3.1)

$$\begin{cases} \dot{x} = \sigma(y-x) \\ \dot{y} = rx - y - xz \\ \dot{z} = xy - bz \end{cases} \quad \begin{cases} \sigma = 10 \\ b = 8/3 \\ r = 28 \end{cases} \Rightarrow \begin{cases} \dot{x} = 10(y-x) & (f) \\ \dot{y} = 28x - y - xz & (g) \\ \dot{z} = xy - \frac{8}{3}z & (h) \end{cases}$$

a)

F.P.S. at $\dot{x} = \dot{y} = \dot{z} = 0$

$$\Rightarrow \begin{cases} 0 = 10(y-x) & (1) \\ 0 = 28x - y - xz & (2) \\ 0 = xy - \frac{8}{3}z & (3) \end{cases}$$

(1): $y = x$ (4)

(4) into (3): $0 = y^2 - \frac{8}{3}z \Rightarrow z = \frac{3y^2}{8}$ (5)

(4), (5) into (2): $0 = 28y - y - y\left(\frac{3y^2}{8}\right) = 27y - y\left(\frac{3y^2}{8}\right)$

$\Rightarrow 0 = \underbrace{y}_{0}\underbrace{\left(27 - \frac{3}{8}y^2\right)}_{\text{or } 0}$ $(y_1 = 0)$

$\Rightarrow 27 - \frac{3}{8}y^2 = 0 \Rightarrow \frac{3}{8}y^2 = 27 \Rightarrow y^2 = 72 \Rightarrow y = \pm\sqrt{72}$

$\Rightarrow \quad y_1 = 0 \quad , \quad y_2 = -\sqrt{72} \quad , \quad y_3 = \sqrt{72}$

$\underline{y=0}: \Rightarrow \quad x = 0 \Rightarrow \quad z = 0$

One F.P. at $(0,0,0)^T$

$\underline{y = -\sqrt{72}}: \Rightarrow \quad x = -\sqrt{72}$

x, y into (3):

$0 = (-\sqrt{72})(-\sqrt{72}) - \frac{8}{3}z \Rightarrow \frac{8}{3}z = 72 \Rightarrow z = 27$

one F.P. at $(-\sqrt{72}, -\sqrt{72}, 27)^T$

$\underline{y = \sqrt{72}}: \Rightarrow \quad x = \sqrt{72}$

x, y into (3):

$0 = \sqrt{72}\sqrt{72} - \frac{8}{3}z \Rightarrow z = 27$

one F.P. at $(\sqrt{72}, \sqrt{72}, 27)^T$

Find eigenvalues of stability matrix to determine type of stability:

$$J = \begin{pmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} & \frac{\partial f}{\partial z} \\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial y} & \frac{\partial g}{\partial z} \\ \frac{\partial h}{\partial x} & \frac{\partial h}{\partial y} & \frac{\partial h}{\partial z} \end{pmatrix} = \begin{pmatrix} -10 & 10 & 0 \\ (28-z) & -1 & -x \\ y & x & -\frac{8}{3} \end{pmatrix}$$

use mathematica...

Ans: 3 F.P.s and 0 stable F.P.s.
(See mathematica pdf for explanation).

3.1) c)

$$\begin{cases} \dot{x} = \sigma(y-x) & = f(x,y,z) \\ \dot{y} = rx - y - xz & = g(x,y,z) \\ \dot{z} = xy - bz & = h(x,y,z) \end{cases}$$

c)

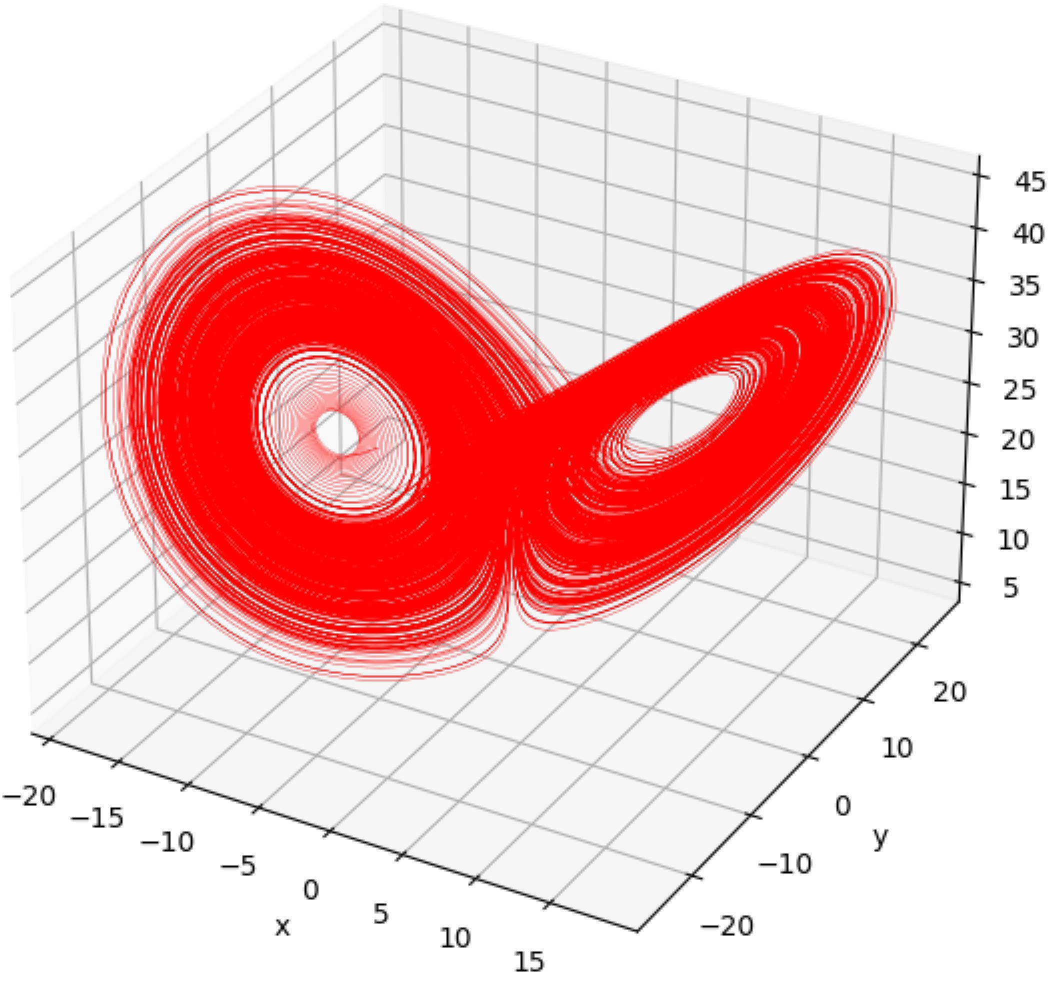
Ans:

$$\mathbb{J} = \begin{pmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} & \frac{\partial f}{\partial z} \\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial y} & \frac{\partial g}{\partial z} \\ \frac{\partial h}{\partial x} & \frac{\partial h}{\partial y} & \frac{\partial h}{\partial z} \end{pmatrix} = \begin{pmatrix} -\sigma & \sigma & 0 \\ (r-z) & -1 & -x \\ y & x & -b \end{pmatrix}$$

d) $\lambda_1 + \lambda_2 + \lambda_3 = tr\,\mathbb{J} = -\sigma - 1 - b$

Ans: $\lambda_1 + \lambda_2 + \lambda_3 = -\sigma - 1 - b$

$3.1b$



```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import odeint
import sys
import os

xmin = -1
ymin = -1
zmin = -1
xmax = 1
ymax = 1
zmax = 1

# Streamplot
# no_points = 100
# x_points = np.linspace(xmin, xmax, no_points)
# y_points = np.linspace(ymin, ymax, no_points)
# z_points = np.linspace(zmin, zmax, no_points)
# X, Y, Z = np.meshgrid(x_points, y_points)

# dxdt_streamplot = mu*X + Y - X**2
# dydt_streamplot = -X + mu*Y + 2*X**2
# dzdt_streamplot =

# Numerical integration
T = 500
t = np.linspace(0,T,T*100)
x = np.zeros(T)
y = x.copy()
z = x.copy()
# fp = np.array([[(1+(mu**2))/(2+mu), (-(2*mu-1)*(1+mu**2))/((2+mu)**2)], [0,0]])
# x[0] = fp[0,0] - 0.01
# y[0] = fp[0,1] - 0.01
x[0] = 0.01
y[0] = 0.01
z[0] = 0.01

def dynamical_system(xyz, t):
    x = xyz[0]
    y = xyz[1]
    z = xyz[2]
    dxdt_integration = 10*(y-x)
    dydt_integration = 28*x-y-x*z
    dzdt_integration =  x*y-(8/3)*z
    return [dxdt_integration, dydt_integration, dzdt_integration]

x0y0z0 = [x[0],y[0],z[0]]
xyz = odeint(dynamical_system, x0y0z0, t)
x = xyz[:,0]
y = xyz[:,1]
z = xyz[:,2]

fig, ax = plt.subplots(subplot_kw={"projection": "3d"}, figsize=(7,7))
# ax.streamplot(X, Y, dxdt_streamplot, dydt_streamplot, density = 2)
ax.plot(x[100:], y[100:], z[100:], '-', color='red', linewidth=0.25)
# ax.plot(fp[0,0],fp[0,1], '.', color='black', markersize=15, label='Saddle node')
# ax.plot(fp[1,0],fp[1,1], '.', color='magenta', markersize=15, label='Unstable
spiral')
ax.set_title('$3.1b$')
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')
# ax.set_xlim(xmin,xmax)
# ax.set_ylim(ymin,ymax)
# ax.set_ylim(zmin,zmax)
# ax.set_box_aspect(1)

# plt.legend(loc="upper left")
script_dir = os.path.dirname(__file__)
results_dir = os.path.join(script_dir, '3.1/')
plt.savefig('Dynamical systems/DS HW3/3.1/3.1b.png', bbox_inches='tight')
plt.show()
```

```
(*3.1a*)
ClearAll["Global`*"];
x = 0;
y = 0;
z = 0;

x = √72;
y = √72;
z = 27;

x = -√72;
y = -√72;
z = 27;

m = Eigenvalues[{{-10, 10, 0}, {28 - z, -1, -x}, {y, x, -8/3}}] // MatrixForm;
m // FullSimplify;
```

$$(*fp1= \begin{pmatrix} \frac{1}{2}\left(-11-\sqrt{1201}\right) \\ \frac{1}{2}\left(-11+\sqrt{1201}\right) \\ -\frac{8}{3} \end{pmatrix}$$

$$fp2= \begin{pmatrix} -13.9\ldots \\ 0.0940\ldots+10.2\ldots\, i \\ 0.0940\ldots-10.2\ldots\, i \end{pmatrix}$$

$$fp3= \begin{pmatrix} -13.9\ldots \\ 0.0940\ldots+10.2\ldots\, i \\ 0.0940\ldots-10.2\ldots\, i \end{pmatrix}$$

```
 For a fixed point to be stable,
all eigenvalues must be strictly smaller than 0. For all three fixed points
 this is not the case. Thus, there are no fixed points in this system.*)
```

In[ ]:=

```
(*3.1d*)
ClearAll["Global`*"];
m = Eigenvalues[{{-σ, σ, 0}, {r - z, -1, -x}, {y, x, -b}}] // MatrixForm;
m // FullSimplify
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} \text{Root}\left[144\,\sigma + 28\,b\,\sigma - b\,r\,\sigma + (72 + b + 28\,\sigma + b\,\sigma - r\,\sigma)\,\#1 + (1 + b + \sigma)\,\#1^2 + \#1^3\,\&,\ 1\right] \\ \text{Root}\left[144\,\sigma + 28\,b\,\sigma - b\,r\,\sigma + (72 + b + 28\,\sigma + b\,\sigma - r\,\sigma)\,\#1 + (1 + b + \sigma)\,\#1^2 + \#1^3\,\&,\ 2\right] \\ \text{Root}\left[144\,\sigma + 28\,b\,\sigma - b\,r\,\sigma + (72 + b + 28\,\sigma + b\,\sigma - r\,\sigma)\,\#1 + (1 + b + \sigma)\,\#1^2 + \#1^3\,\&,\ 3\right] \end{pmatrix}$$

```
Integrate[]
```