



Préparation à l'épreuve pratique NSI

11

Déterminer si une matrice, c'est-à-dire un tableau contenant $[N,N]$ éléments, est triangulaire supérieure.

Nom :

Note :

/ 20

Classe :

1 Spécification du programme à réaliser

1.1 Description générale

On dispose d'un tableau carré de nombres X de dimensions $[N, N]$. Ce tableau possède autant d'éléments en lignes et en colonnes. On peut le représenter comme ci-dessous :

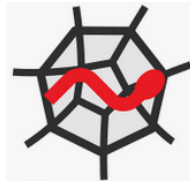
L'élément a_{ij} représente l'élément présent ligne $n^{\circ}i$ et colonne $n^{\circ}j$.

$$i \rightarrow \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1j} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2j} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{i1} & a_{i2} & \dots & a_{ij} & \dots & a_{in} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nj} & \dots & a_{nn} \end{pmatrix}$$

\uparrow
 j

On souhaite déterminer si la matrice, le tableau de nombres, est de forme triangulaire supérieure c'est-à-dire si tous les éléments présents sous la diagonale sont nuls. La forme est indiquée ci-contre

$$\begin{pmatrix} a_{1,1} & \dots & \dots & a_{1,n} \\ 0 & \ddots & \ddots & a_{2,n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & a_{n,n} \end{pmatrix}$$



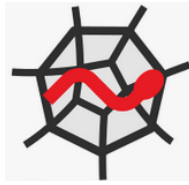
<u>Entrées</u>	<u>Sortie</u>
La matrice $X [1\dots N, 1\dots N]$	Un booléen valant vrai si la matrice est triangulaire, faux sinon

1.2 Pseudo code de l'algorithme

```
B ← vrai  
I ← 2  
tant que (I ≤ N) et (B) faire  
    J ← 1  
    tant que (J ≤ (I-1)) and (B) faire  
        si  $X(I,J) \neq 0$  alors  
            B ← Faux  
        fin si  
        J ← J + 1  
    fin tant que  
    I ← I + 1  
fin tant que  
retourne B
```

1.3 Amélioration


On vérifiera dans la fonction la pré-condition nécessaire à savoir que la matrice est bien carrée. Si ce n'est pas le cas la fonction doit renvoyer None.



2 Codage et mise en œuvre

2.1 Script de l'exercice



 NSI-PROG-011-Test-Matrice-Triangulaire.py

2.2 Fonction à compléter

Votre fonction à réaliser

```
def verif(table):
```

2.3 Exemples de résultats attendus

```
Test de la matrice tableau_1 :
```

```
[2, 7, 6, 5]  
[0, 5, 1, 0]  
[0, 0, 1, 2]  
[0, 0, 0, 3]
```

```
True
```

```
Test de la matrice tableau_2 :
```

```
[2, 0, 2, 5, 3, 4, 8]  
[0, 5, 1, 0, 1, 2, 0]  
[8, 0, 1, 2, 3, 0, 1]  
[0, 0, 4, 3, 0, 0, 0]  
[0, 5, 1, 0, 1, 2, 0]  
[1, 0, 1, 2, 3, 0, 1]  
[0, 0, 4, 3, 0, 0, 0]
```

```
False
```

