1.

$$\frac{dc(S)}{dt} = k2 * c(ES) - k1 * c(E) * c(S)$$
$$\frac{dc(E)}{dt} = (k2 + k3) * c(ES) - k1 * c(E) * c(S)$$
$$\frac{dc(ES)}{dt} = k1 * c(E) * c(S) - (k2 + k3) * c(ES)$$
$$\frac{dc(P)}{dt} = k3 * c(ES)$$

2.fourth-order Runge-Kutta method by Python：

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

# define each rate of change
def dx(x, y, z, w):
    return 600*z - 100*x*y

def dy(x, y, z, w):
    return 750*z - 100*x*y

def dz(x, y, z, w):
    return 100*x*y - 750*z

def dw(x, y, z, w):
    return 150*z

# fourth-order Runge-Kutta
def runge_kutta_4(t0, x0, y0, z0, w0, h, n):
    t = t0
    x = x0
    y = y0
    z = z0
    w = w0
    for i in range(n):
        k1x = h * dx(x, y, z, w)
        k1y = h * dy(x, y, z, w)
        k1z = h * dz(x, y, z, w)
        k1w = h * dw(x, y, z, w)
        k2x = h * dx(x + k1x/2, y + k1y/2, z + k1z/2, w + k1w/2)
        k2y = h * dy( x + k1x/2, y + k1y/2, z + k1z/2, w + k1w/2)
        k2z = h * dz( x + k1x/2, y + k1y/2, z + k1z/2, w + k1w/2)
        k2w = h * dw( x + k1x/2, y + k1y/2, z + k1z/2, w + k1w/2)
```

```
        k3x = h * dx( x + k2x/2, y + k2y/2, z + k2z/2, w + k2w/2)
        k3y = h * dy( x + k2x/2, y + k2y/2, z + k2z/2, w + k2w/2)
        k3z = h * dz( x + k2x/2, y + k2y/2, z + k2z/2, w + k2w/2)
        k3w = h * dw( x + k2x/2, y + k2y/2, z + k2z/2, w + k2w/2)
        k4x = h * dx( x + k3x, y + k3y, z + k3z, w + k3w)
        k4y = h * dy( x + k3x, y + k3y, z + k3z, w + k3w)
        k4z = h * dw(x + k3x, y + k3y, z + k3z, w + k3w)
        k4w = h * dz( x + k3x, y + k3y, z + k3z, w + k3w)
        x += (k1x + 2*k2x + 2*k3x + k4x)/6
        y += (k1y + 2*k2y + 2*k3y + k4y)/6
        z += (k1z + 2*k2z + 2*k3z + k4z)/6
        w += (k1w + 2*k2w + 2*k3w + k4w)/6
        t += h

    return t, x, y, z, w
```

3. use the same method to get the plot:

```
t = 0
x = 10
y = 1
z = 0
w = 0
h=0.001
fig, ax = plt.subplots()
for i in range(90):
    ax.plot(x, 150*z,'ro')
    k1x = h * dx(x, y, z, w)
    k1y = h * dy(x, y, z, w)
    k1z = h * dz(x, y, z, w)
    k1w = h * dw(x, y, z, w)
    k2x = h * dx(x + k1x/2, y + k1y/2, z + k1z/2, w + k1w/2)
    k2y = h * dy( x + k1x/2, y + k1y/2, z + k1z/2, w + k1w/2)
    k2z = h * dz( x + k1x/2, y + k1y/2, z + k1z/2, w + k1w/2)
    k2w = h * dw( x + k1x/2, y + k1y/2, z + k1z/2, w + k1w/2)
    k3x = h * dx( x + k2x/2, y + k2y/2, z + k2z/2, w + k2w/2)
    k3y = h * dy( x + k2x/2, y + k2y/2, z + k2z/2, w + k2w/2)
    k3z = h * dz( x + k2x/2, y + k2y/2, z + k2z/2, w + k2w/2)
    k3w = h * dw( x + k2x/2, y + k2y/2, z + k2z/2, w + k2w/2)
    k4x = h * dx( x + k3x, y + k3y, z + k3z, w + k3w)
    k4y = h * dy( x + k3x, y + k3y, z + k3z, w + k3w)
    k4z = h * dw(x + k3x, y + k3y, z + k3z, w + k3w)
```

```
        k4w = h * dz( x + k3x, y + k3y, z + k3z, w + k3w)
        x += (k1x + 2*k2x + 2*k3x + k4x)/6
        y += (k1y + 2*k2y + 2*k3y + k4y)/6
        z += (k1z + 2*k2z + 2*k3z + k4z)/6
        w += (k1w + 2*k2w + 2*k3w + k4w)/6
        t += h
plt.show()
```

As seen in figure, Vm is approximately 120