

```

"""@author: campioni
Cap 4
"""
#EQUAZIONI DI LORENZ
#USO DEL METODO ITERATIVO PER IL CALCOLO DELLE INTERSEZIONI DELLE
#CURVE SUL PIANO XY
import matplotlib.pyplot as plt
from pylab import figure, show, setp
from mpl_toolkits.mplot3d import Axes3D
print()
print('Risoluzione e rappresentazione delle equazioni di Lorenz')
print()
t=0
dt=0.01 # imposta il passo d'iterazione
n=int(input('numero massimo dei passi (Es. 100000) = '))
print()
tmax=n*dt

x,y,z =.5 ,.5 ,.5

A , B, C = float(10), float(28), float(8/3) # valori scelti da Lorenz
T=[0] # inizializza la lista dei tempi da 0 a tmax
X=[0] # inizializza la lista dei valori di x
Y=[0]
Z=[0]
while t <= tmax: # calcola t, x , y ,z tramite le equazioni di Lorenz
    t += dt
    T.append(t)
    dx=A*(y-x)*dt
    dy=(B*x-y-x*z)*dt
    dz=(x*y-C*z)*dt
    x +=dx
    y +=dy
    z +=dz
    X.append(x)
    Y.append(y)

print('-Ingrandimento degli assi per aumentarne la risoluzione-')
print()

x1=float(input("Immetti l'estremo inferiore x1_ "))
x2=float(input("Immetti l'estremo superiore x2_ "))
y1=float(input("Immetti l'estremo inferiore y1_ "))
y2=float(input("Immetti l'estremo superiore y2_ "))

plt.subplots(facecolor='olive')
plt.rcParams['axes.facecolor'] = 'white'
plt.title("ATTRATTORE DI LORENZ Tracce delle curve sull'asse xy") # diagramma
x , y
plt.xlabel(' x=volume di rotazione cilindro')
plt.ylabel('y= delta Temperatura')
plt.grid()
plt.axvline(x=0 , color= 'r')
plt.axis([x1,x2,y1,y2])
plt.scatter(X,Y, s=.5)
plt.show()

```