(0, 0) \rightarrow (0, 1) \rightarrow (1, 1) \rightarrow (1, 2) \rightarrow (1, 3) \rightarrow (2, 3)$	9
$(0, 0) \rightarrow (0, 1) \rightarrow (0, 2) \rightarrow (1, 2) \rightarrow (2, 2) \rightarrow (2, 3)$	14
$(0, 0) \rightarrow (0, 1) \rightarrow (1, 1) \rightarrow (1, 2) \rightarrow (2, 2) \rightarrow (2, 3)$	13
$(0, 0) \rightarrow (1, 0) \rightarrow (1, 1) \rightarrow (1, 2) \rightarrow (1, 3) \rightarrow (2, 3)$	10
$(0, 0) \rightarrow (0, 1) \rightarrow (1, 1) \rightarrow (2, 1) \rightarrow (2, 2) \rightarrow (2, 3)$	12
$(0, 0) \rightarrow (1, 0) \rightarrow (1, 1) \rightarrow (1, 2) \rightarrow (2, 2) \rightarrow (2, 3)$	14
$(0, 0) \rightarrow (1, 0) \rightarrow (1, 1) \rightarrow (2, 1) \rightarrow (2, 2) \rightarrow (2, 3)$	13
$(0, 0) \rightarrow (1, 0) \rightarrow (2, 0) \rightarrow (2, 1) \rightarrow (2, 2) \rightarrow (2, 3)$	16



4	1	1	3
2	0	2	1
3	1	5	1

Le chemin qui permet d'obtenir la somme maximale (16) est le chemin :  $(0, 0) \rightarrow (1, 0) \rightarrow (2, 0) \rightarrow (2, 1) \rightarrow (2, 2) \rightarrow (2, 3)$ .

3. 1. Tableau des sommes maximales complétés :

4	5	6	9
6	10	8	10
9	10	15	16

3.2.  $T'[0][j] = T[0][j] + T'[0][j - 1]$  est vrai pour  $j$  différent de 0 car cela revient à parcourir la première ligne du tableau. En effet, pour obtenir la valeur d'une case  $(0, j)$ , il faut faire l'addition de la valeur de cette case avec la somme maximale déjà calculée dans la case précédente  $(0, j - 1)$ .

4. Nous recherchons le chemin réalisant la somme maximale pour aller sur une case en venant du haut ou de la gauche. Ainsi, il faut ajouter la valeur de cette case  $(i, j)$  à la plus grande somme déjà calculée sur la case du dessus  $(i - 1, j)$  ou sur celle de gauche  $(i, j - 1)$ .

5.1. Le cas de base (ou cas trivial) est celui de la case  $(0, 0)$  ou l'on retourne directement la valeur 4.

5.2.

```
def somme_max(T, i, j):  
    if i == 0 and j == 0: # Cas de base  
        return T[0][0]  
    elif i == 0: # Cas de la première ligne (récursif simple)  
        return T[i][j] + somme_max(T, i, j-1)  
    elif j == 0: # Cas de la première colonne (récursif simple)  
        return T[i][j] + somme_max(T, i-1, j)  
    else: # Cas général (récursif double)  
        return T[i][j] + max(somme_max(T, i-1, j), somme_max(T, i, j-1))
```

5.3. Pour résoudre le problème il faut faire l'appel : `somme_max(T, len(T)-1, len(T[0])-1)`