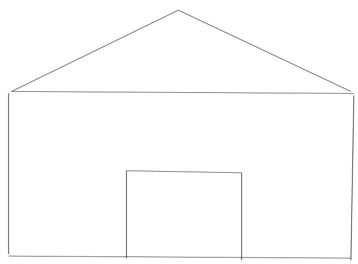
TP2: transformations affines

.

Partie 1:

Dans cette partie TP nous allons travailler avec l'image “vectorielle” suivante:



On représente la maison ci-dessus par la matrice suivante:

$$M = \begin{pmatrix} 0 & 1 & 1 & 2 & 2 & 1 & 3 & 3 & 1,5 & 0 & 3 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 2 & 3 & 2 & 2 & 2 & 0 \end{pmatrix}$$

Dans cette matrice la première ligne (**M[0,:]** en python) contient les abscisses des extrémités des segments représentant la maison et la deuxième ligne (**M[1,:]**) contient les ordonnées. Les coordonnées de certains points apparaissent plusieurs fois car on veut tracer la maison avec un seul “nuage de point”, c'est à dire avec une seule ligne brisée.
Les coordonnées sont déterminées dans un repère orthogonal qui est celui de la fonction **plot** de **pyplot**. Par défaut ce repère est situé en bas à gauche de l'image, l'axe des x est horizontal orienté de la gauche vers la droite, celui des y est vertical orienté du bas vers le haut. On peut fixer l'échelle grace au paramètre **axis**.
Le programme suivant permet de créer la matrice M et visualiser la maison:

```python

import numpy as np
import matplotlib.pyplot as plt
M=np.array([[0,1,1,2,2,1,3,3,1.5,0,3,0,0],[0,0,1,1,0,0,0,2,3,2,2,2,0]])
plt.plot(M[0,:],M[1,:])
plt.axis([0,8,0,8])
plt.show()

```

la bibliothèque **numpy** propose la fonction **dot** pour effectuer la multiplication des matrices: **np.dot(A,B)**
On peut aussi déterminer l'inverse d'une matrice **A** par **np.linalg.inv(A)**

1. Vérifiez, en recopiant le programme ci-dessus, que l'on obtient bien la maison.

   Dans la suite du TP nous allons appliquer différentes transformations affines sur notre “maison”. Nous utiliserons pour cela les coordonnées homogènes qui permettent une écriture simplifiée des transformations affines du plan. On notera MH la matrice de la “maison” en coordonnées homogènes.

2. Déterminez MH et donnez les commandes python pour créer cette matrice (pour créer MH à partir de M vous pouvez vous renseigner sur la fonction **concatenate** de numpy).

3. **Translations**:
Nous allons créer sous python une fonction **trans** permettant d'effectuer une translation d'une image "vectorielle" définie par sa matrice en coordonnées homogènes MH. Les entrées seront les coordonnées ux et uy du vecteur $\overrightarrow{u}$ de la translation et la matrice MH. La sortie sera la matrice RH en coordonnées homogènes de l'image de MH par la translation. La fonction affichera aussi MH et son image.

   (a) Déterminez en fonction de ux et uy la matrice T telle que $RH = T \times MH$ (le symbole $\times$ représente la produit matriciel)

   (b) Voici le code de la fonction **trans**. Completez les pointillés:

```
def trans(ux,uy,MH):
    T=.......................
    RH=......................
    plt.plot(MH[0,:],MH[1,:],RH[0,:],RH[1,:])
    plt.axis([0,8,0,8])
    plt.show()
    return RH
```

4. **Rotations:**

   (a) On considère une rotation de centre $C\begin{pmatrix} xc \\ xy \end{pmatrix}$ et d'angle a. Soit R la matrice de cette transformation en coordonnées homogènes. Ecrivez R comme le produit de 3 matrices $R = t1 \times ro \times t2$ où $t1$ et $t2$ sont des matrices de translations et où ro est une matrice d'une rotation de centre O . On ne demande pas d'effectuer le produit mais uniquement de déterminez $t1, t2$ et ro en fonction des coordonnées de C et de l'angle a .

   (b) En vous inspirant du programme de la translation de la question 3, programmez la fonction **rot** qui admet pour entrées les coordonnées xc et yc du centre de la rotation, l'angle a en radian et la matrice MH de l'image à transformer. Comme pour la translation la sortie sera la matrice RH en coordonnées homogènes de l'image de MH par la rotation et la fonction affichera à la fois MH et RH.

   (c) appliquez à l'image MH une translation de vecteur $\overrightarrow{u}\begin{pmatrix} 3 \\ 3 \end{pmatrix}$ suivi d'une rotation d'angle $\pi/3$ (**np.pi/3**) et de centre $C\begin{pmatrix} 4 \\ -4 \end{pmatrix}$

5. **Homothéties:**

   (a) programmez la fonction **hom(xc,yc,k,MH)** où xc et yc sont les coordonnées du centre de l'homothétie, k le rapport et MH la matrice de l'image à transformer.

6. **Dilatations:**

Pour définir une dilatation affine il faut disposer d'une droite de base $H : ax + by + c = 0$, d'une direction donnée par la droite $D : dx + ey + f = 0$ et d'un rapport $k \in \mathbb{R}$.

   (a) déterminez en fonction de a, b et c les coordonnées d'un point A de la droite H et les coordonnées d'un vecteur directeur $\overrightarrow{u}$.

   (b) déterminez en fonction de d, e et f les coordonnées d'un vecteur directeur $\overrightarrow{v}$ de la droite D.

   (c) Donnez une condition nécessaire et suffisante pour que les droites H et D permettent de définir une dilatation affine.

   (d) On note K la matrice en coordonnées homogènes de la dilatation de base H parallèlement à D et de rapport k. Décomposez K en un produit de 5 matrices "élémentaires".

   (e) Programmez la fonction **dilat** qui permet de réaliser une dilatation d'une image vectorielle définie par sa matrice MH en coordonnées homogènes.

Partie 2:

Dans cette partie on travaille à partir de la matrice MH en coordonnées homogènes de l'exercice précédent.

1. On considère l'application affine f dont la partie linéaire a pour matrice $Mat\left(\overrightarrow{f}\right) = \begin{pmatrix} 1 & 0,5 \\ 0 & 1 \end{pmatrix}$ et qui vérifie de plus $f(O) = \begin{pmatrix} 4 \\ 3 \end{pmatrix}$. Donnez la matrice de f en coordonnées homogènes et représentez l'image de la maison. Vous pouvez programmez une fonction **transfo(Mat,MH)** qui admet pour entrées la matrice Mat de l'application affine en coordonnées homogènes et la matrice MH de la ligne brisée à transformer.

2. Représentez l'image obtenue par l'application affine g dont la partie linéaire a pour matrice $Mat\left(\overrightarrow{g}\right) = \begin{pmatrix} 1 & -3 \\ 2 & -6 \end{pmatrix}$ et qui vérifie de plus $g(O) = O$:

3. Démontrez que l'application affine g de l'exercice précédent n'est pas une bijection.

4. Quel est l'ensemble des antécédents du point $O\begin{pmatrix} 0 \\ 0 \end{pmatrix}$ par l'application affine g ?

5. Soit l'application affine h définie par $\left\{ \begin{array}{l} x' = 3x - 2y + 5 \\ y' = 2x - y - 1 \end{array} \right.$

   (a) Donnez la matrice $Mat(h)$ en coordonnées homogènes de l'application affine h .

   (b) Démontrez que l'application affine h est une bijection du plan affine.

   (c) Donnez la définition exacte de l'application affine réciproque de h, que l'on notera h^{-1} .
   Quelle est la matrice, en coordonnées homogènes, de l'application affine h^{-1} ?

   (d) Vérifiez avec Python qu'en appliquant h puis h^{-1} à la maison MH on retrouve MH .

Partie 3:

Coordonnées homogènes dans l'espace affine:

En dimension 3, c'est à dire dans l'espace affine, par analogie avec la dimension 2, on dira que $\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$ sont les coordonnées homogènes du point $M\begin{pmatrix} x \\ y \\ z \end{pmatrix}$ (où $\begin{pmatrix} x \\ y \\ z \end{pmatrix}$ sont les coordonnées du point M dans un repère orthonormé direct $R = \left(O, \overrightarrow{i}, \overrightarrow{j}, \overrightarrow{k}\right)$).
Plus généralement $\begin{pmatrix} X \\ Y \\ Z \\ H \end{pmatrix}$ sont des coordonnées homogènes du point $M\begin{pmatrix} x \\ y \\ z \end{pmatrix}$ si $H \neq 0$ et si $x = \frac{X}{H}$, $y = \frac{Y}{H}$, $z = \frac{Z}{H}$. Par exemple $\begin{pmatrix} 1 \\ 2 \\ 3 \\ 1 \end{pmatrix}$, $\begin{pmatrix} 2 \\ 4 \\ 6 \\ 2 \end{pmatrix}$ et $\begin{pmatrix} 10 \\ 20 \\ 30 \\ 10 \end{pmatrix}$ sont des coordonnées homogènes du même point M de coordonnées $\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$ dans le repère R.

Les transformations de l'espace se modéliseront à l'aide de matrices 4x4 en coordonnées homogènes.

Par analogie avec les transformations affines du plan, la matrice en coordonnées homogènes d'un transformation affine de l'espace f aura la forme suivante:

$$\begin{pmatrix} a & a' & a" & u \\ b & b' & b" & v \\ c & c' & c" & w \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

avec $f\left(\overrightarrow{i}\right)=\begin{pmatrix} a \\ b \\ c \end{pmatrix}, f\left(\overrightarrow{j}\right)=\begin{pmatrix} a' \\ b' \\ c' \end{pmatrix}, f\left(\overrightarrow{k}\right)=\begin{pmatrix} a" \\ b" \\ c" \end{pmatrix}$ et $f(O)=\begin{pmatrix} u \\ v \\ w \end{pmatrix}$

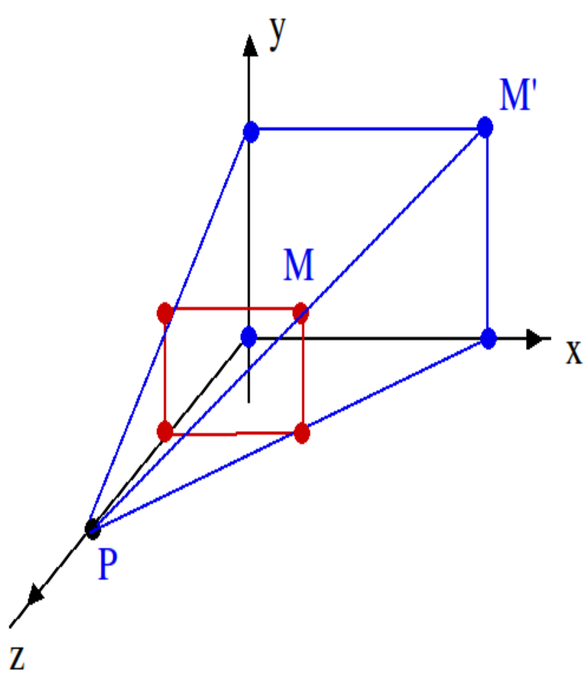
Exercice 3-A:

1. Donnez la matrice d'une homothétie de centre O et de rapport 2.
2. D'une rotation d'axe Oz et d'angle $\frac{\pi}{6}$.
3. D'une translation de vecteur $\overrightarrow{u}\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$.
4. D'une homothétie de rapport 2 et de centre $A\begin{pmatrix} -1 \\ 2 \\ 1 \end{pmatrix}$.

Projections 3D-2D :

Un objet réel (donc de dimension 3) est représenté par une image en 2 dimensions sur l'écran d'un ordinateur. Il faut donc, pour pouvoir représenter un objet de dimension 3 sur un écran de dimension 2, définir une application de l'espace affine vers le plan affine. Si on veut que l'image résultante soit réaliste, c'est à dire si on veut qu'elle reproduise le travail de l'oeil humain, cette application ne sera pas affine, donc pas représentable par un produit matriciel classique. Pourtant nous allons voir, que grâce aux coordonnées homogènes, on peut représenter une telle application par un produit matriciel.

L'objet "réel" sera un polyèdre de l'espace affine représenté par une matrice contenant les coordonnées homogènes de ses sommets dans un repère orthonormé direct $R=\left(O,\overrightarrow{i},\overrightarrow{j},\overrightarrow{k}\right)$. Pour simplifier, l'écran de l'ordinateur sera le plan affine Oxy (c'est à dire le plan passant par O et engendré par les vecteurs $\overrightarrow{i}$ et $\overrightarrow{j}$), et l'œil de l'observateur se situera sur l'axe des z au point $P\begin{pmatrix} 0 \\ 0 \\ d \end{pmatrix}$ (cf figure ci-dessous).



Un point M de coordonnées $\begin{pmatrix} x \\ y \\ z \end{pmatrix}$ sera représentée sur l'écran (Oxy) par un point M' de coordonnées $\begin{pmatrix} x' \\ y' \end{pmatrix}$ avec, d'après le théorème de Thalès:

$$\frac{x'}{x}=\frac{d}{d-z} \Leftrightarrow x'=\frac{x}{1-\frac{z}{d}}$$

et de même

$$y' = \frac{y}{1 - \frac{z}{d}}$$

Avec les coordonnées homogènes on peut représenter la projection précédente par le produit matriciel suivant:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{1}{d} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ 1 - \frac{z}{d} \end{pmatrix}$$

Exercice 3-B:

On considère le cube OABCDEFG définit par les coordonnées suivantes :

$$O\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}; A\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}; B\begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}; C\begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix}; D\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}; E\begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}; F\begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix}; G\begin{pmatrix} 0 \\ -1 \\ 1 \end{pmatrix}$$

Pour créer sous python la matrice K permettant de définir en coordonnées homogènes le cube OABCDEFG on peut écrire les lignes de code suivantes:

```python

import numpy as np
import matplotlib.pyplot as plt
#points du cube en coordonnées homogènes:
O=np.array([[0],[0],[0],[1]])
A=np.array([[1],[0],[0],[1]])
B=np.array([[1],[-1],[0],[1]])
C=np.array([[0],[-1],[0],[1]])
D=np.array([[0],[0],[1],[1]])
E=np.array([[1],[0],[1],[1]])
F=np.array([[1],[-1],[1],[1]])
G=np.array([[0],[-1],[1],[1]])

#Matrice pour le tracé du cube:
K=np.concatenate((O,A,B,C,O,D,E,A,E,F,B,F,G,C,G,D,O,B,A,C),axis=1)
```

1. On considère que l'observateur est en $P\begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix}$. Déterminer la matrice K' des coordonnées homogènes de la projection du cube sur l'écran de l'ordinateur, puis tracer une représentation de la projection de ce cube. Vous utiliserez un programme en Python pour réaliser ce calcul et l'affichage de cette image.

2. On considère à présent que l'observateur est en $P\begin{pmatrix} 0 \\ 0 \\ 4 \end{pmatrix}$ et que le cube effectue une translation de vecteur $\overrightarrow{u}\begin{pmatrix} 1 \\ -0,5 \\ 0 \end{pmatrix}$ puis une rotation d'angle $\frac{2\pi}{3}$ par rapport à l'axe des x. Représentez la projection du cube résultant sur l'écran de l'ordinateur.