Suite de Fibonnacci Pour programmer le calcul d'un terme d'une suite récurrente du type $u_n = f(u_{n-1})$ nous avons besoin d'une seule variable u et d'une boucle, par exemple pour calculer le dixième terme de la suite $u_n = \sqrt{u_{n-1} + 4}$ de premier terme $u_0 = 1$

Compléter le programme suivant en Python

```
from math import *
u = ....
for i in range(....):
     ......
```

La suite de Fibonacci est définie par $F_0 = 0$, $F_1 = 1$ et $F_n = F_{n-1} + F_{n-2}$, ce n'est pas une suite récurrente du type $u_n = f(u_{n-1})$, car pour calculer un terme nous avons besoin des deux termes précédents Faut il deux variables pour programmer le calcul de F_n ?

On aimerait se ramener à une situation où comme précédemment une seule variable est mise à jour. Pour cela on va utiliser un **vecteur**

$V_n = (F_n, F_{n+1})$ pour $n \geqslant 0$

Exercice

1. Que vaut V_{n+1} ?
2. On cherche **une relation de passage** de V_n à V_{n+1}, pour cela on note $V_{n+1} = (x', y')$ et $V_n = (x, y)$. Exprimer x' et y' en fonction de x et y
3. Si on a deux relations **linéaires** telles que $x' = ax+by$ et $y' = cx+dy$ permettant de passer du vecteur (x, y) au vecteur (x', y') on traduit cela en langage matriciel
   $\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$
   Cette traduction est une sorte de compression où il y avait 2 égalités **linéaires** on a une seule égalité où $\begin{pmatrix} x' \\ y' \end{pmatrix}$ *et* $\begin{pmatrix} x \\ y \end{pmatrix}$ sont des **vecteurs colonnes** et $\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$ est **le produit de la matrice** $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ par le vecteur colonne $\begin{pmatrix} x \\ y \end{pmatrix}$
   Donner la matrice associée A à la suite de Fibonnacci
4. Calculer $V_1 = AV_0$ puis $V_2 = AV_1$ mais on aimerait que $V_2 = A(AV_0)$ est aussi égal à $(A \times A)V_0$ (associativité) et il reste à définir ce qu'est le produit $\times$ de deux matrices
   Revenons au cas général avec les notations utilisées ci-dessus :
   $\begin{pmatrix} x' \\ y' \end{pmatrix} = \underbrace{\begin{pmatrix} a & b \\ c & d \end{pmatrix}}_{M} \begin{pmatrix} x \\ y \end{pmatrix}$

Vérifier que l'on peut écrire cette relation $\begin{pmatrix} x' \\ y' \end{pmatrix} = x \begin{pmatrix} a \\ c \end{pmatrix} + y \begin{pmatrix} b \\ d \end{pmatrix}$ où $x \begin{pmatrix} a \\ c \end{pmatrix}$ est égal à $\begin{pmatrix} xa \\ xc \end{pmatrix}$

Autrement dit on peut écrire le vecteur colonne $\begin{pmatrix} x' \\ y' \end{pmatrix}$ **comme une combinaison linéaire** des vecteurs colonnes $M_1 = \begin{pmatrix} a \\ c \end{pmatrix}$ et $M_2 = \begin{pmatrix} b \\ d \end{pmatrix}$ composant la matrice $M = (M_1\ M_2) = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$

Autrement dit si on fait le produit de M par le vecteur colonne $\begin{pmatrix} x' \\ y' \end{pmatrix}$ on obtiendra un nouveau vecteur colonne $\begin{pmatrix} x" \\ y" \end{pmatrix}$ et **les vecteurs colonnes de la matrice** $M \times M = M^2$ **sont les vecteurs d'une combinaison linéaire donnant** $\begin{pmatrix} x" \\ y" \end{pmatrix}$ **dont les coefficients sont** x **et** y

$\begin{pmatrix} x" \\ y" \end{pmatrix} = xM_1^2 + yM_2^2$

Vérifier que $M^2 = (M(M_1)\ M(M_2))$

5. D'une certaine manière $V_{n+1} = AV_n$ nous fait penser à une suite géométrique.On définit A^n par récurrence $A^n = AA^{n-1}$ avec $A^2 = AA$

   Donner l'expression de V_n en fonction de V_0
6. Utiliser la bibliothèque numpy de Python pour calculer V_10

Algorithme d'Euclide étendu On revient sur l'algorithme d'Euclide étendu en utilisant cette fois ci le calcul matriciel