TP Courbes paramétrées 24/03/2021

DOS SANTOS Mathieu
SARTORI Tom

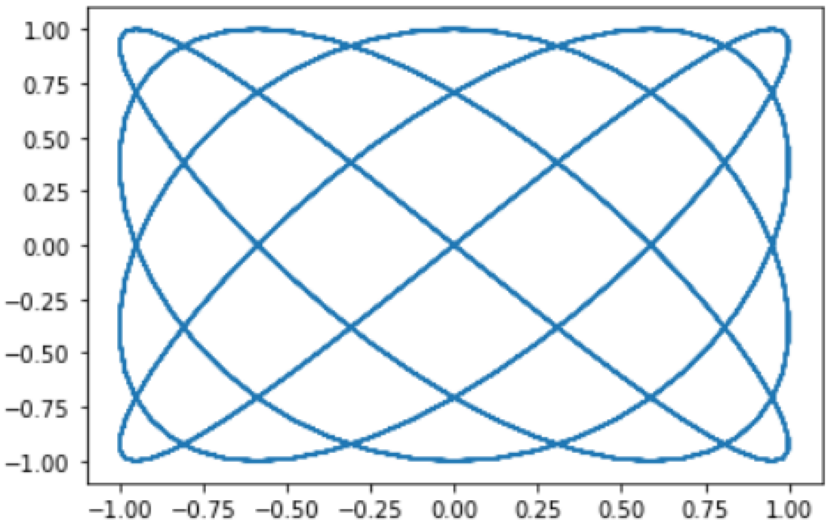
Sommaire

Partie 1

Question 1

```python
import numpy as np
import matplotlib.pyplot as plt

# Question 1 :
T = np.linspace(0, 2 * np.pi, 1001)

X = np.sin(20 * T )
Y =np.sin(25*T)

plt.plot(X, Y)
plt.show()
```

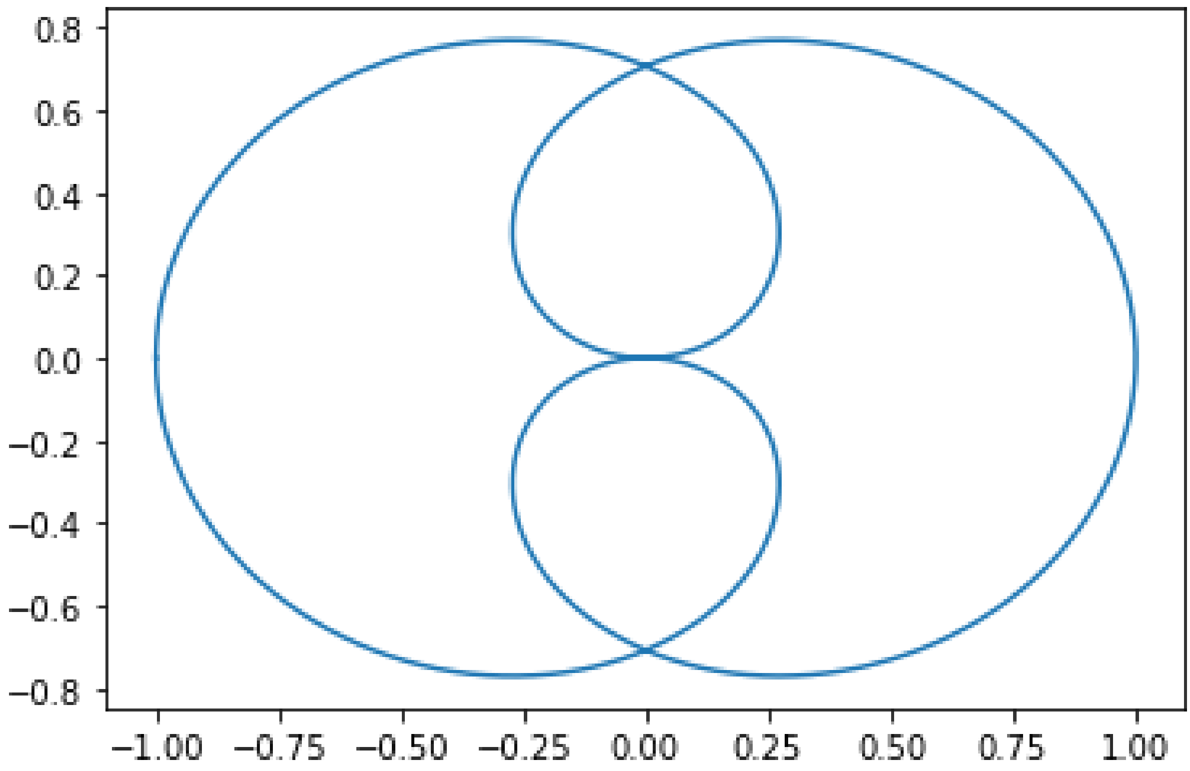
$$\begin{cases} x(t) = sin(20t) \\ y(t) = sin(25t) \end{cases}$$



Question 2.a

$$\text{(a)} \quad \begin{cases} x(t) = -cos(2t)\,cos(t) \\ y(t) = -sin(2t)\,cos(t) \end{cases} \qquad t \in [0, 2\pi]$$

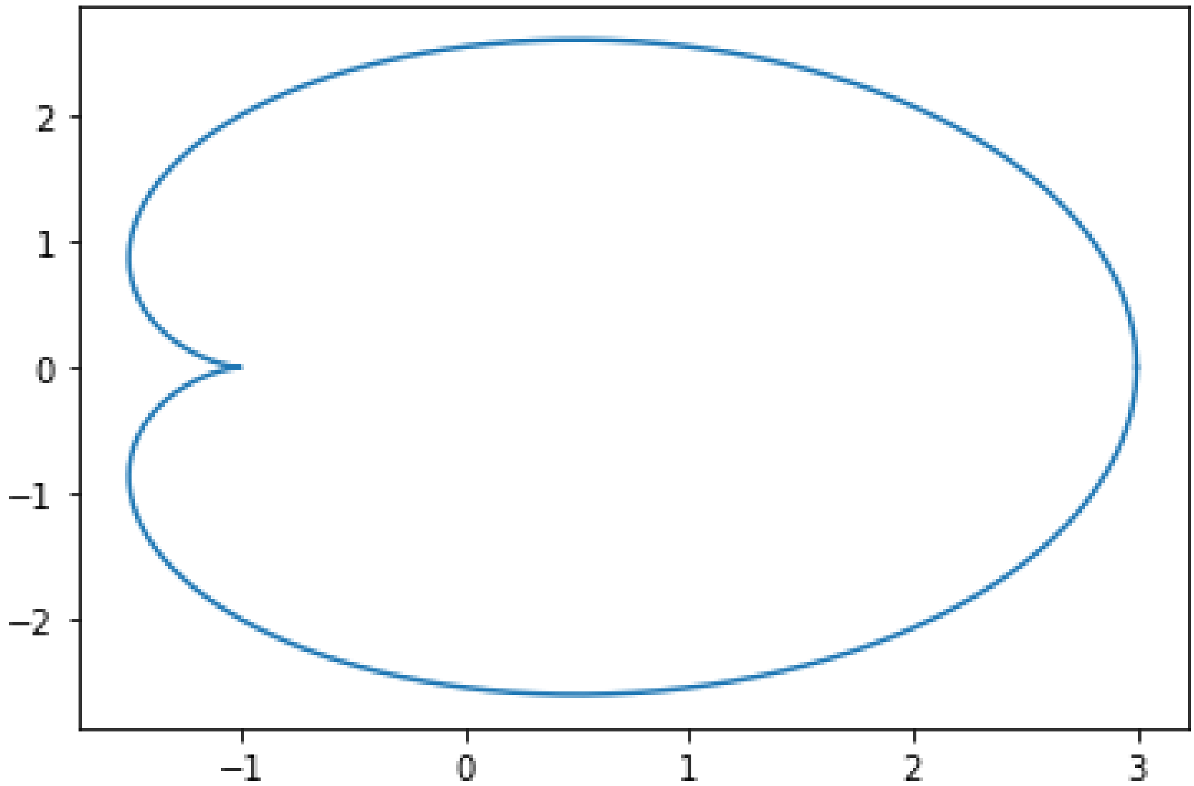
```
# Question 2.a

T = np.linspace(0, 2 * np.pi, 1001)

X = - np.cos(2 * T) * np.cos(T)
Y = - np.sin(2 * T) * np.cos(T)

plt.plot(X, Y)
plt.show()
```



Question 2.b

$$\text{(b)} \quad \begin{cases} x(t) = 2cos(t) + cos(2t) \\ y(t) = 2sin(t) + sin(2t) \end{cases} \qquad t \in [0, 2\pi]$$

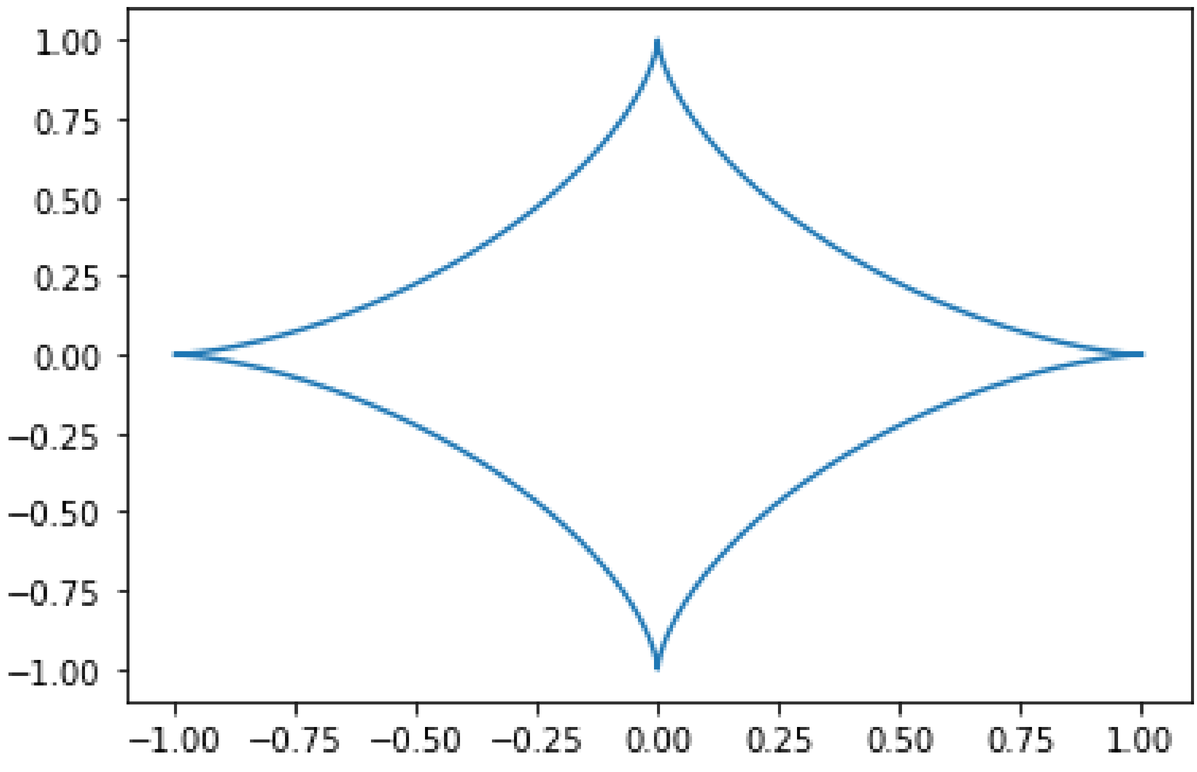
```
# Question 2.b

T = np.linspace(0, 2 * np.pi, 1001)

X = 2 * np.cos(T) + np.cos(2 * T)
Y = 2 * np.sin(T) + np.sin(2 * T)

plt.plot(X, Y)
plt.show()
```



Question 2.c

$$\text{(c)} \quad \begin{cases} x(t) = cos^3(t) \\ y(t) = sin^3(t) \end{cases} \qquad t \in [0, 2\pi]$$

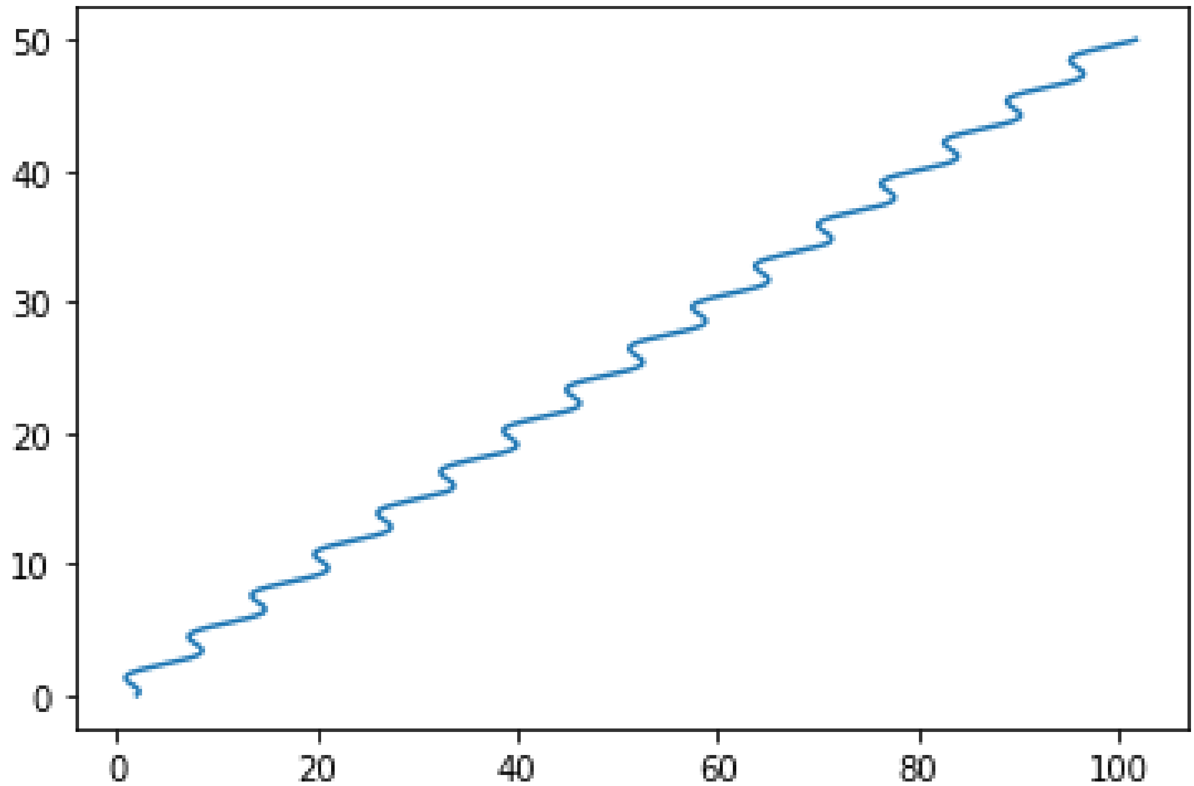
```
# Question 2.c

T = np.linspace(0, 2 * np.pi, 1001)

X = np.cos(T) ** 3
Y = np.sin(T) ** 3

plt.plot(X, Y)
plt.show()
```



Question 2.d

$$\text{(d)} \begin{cases} x(t) = t + 2cos(t) \\ y(t) = \frac{1}{2}t \end{cases} \qquad t \in [0, 100]$$

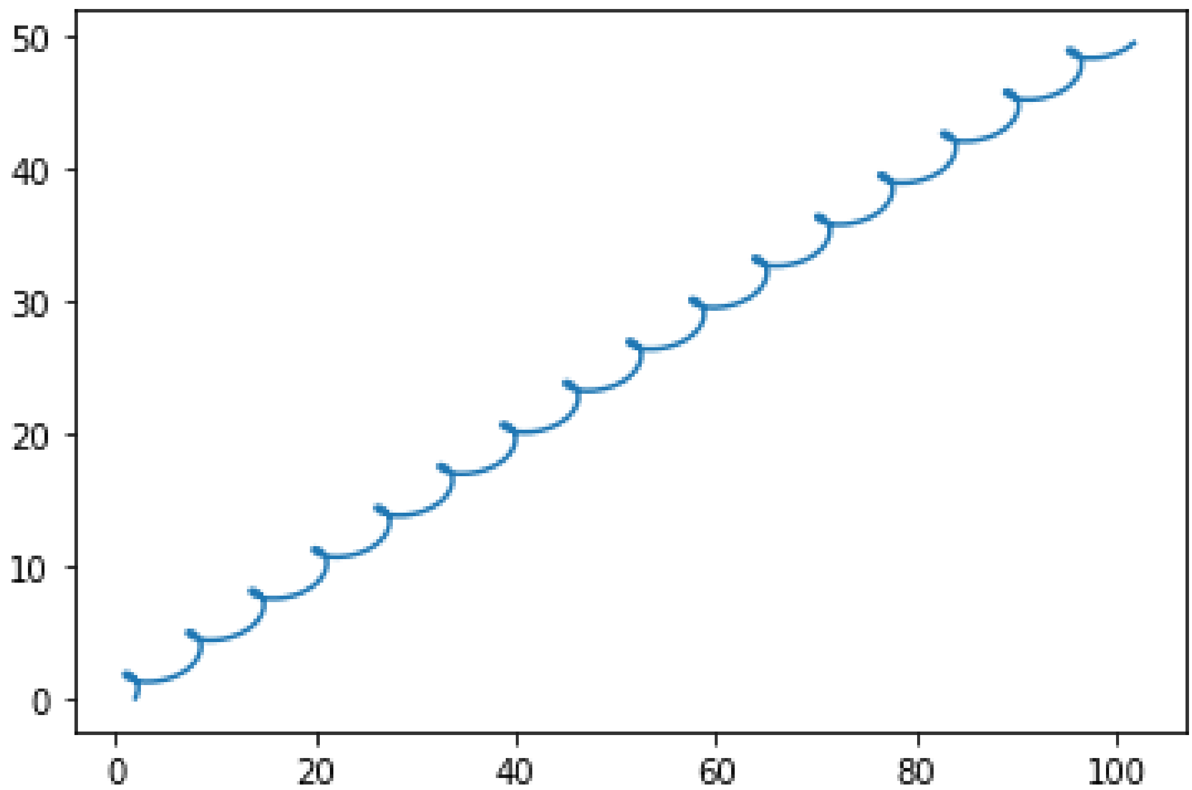
```
# Question 2.d

T = np.linspace(0, 100, 1001)

X = T + (2 * np.cos(T))
Y = 0.5 * T

plt.plot(X, Y)
plt.show()
```



Question 2.e

$$\text{(e)} \quad \begin{cases} x(t) = t + 2cos(t) \\ y(t) = \frac{1}{2}t + sin(t) \end{cases} \qquad t \in [0, 100]$$

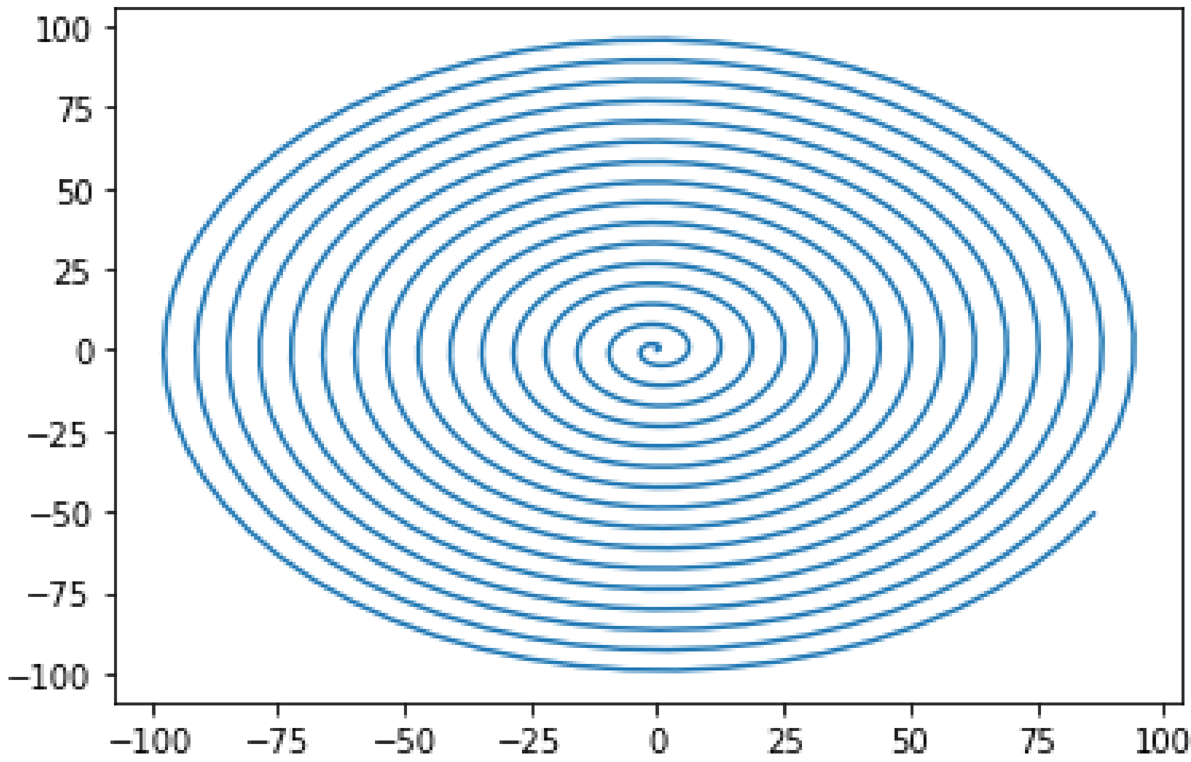
```
# Question 2.e

T = np.linspace(0, 100, 1001)

X = T + (2 * np.cos(T))
Y = (0.5 * T) + np.sin(T)

plt.plot(X, Y)
plt.show()
```



Question 2.f

$$(\mathrm{f}) \quad \begin{cases} x(t) = t \times cos(t) \\ y(t) = t \times sin(t) \end{cases} \qquad t \in [0, 100]$$

```
# Question 2.f

T = np.linspace(0, 100, 1001)

X = T * np.cos(T)
Y = T * np.sin(T)

plt.plot(X, Y)
plt.show()
```



Partie 2

Question 1.a

$$M(t)\begin{cases} x(t) = x_0(1-3t+3t^2-t^3) + x_1(3t-6t^2+3t^3) + x_2(3t^2-3t^3) + x_3(t^3) & y(t) = y_0(1-3t+3t^2-t^3) + y_1(3t-6t^2+3t^3) + y_2(3t^2-3t^3) + y_3(t^3) \\ x(t) = x_0 + t(-3x_0+3x_1) + t^2(3x_0-6x_1+3x_2) + t^3(-x_0+3x_1-3x_2+x_3) & \end{cases}$$

b) En lisant les polynomes précédents, on en déduit

$$B = \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Question 1.b

$$M(r) = P \times B \times \begin{pmatrix} 1 \\ r \\ r^2 \\ r^3 \end{pmatrix}$$

$$= P \times \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 \\ r \\ r^2 \\ r^3 \end{pmatrix}$$

$$= P \times \begin{pmatrix} 1 - 3r + 3r^2 - r^3 \\ 3r - 6r^2 + 3r^3 \\ 3r^2 - 3r^3 \\ r^3 \end{pmatrix}$$

$$= \begin{pmatrix} x_0 & x_1 & x_2 & x_3 \\ y_0 & y_1 & y_2 & y_3 \end{pmatrix} \times \begin{pmatrix} 1 - 3r + 3r^2 - 3r^3 \\ 3r - 6r^2 + 3r^3 \\ 3r^2 - 3r^3 \\ r^3 \end{pmatrix}$$

$$M(r) = \begin{pmatrix} x_0(1-3r+3r^2-3r^3) + x_1(3r-6r^2+3r^3) + x_2(3r^2-3r^3) + x_3 r^3 \\ y_0(1-3r+3r^2-3r^3) + y_1(3r-6r^2+3r^3) + y_2(3r^2-3r^3) + y_3 r^3 \end{pmatrix}$$

On retombe sur l'expression du début. Donc

$$M(r) = P \times B \times \begin{pmatrix} 1 \\ r \\ r^2 \\ r^3 \end{pmatrix}$$

avec P la matrice des 4 points A_0, A_1, A_2, A_3

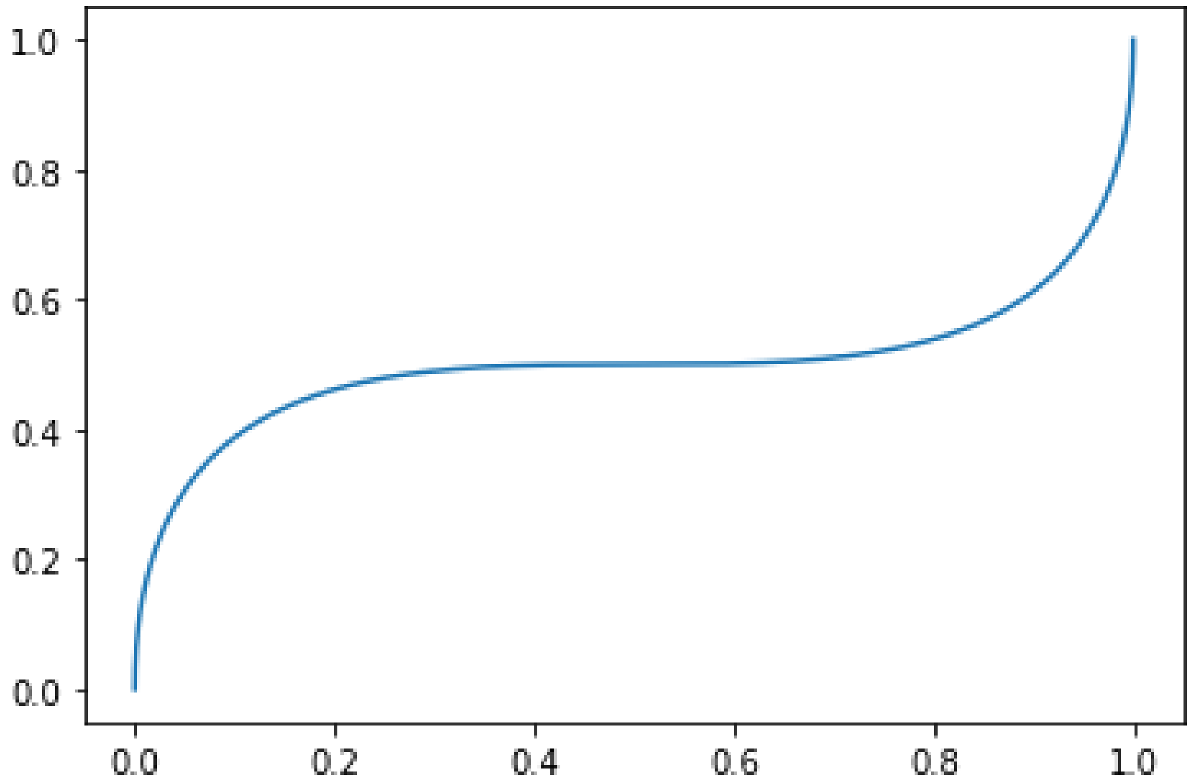
et $B = \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Question 2 (traceur)

```python
def traceur (P):
    B = np.array(
        [[1, -3, 3, -1],
         [0, 3, -6, 3],
         [0, 0, 3, -3],
         [0, 0, 0, 1]]
        )

    T = np.linspace(0, 1, 1001)

    Vect = np.array([np.ones(1001), T, T**2, T**3])

    res = np.dot(P, np.dot(B, Vect))

    plt.plot(res[0,:], res[1, :])
    return res;
```

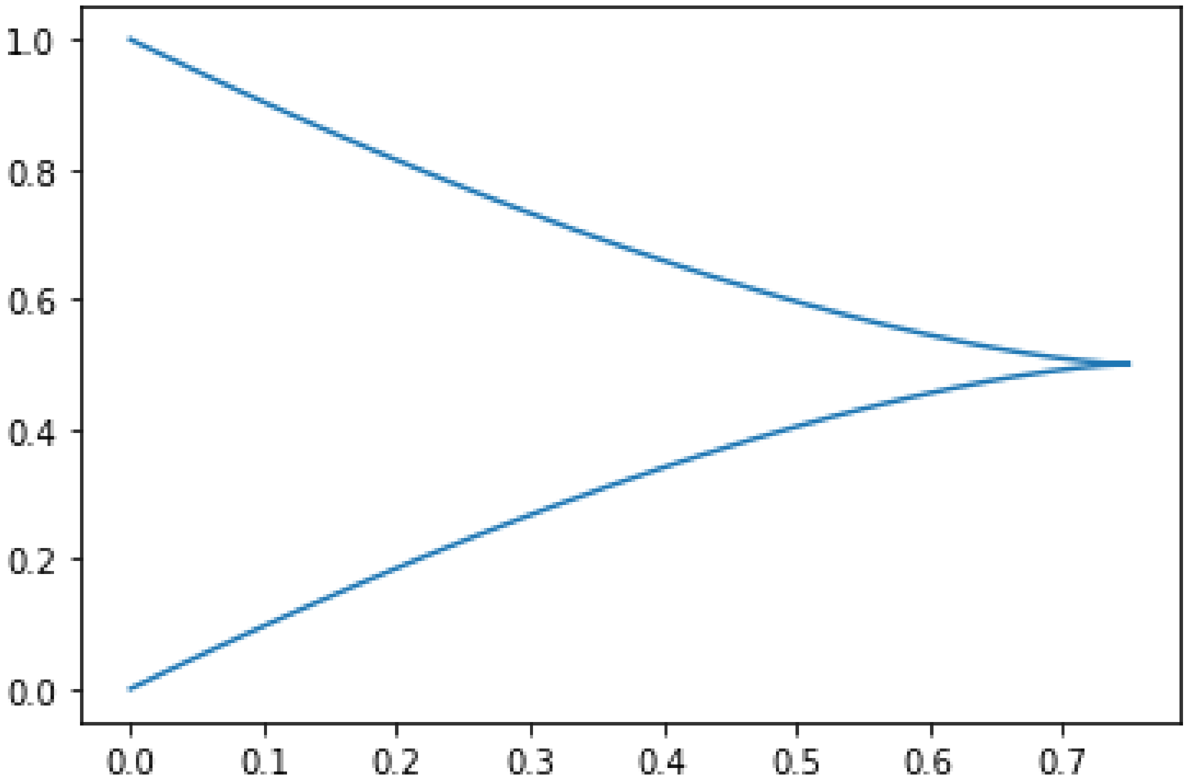
Question 3.a

```
# Question 3.a
P = np.array(
    [[0., 0, 1, 1],
     [0, 1, 0, 1]]
    )

traceur(P)
plt.show()
```



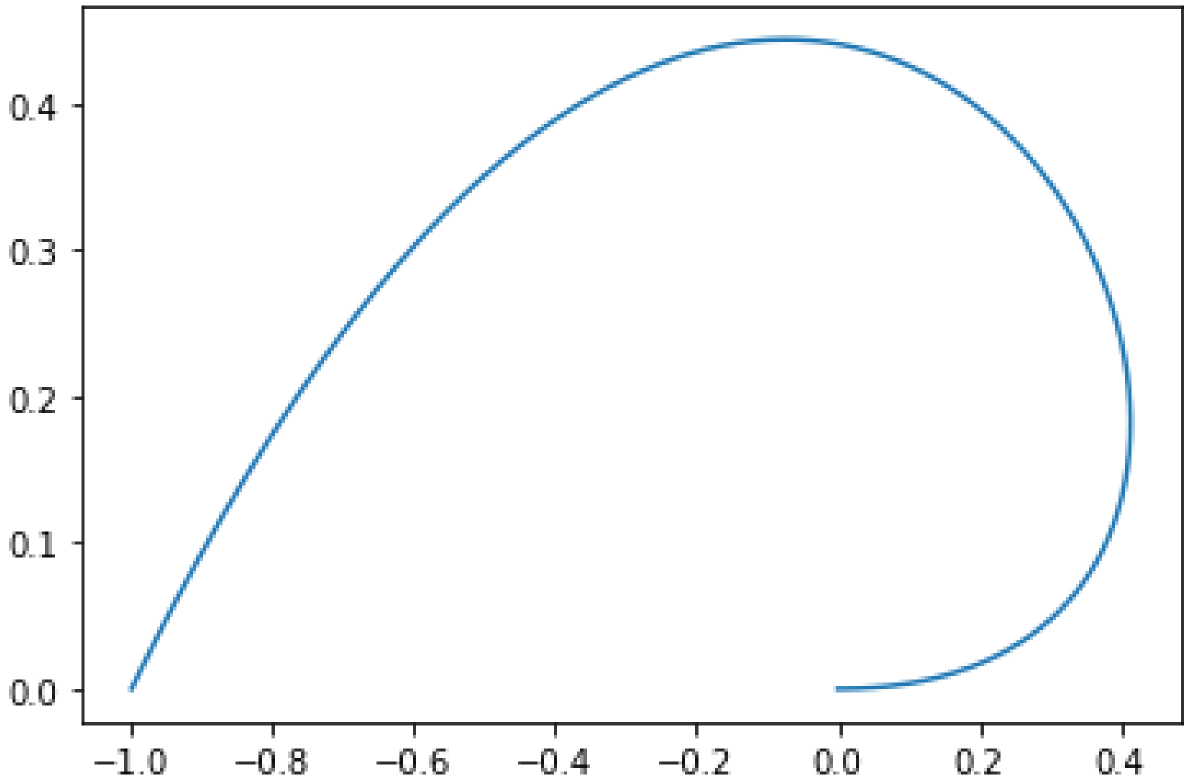
Question 3.b

```
# Question 3.b
P = np.array(
    [[0., 1, 1, 0],
     [0, 1, 0, 1]]
    )

traceur(P)
plt.show()
```



Question 3.c

```
# Question 3.c
P = np.array(
    [[0., 1, 0, -1],
     [0, 0, 1, 0]]
    )

traceur(P)
plt.show()
```



Question 3.d

```
# Question 3.d
P = np.array(
    [[0., 1, -3, -1],
     [0, 2, 2, 0]]
    )

traceur(P)
plt.show()
```