Matrice associée à une transformation du plan et à sa réciproque

Cette proposition d'activité peut être envisagée sous plusieurs motivations :
→ travailler la correspondance équation matricielle ↔ système d'équations linéaires (question 5) ;
→ penser le produit matriciel comme une action géométrique (ensemble de l'activité) ;
→ utiliser l'interprétation matricielle d'un problème géométrique pour donner du sens à l'inversion d'une matrice (questions 6bc).

Le plan est muni d'un repère orthonormé $(O ; \vec{i}, \vec{j})$.
On considère la transformation géométrique du plan qui à tout point M de coordonnées $(x_M ; y_M)$ associe le point N de coordonnées :

$$(3x_M + 2y_M ; 2x_M + 3y_M).$$

1. Définir un point M du plan par ses coordonnées, et calculer les coordonnées du point image N par cette application.

2. Rédiger le programme ci-contre à l'aide du logiciel Algobox.
Que permet-il de faire ?

```
# Importation des modules

from random import random
import matplotlib
import matplotlib.pyplot as plt

# ...

def points_aleatoire(n) :
    l_abs, l_ord = [],[]
    for i in range(n):
        l_abs.append(10*random())
        l_ord.append(10*random())
    return l_abs,l_ord

# ...

x,y = points_aleatoire(1000)
plt.scatter(x,y,marker='.')
plt.show()
```

3. On souhaite maintenant observer l'image d'un carré par cette application.

Modifier le programme ci-contre pour qu'il choisisse au hasard 10000 fois un point dans un carré de côté 10 unités, et affiche son point image par cette application.

Que constate-t-on ?

Peut-on obtenir n'importe quel point du plan par cette application ?

4. A l'aide d'un système d'équations linéaires, montrer qu'il existe un unique point M dont l'image par cette application est N(4 ; 5). Donner ses coordonnées.

Remarque : D'une façon générale, pour un point quelconque $N(x_N ; y_N)$ du plan, on peut montrer qu'il existe un unique point M dont l'image par cette application est N. Les coordonnées du point M sont alors données par les formules :

$$\begin{cases} \frac{3}{5}x_N - \frac{2}{5}y_N = x_M \\ -\frac{2}{5}x_N + \frac{3}{5}y_N = y_M \end{cases}.$$

5. Déterminer la matrice A telle que le système d'équations $\begin{cases} 3x_M + 2y_M = x_N \\ 2x_M + 3y_M = y_N \end{cases}$ soit équivalent à l'égalité : $A \times \begin{pmatrix} x_M \\ y_M \end{pmatrix} = \begin{pmatrix} x_N \\ y_N \end{pmatrix}$.

De même, déterminer la matrice B telle que le système $\begin{cases} \frac{3}{5}x_N - \frac{2}{5}y_N = x_M \\ -\frac{2}{5}x_N + \frac{3}{5}y_N = y_M \end{cases}$ soit équivalent à l'égalité : $B \times \begin{pmatrix} x_N \\ y_N \end{pmatrix} = \begin{pmatrix} x_M \\ y_M \end{pmatrix}$.

6.a. Entrer les matrices A et B dans votre calculatrice.

Casio
→ accéder au menu RUN-MAT
→ appuyer sur F3 (>MAT)
→ sélectionner une matrice (Mat A, Mat B, …) et entrer ses dimensions, puis ses coefficients.
La matrice est mémorisée ; lors des calculs, appeler la matrice A par SHIFT 2 ALPHA A

TI
→ appuyer sur la touche Matrice
→ grâce à la flèche droite, accéder au menu EDIT
→ sélectionner une matrice ([A], [B], …) et entrer ses coefficients.
La matrice est mémorisée ; lors des calculs, appeler la matrice A par la touche Matrice et choisir [A]

b. A l'aide de la calculatrice, calculer : $C = B \times A$.

c. Calculer alors la produit matriciel $C \times \begin{pmatrix} x_M \\ y_M \end{pmatrix}$.

Le résultat obtenu est-il normal compte tenu de ce qu'il représente géométriquement ?

Éléments de correction :

1/ Prenons par exemple le point M(4 ; 9).

Alors $x_N = 3x_M + 2y_M = 3\times4+2\times9=12+18=30$ et $y_N = 2x_M + 3y_M = 2\times4+3\times9=8+27=35$.

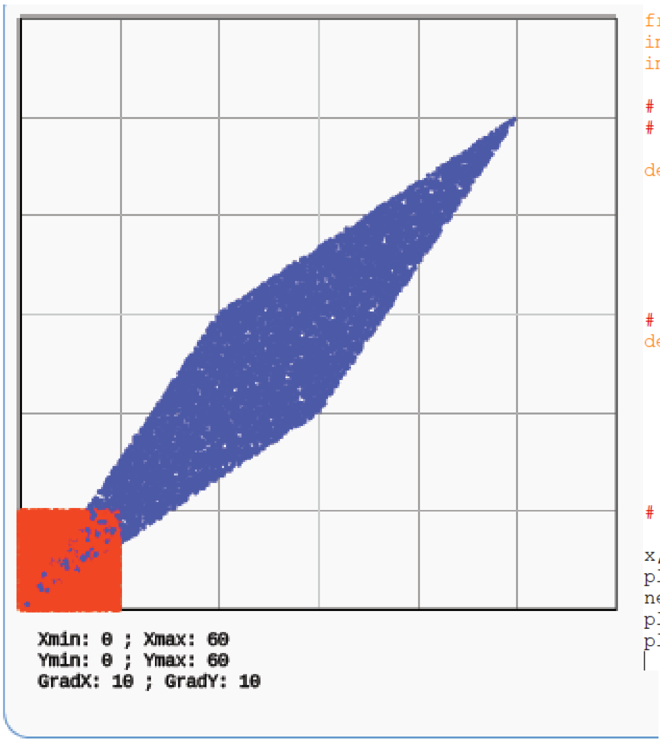
L'image obtenue est alors le point N(30 ; 35).

2/ Ce programme permet d'obtenir un carré de côté 10 unités contenant *n* points avec *n* le nombre donné à l'entrée.

3/ Le programme demandé est le suivant :

On constate que l'image du carré par cette application est un losange passant par l'origine et le point de coordonnées (50 ; 50).

(Penser à modifier la fenêtre sous Algobox).




```python
# Importation des modules

from random import random
import matplotlib
import matplotlib.pyplot as plt

# Création d'une liste de points aléatoires dont les coordonnées
# sont comprises entre (0,0) et (10,10) exclu

def points_aleatoire(n) :
    l_abs, l_ord = [],[]
    for i in range(n):
        l_abs.append(10*random())
        l_ord.append(10*random())
    return l_abs,l_ord

# Création de la liste des points image
def image(x,y) :
    l = len(x)
    newx, newy = [],[]
    for i in range(l) :
        newx.append(3*x[i]+2*y[i])
        newy.append(2*x[i]+3*y[i])
    return newx,newy

# Tracé des points

x,y = points_aleatoire(10000)
plt.scatter(x,y,marker='.')
newx,newy = image(x,y)
plt.scatter(newx,newy,marker='.')
plt.show()
```

On peut conjecturer que tout point contenu dans ce losange admet un antécédent par l'application précédente se trouvant dans le carré initial mais qu'en est-il pour un point extérieur à ce losange ?

Prolongements possibles : Caractériser le point antécédent d'un point situé sur un des côtés du losange.
Trouver un point antécédent du point E(10 ; 30) ne se trouvant pas dans le losange.

4/ Cela revient à résoudre le système : $\begin{cases} 3x+2y=4 & (1) \\ 2x+3y=5 & (2) \end{cases}$.

Par opérations, on multiplie par 2 la ligne (1) puis par 3 la ligne (2) et on fait leur différence pour annuler les *x*.

D'où $2(3x+2y)-3(2x+3y) = 2\times4-3\times5$

Soit $6x+4y-6x-9y = 8-15$ càd $-5y = -7$ et alors $y = \frac{7}{5}$.

En réimplantant dans l'équation (1) : $3x+2\times\frac{7}{5}=4$ et alors $3x=4-\frac{14}{5}=\frac{6}{5}$ et donc $x = \frac{2}{5}$.

Donc le point M($\frac{2}{5}$; $\frac{7}{5}$) est l'unique point du plan dont l'image par l'application est le point N(4 ; 5).

5/ On obtient les matrices $A = \begin{pmatrix} 3 & 2 \\ 2 & 3 \end{pmatrix}$ et $B = \begin{pmatrix} \frac{3}{5} & -\frac{2}{5} \\ -\frac{2}{5} & \frac{3}{5} \end{pmatrix}$.

6/ B/et C/ On trouve que $C = B\times A = I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ et donc $C\begin{pmatrix} x_M \\ y_M \end{pmatrix} = \begin{pmatrix} x_M \\ y_M \end{pmatrix}$.

Multiplier par A permet de trouver l'image et multiplier par B permet de déterminer l'antécédent donc la matrice C est associée à l'application « identité » qui à un point M du plan, associe ce même point M.