Exercice 1 : On considère la matrice $A = \begin{pmatrix} -3 & 0 & 4 \\ 1 & 2 & -1 \\ -6 & 0 & 7 \end{pmatrix}$.

Compléter la ligne de code suivante pour définir la matrice A en **Python**, ainsi que la matrice identité $I = I_3$.

```
A = np.array([[-3,0,4], [1,2,-1], [-6,0,7]])
I = np.eye(3)
```

Le script

```
M = A - I
N = np.dot(A-2*I, A-3*I)
print(np.dot(M,N))
```

```
[[0. 0. 0.]
 [0. 0. 0.]
 [0. 0. 0.]]
```

produit l'affichage ci-contre :

Que peut-on en déduire ? En particulier, lister les valeurs propres possibles de A.

Démonstration.
On en déduit que $P(x) = (x-1)(x-2)(x-3)$ est un polynôme annulateur de A. Ainsi, les valeurs propres possibles de A sont : 1, 2 et 3. □

Compléter le script suivant afin de vérifier que ce sont bien des valeurs propres via un calcul de rang.

```
r1 = al.matrix_rank(A-I)
r2 = al.matrix_rank(A-2*I)
r3 = al.matrix_rank(A-3*I)
print(r1,r2,r3)
```

On admet que **Python** renvoie `2,2,2`. Que peut-on en déduire sur la dimension des sous-espaces propres associés ?

Démonstration.
D'après le théorème du rang :

$$\dim(E_1(A)) = \dim(E_2(A)) = \dim(E_3(A)) = 1$$

□

Compléter le script suivant afin qu'il affiche la matrice A^n après que l'utilisateur ait choisi la valeur de $n \in \mathbb{N}$.

```
n = int(input('Entrez la valeur de n :'))
print(al.matrix_power(A,n))
```