

ET 1 Ex2 Corrigé (Récursivité)

Exercice 2 : Programmation et récursivité

Question 1

On considère tous les chemins allant de la case (0, 0) à la case (2, 3) du tableau T donné en exemple.

Exemple avec $n = 3$ lignes et $p = 4$ colonnes.

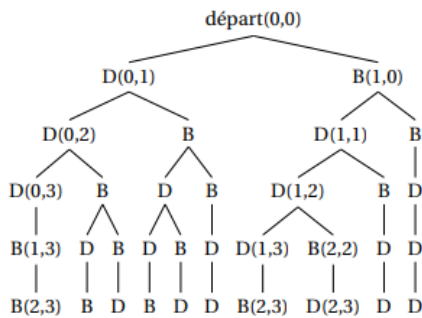
(0,0)	(0,1)	(0,2)	(0,3)
(1,0)	(1,1)	(1,2)	(1,3)
(2,0)	(2,1)	(2,2)	(2,3)

1. Un tel chemin comprend nécessairement 3 déplacements vers la droite. Combien de déplacements vers le bas comprend-il ?

Il comprend 2 chemins vers le bas.

2. La longueur d'un chemin est égal au nombre de cases de ce chemin. Justifier que tous les chemins allant de (0, 0) à (2, 3) ont une longueur égale à 6.

- Pour chaque déplacement Droite ou Bas, on arrive sur une nouvelle case. On ne peut faire que 3 déplacements vers la droite et 2 vers le bas. Ce qui fait 5 cases accessibles plus la case de départ donc un chemin est de longueur 6.
- On peut aussi faire un arbre de tous les chemins possibles (il y en a 10) et on s'aperçoit que leur longueur est 6 car chaque noeud de l'arbre correspond à une case.

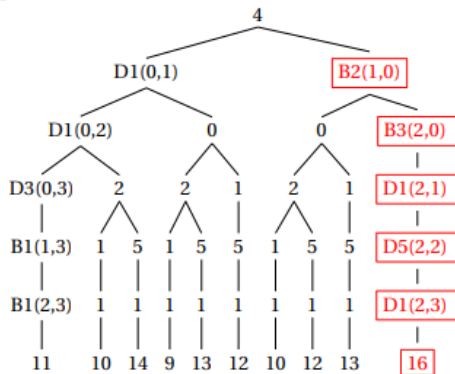


Exemple avec $n = 3$ lignes et $p = 4$ colonnes.

(0,0)	(0,1)	(0,2)	(0,3)
(1,0)	(1,1)	(1,2)	(1,3)
(2,0)	(2,1)	(2,2)	(2,3)

Question 2

En listant tous les chemins possibles allant de (0, 0) à (2, 3) du tableau T, déterminer un chemin qui permet d'obtenir la somme maximale et la valeur de cette somme.



(0,0)	(0,1)	(0,2)	(0,3)
(1,0)	(1,1)	(1,2)	(1,3)
(2,0)	(2,1)	(2,2)	(2,3)

Le chemin qui donne la somme maximale 16 est donc le chemin :

(0, 0) → (1, 0) → (2, 0) → (2, 1) → (2, 2) → (2, 3)

Question 3

On veut créer le tableau T' où chaque élément $T'[i][j]$ est la somme maximale pour tous les chemins possibles allant de $(0, 0)$ à (i, j) .

1. Compléter et recopier sur votre copie le tableau T' donné ci-dessous associé au tableau T .

$$T = \begin{array}{|c|c|c|c|} \hline 4 & 1 & 1 & 3 \\ \hline 2 & 0 & 2 & 1 \\ \hline 3 & 1 & 5 & 1 \\ \hline \end{array}$$

$$T' = \begin{array}{|c|c|c|c|} \hline 4 & 5 & 6 & 9 \\ \hline 6 & 6 & 8 & 10 \\ \hline 9 & 10 & 15 & 16 \\ \hline \end{array}$$

- Pour $T'[0][3] = 9$.

Il n'y a qu'un seul chemin possible, il est de somme $9 = 4 + 1 + 1 + 3$.

$$(0,0) = 4 \longrightarrow (0,1) = 1 \longrightarrow (0,2) = 1 \longrightarrow (0,3) = 3 \quad \text{donc} \quad S = 4 + 1 + 1 + 3 = 9$$

- Pour $T'[1][1] = 6$.

Il n'y a que 2 chemins possibles de $(0,0)$ à $(1,1)$, de somme 6 et 5.

$$(0,0) = 4 \longrightarrow (1,0) = 2 \longrightarrow (1,1) = 0 \quad \text{donc} \quad S = 4 + 2 + 0 = 6$$

$$(0,0) = 4 \longrightarrow (0,1) = 1 \longrightarrow (1,1) = 0 \quad \text{donc} \quad S = 4 + 1 + 0 = 5$$

- Pour $T'[2][2] = 15$.

Il y a 6 chemins possibles de $(0,0)$ à $(2,2)$. Remarque $\binom{4}{2} = \frac{4!}{2!2!} = 6$.

$$(0,0) = 4 \longrightarrow (1,0) = 2 \longrightarrow (2,0) = 3 \longrightarrow (2,1) = 1 \longrightarrow (2,2) = 5 \quad \text{donc} \quad S = 4 + 2 + 3 + 1 + 5 = 15;$$

$$(0,0) = 4 \longrightarrow (0,1) = 1 \longrightarrow (0,2) = 1 \longrightarrow (1,2) = 2 \longrightarrow (2,2) = 5 \quad \text{donc} \quad S = 4 + 1 + 1 + 2 + 5 = 13;$$

— Les 4 autres sont de sommes : $4 + 2 + 0 + 1 + 5 = 12$; $4 + 2 + 0 + 2 + 5 = 12$; $4 + 1 + 0 + 2 + 5 = 12$ et $4 + 1 + 0 + 1 + 5 = 11$.

2. Justifier que si j est différent de 0, alors : $T'[0][j] = T[0][j] + T'[0][j-1]$.

$(0,0)$...	$(0,j-1)$	$(0,j)$
$(1,0)$
$(2,0)$

Le seul chemin possible pour aller à la case $(0, j)$ passe par la case $(0, j-1)$.

De ce fait $T'[0][j]$ qui est la somme maximale pour tous les chemins possibles allant de $(0,0)$ à $(0, j)$ est la somme de :

- la valeur de la case $(0, j)$ soit $T[0][j]$;
- la somme maximale pour tous les chemins possibles allant de $(0, 0)$ à $(0, j-1)$.

Soit

$$T'[0][j] = T[0][j] + T'[0][j-1]$$

Question 4

Justifier que si i et j sont différents de 0, alors : $T'[i][j] = T[i][j] + \max(T'[i-1][j], T'[i][j-1])$.

...
$(i-1,0)$...	$(i-1,j-1)$	$(i-1,j)$
$(i,0)$...	$(i,j-1)$	(i,j)

- Le seul chemin possible pour aller à la case (i, j) passe par la case $(i, j-1)$ ou par la case $(i-1, j)$.

— De ce fait $T'[i][j]$ qui est la somme maximale pour tous les chemins possibles allant de $(0,0)$ à (i, j) est la somme de :

- la valeur de la case (i, j) soit $T[i][j]$;
- le maximum entre :
 - la somme maximale pour tous les chemins possibles allant de $(0, 0)$ à $(i, j-1)$ soit $T'[i][j-1]$;
 - la somme maximale pour tous les chemins possibles allant de $(0, 0)$ à $(i-1, j)$ soit $T'[i-1][j]$.

Soit

$$T'[i][j] = T[i][j] + \max(T'[i-1][j], T'[i][j-1])$$

Question 5

On veut créer la fonction récursive `somme_max` ayant pour paramètres un tableau `T`, un entier `i` et un entier `j`. Cette fonction renvoie la somme maximale pour tous les chemins possibles allant de la case (0, 0) à la case (i, j).

1. Quel est le cas de base, à savoir le cas qui est traité directement sans faire appel à la fonction `somme_max`? Que renvoie-t-on dans ce cas?
Le cas de base est le cas où le tableau ne contient qu'une seule case dans ce cas on renvoie `T[0][0]`.
2. À l'aide de la question précédente, écrire en Python la fonction récursive `somme_max`.

```
1 def somme_max(T, i, j):
2     # Cas d'arrêt (ou de base)
3     if i == 0 and j == 0 :
4         return T[0][0]
5
6     # Cas général, on utilise les résultats de la question 4/
7     else :
8         if j == 0: # On ne peut que "descendre"
9             return T[i][0] + somme_max(T, i-1, 0)
10        elif i == 0 : # On ne peut qu'aller vers la droite
11            return T[0][j] + somme_max(T, 0, j-1)
12        else:
13            return T[i][j] + max(somme_max(T, i-1, j), somme_max(T, i, j-1))
14
15
16 # Jeu de test avec l'exemple de l'exercice
17 T = [ [4,1,1,3], [2,0,2,1], [3,1,5,1] ]
18 print(somme_max(T,2,3)) # Attendu : 16
19 print(somme_max(T,1,1)) # Attendu : 6
20 print(somme_max(T,2,2)) # Attendu : 15
```

16
6
15