> #Okay to post

> #Anusha Nagar, Homework 20, November 13, 2021

>

> **read** "C://Users/an646/Documents/DMB.txt"

First Written: Nov. 2021

This is DMB.txt, A Maple package to explore Dynamical models in Biology (both discrete and continuous)
accompanying the class Dynamical Models in Biology, Rutgers University. Taught by Dr. Z. (Doron Zeilbeger)

The most current version is available on WWW at:
http://sites.math.rutgers.edu/~zeilberg/tokhniot/DMB.txt .
Please report all bugs to: DoronZeil at gmail dot com .

For general help, and a list of the MAIN functions,
type "Help();". For specific help type "Help(procedure_name);"

For a list of the supporting functions type: Help1();
For help with any of them type: Help(ProcedureName);

For a list of the functions that give examples of Discrete-time dynamical systems (some famous), type: HelpDDM();
For help with any of them type: Help(ProcedureName);

For a list of the functions continuous-time dynamical systems (some famous) type: HelpCDM();
For help with any of them type: Help(ProcedureName);

------------------------------ **(1)**

> *Help*( )

DMB.txt: A Maple package for exploring Dynamical models in Biology
The MAIN procedures are
ComK, Dis, EquP, FP, Orb, OrbF, Orbk, PhaseDiag, SEquP, SFP, TimeSeries **(2)**

> *Help*(*SIRS*)

SIRS(s,i,beta,gamma,nu,N): The SIRS dynamical model with parameters beta,gamma, nu,N (see section 6.6 of Edelstein-Keshet), s is the number of Susceptibles, i is the number of infected, (the number of removed is given by N-s-i). N is the total population. Try:

SIRS(s,i,beta,gamma,nu,N); **(3)**

> Help(EquP)

EquP(F,x): Given a transformation F in the list of variables finds all the Equilibrium points of the continuous-time dynamical system x'(t)=F(x(t))

EquP([5/2*x*(1-x)],[x]);

EquP([y*(1-x-y),x*(3-2*x-y)],[x,y]); **(4)**

> Help(SEquP)

SEquP(F,x): Given a transformation F in the list of variables finds all the Stable Equilibrium points of the continuous-time dynamical system x'(t)=F(x(t))

SEquP([5/2*x*(1-x)],[x]);

SEquP([y*(1-x-y),x*(3-2*x-y)],[x,y]); **(5)**

> Help(TimeSeries)

TimeSeries(F,x,pt,h,A,i): Inputs a transformation F in the list of variables x

The time-series of x[i] vs. time of the Dynamical system approximating the the autonomous continuous dynamical process

dx/dt=F(x(t)) by a discrete time dynamical system with step-size h from t=0 to t=A

Try:

TimeSeries([x*(1-y),y*(1-x)],[x,y],[0.5,0.5], 0.01, 10,1); **(6)**

> Help(PhaseDiag)

PhaseDiag(F,x,pt,h,A): Inputs a transformation F in the list of variables x (of length 2), i.e. a mapping from R^2 to R^2 gives the

The phase diagram of the solution with initial condition x(0)=pt

dx/dt=F[1](x(t)) by a discrete time dynamical system with step-size h from t=0 to t=A

Try:

PhaseDiag([x*(1-y),y*(1-x)],[x,y],[0.5,0.5], 0.01, 10); **(7)**

> #Problem 1

>

> #Part (i)

> EquP(SIRS(s, i, beta, gamma, nu, N), [s, i])

$$\left\{[N, 0], \left[\frac{\nu}{\beta}, \frac{\gamma(N\beta-\nu)}{\beta(\gamma+\nu)}\right]\right\} \quad (8)$$

> SEquP(SIRS(s, i, beta, gamma, nu, N), [s, i])

```
Error, (in SEquP) cannot determine if this expression is true or
false: max(-.5772156649, N*beta-1.*nu) < 0
|C:/Users/an646/Documents/DMB.txt:639|
```

>

> $F\_i1 := SIRS\left(s, i, \frac{0.3 \cdot 2}{1000}, 5, 2, 1000\right)$

$$F\_i1 := [-0.0006000000000\, s\, i + 5000 - 5\, s - 5\, i, 0.0006000000000\, s\, i - 2\, i] \quad \textbf{(9)}$$

> $F\_i2 := SIRS\left(s, i, \frac{0.9 \cdot 2}{1000}, 5, 2, 1000\right)$

$$F\_i2 := [-0.001800000000\, s\, i + 5000 - 5\, s - 5\, i, 0.001800000000\, s\, i - 2\, i] \quad \textbf{(10)}$$

> $F\_i3 := SIRS\left(s, i, \frac{3.9 \cdot 2}{1000}, 5, 2, 1000\right)$

$$F\_i3 := [-0.007800000000\, s\, i + 5000 - 5\, s - 5\, i, 0.007800000000\, s\, i - 2\, i] \quad \textbf{(11)}$$

> $EquP(F\_i1, [s, i])$

$$\{[1000., 0.], [3333.333333, -1666.666667]\} \quad \textbf{(12)}$$

> $SEquP(F\_i1, [s, i])$

$$\{[1000., 0.]\} \quad \textbf{(13)}$$

> $EquP(F\_i2, [s, i])$

$$\{[1000., 0.], [1111.111111, -79.36507937]\} \quad \textbf{(14)}$$

> $SEquP(F\_i2, [s, i])$

$$\{[1000., 0.]\} \quad \textbf{(15)}$$

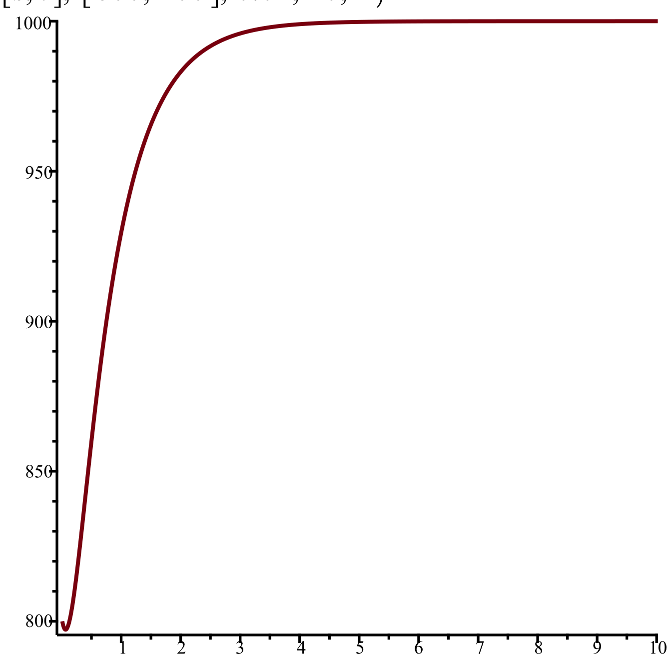
> $EquP(F\_i3, [s, i])$

$$\{[256.4102564, 531.1355311], [1000., 0.]\} \quad \textbf{(16)}$$

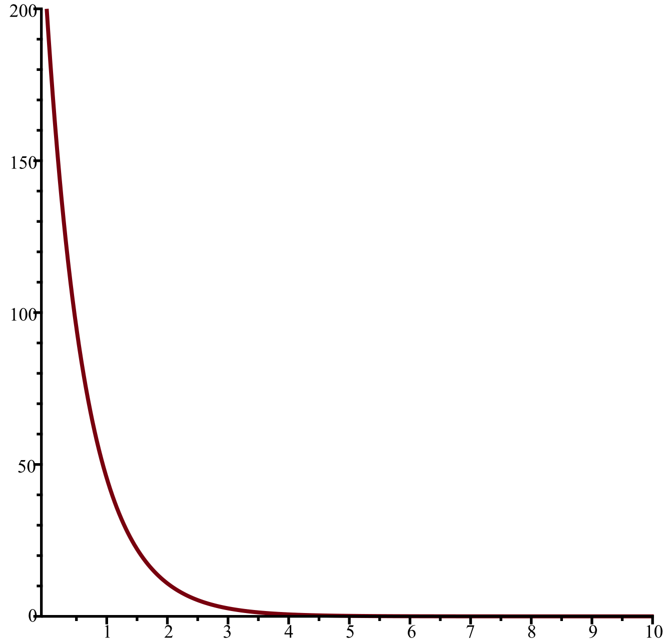
> $SEquP(F\_i3, [s, i])$

$$\{[256.4102564, 531.1355311]\} \quad \textbf{(17)}$$

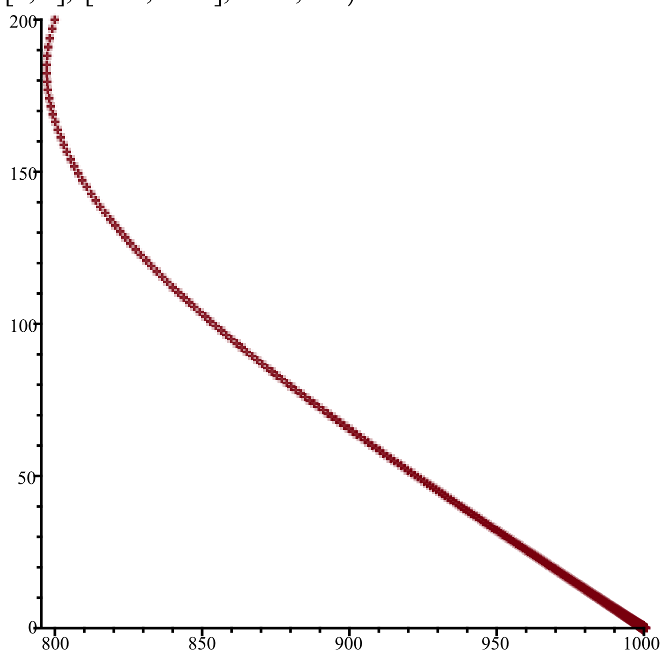
> $TimeSeries(F\_i1, [s, i], [800, 200], 0.01, 10, 1)$



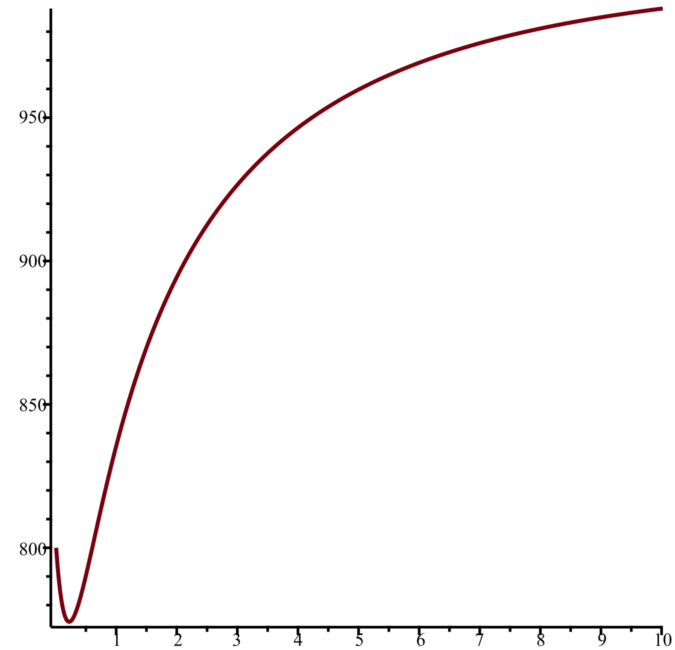
> $TimeSeries(F\_i1, [s, i], [800, 200], 0.01, 10, 2)$



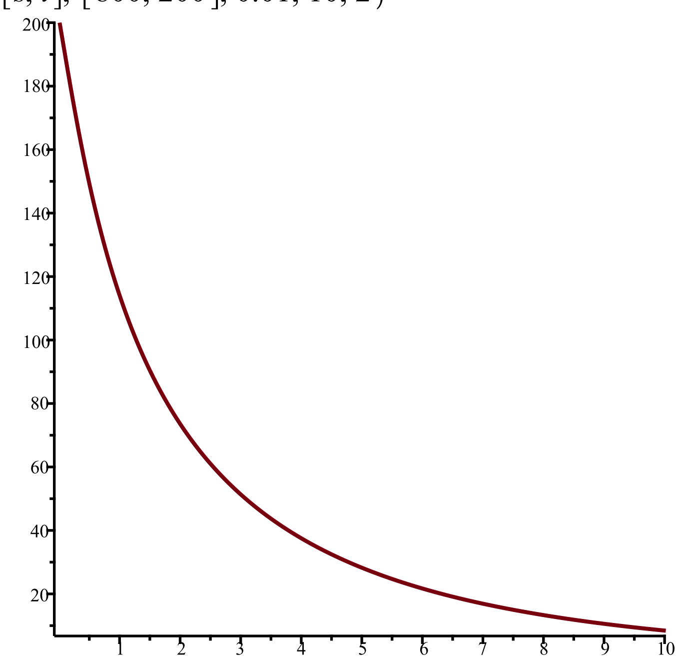
> *PhaseDiag*(*F_i1*, [*s*, *i*], [800, 200], 0.01, 10)



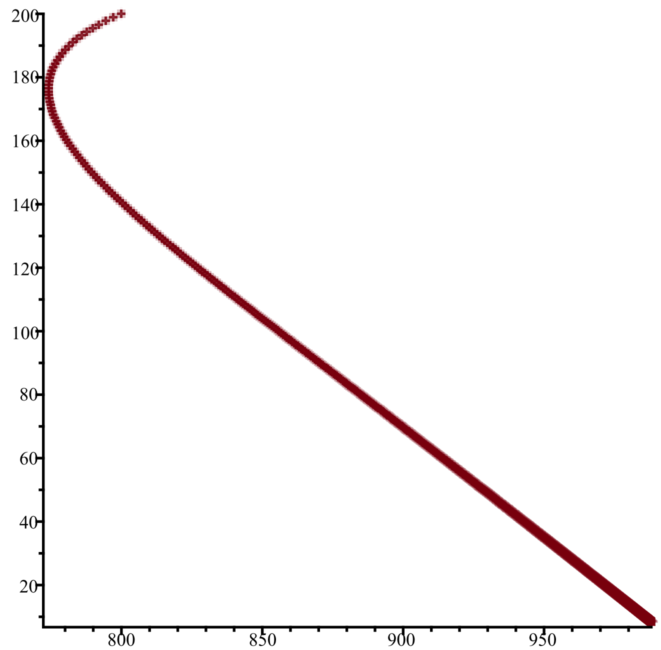
> *TimeSeries*(*F_i2*, [*s*, *i*], [800, 200], 0.01, 10, 1)



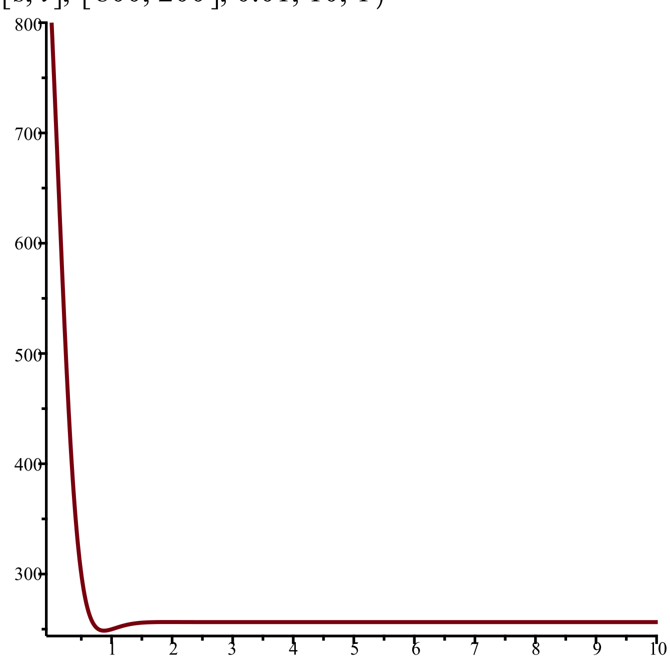
> *TimeSeries*(*F_i2*, [*s*, *i*], [800, 200], 0.01, 10, 2)



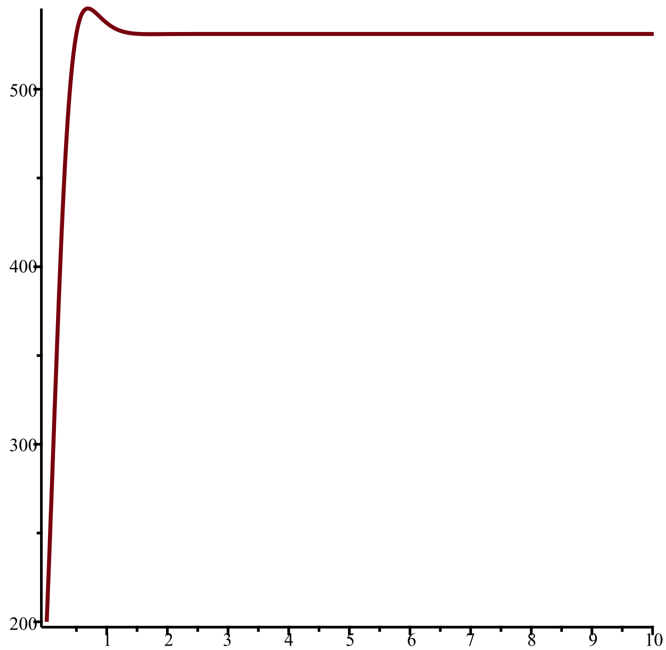
> *PhaseDiag*(*F_i2*, [*s*, *i*], [800, 200], 0.01, 10)



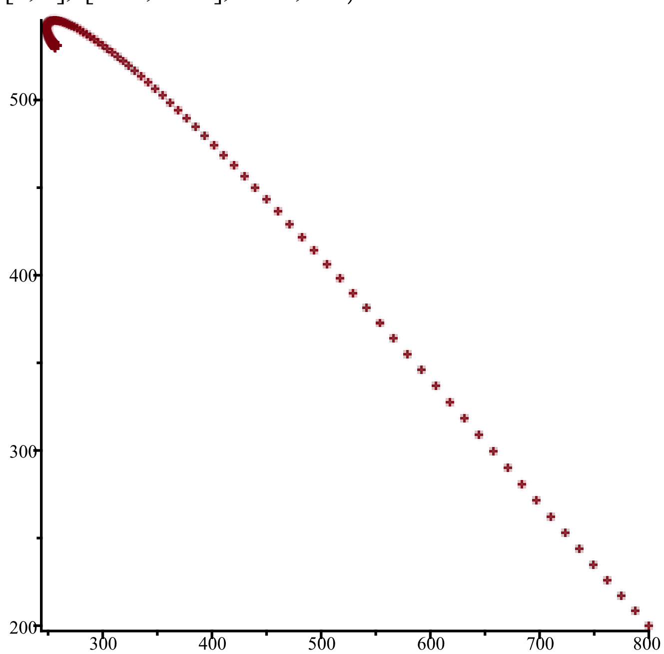
> *TimeSeries*(*F_i3*, [*s*, *i*], [800, 200], 0.01, 10, 1)



> *TimeSeries*(*F_i3*, [*s*, *i*], [800, 200], 0.01, 10, 2)



> $PhaseDiag(F\_i3, [s, i], [800, 200], 0.01, 10)$



> #Part (ii)

> $F\_ii1 := SIRS\left(s, i, \frac{0.3 \cdot 3}{1000}, 6, 2, 1000\right)$

$$F\_ii1 := [-0.0009000000000\, s\, i + 6000 - 6\, s - 6\, i,\ 0.0009000000000\, s\, i - 2\, i] \quad \textbf{(18)}$$

> $F\_ii2 := SIRS\left(s, i, \frac{0.9 \cdot 3}{1000}, 6, 2, 1000\right)$

$$F\_ii2 := [-0.002700000000\, s\, i + 6000 - 6\, s - 6\, i,\ 0.002700000000\, s\, i - 2\, i] \quad \textbf{(19)}$$

> $F\_ii3 := SIRS\left(s, i, \frac{3.9 \cdot 3}{1000}, 6, 2, 1000\right)$

$$F\_ii3 := [-0.01170000000\, s\, i + 6000 - 6\, s - 6\, i,\ 0.01170000000\, s\, i - 2\, i] \quad \textbf{(20)}$$

> $EquP(F\_ii1, [s, i])$

$$\{[1000., 0.], [2222.222222, -916.6666667]\} \quad (21)$$

> $SEquP(F\_ii1, [s, i])$

$$\{[1000., 0.]\} \quad (22)$$

> $EquP(F\_ii2, [s, i])$

$$\{[740.7407407, 194.4444444], [1000., 0.]\} \quad (23)$$

> $SEquP(F\_ii2, [s, i])$

$$\{[740.7407407, 194.4444444]\} \quad (24)$$

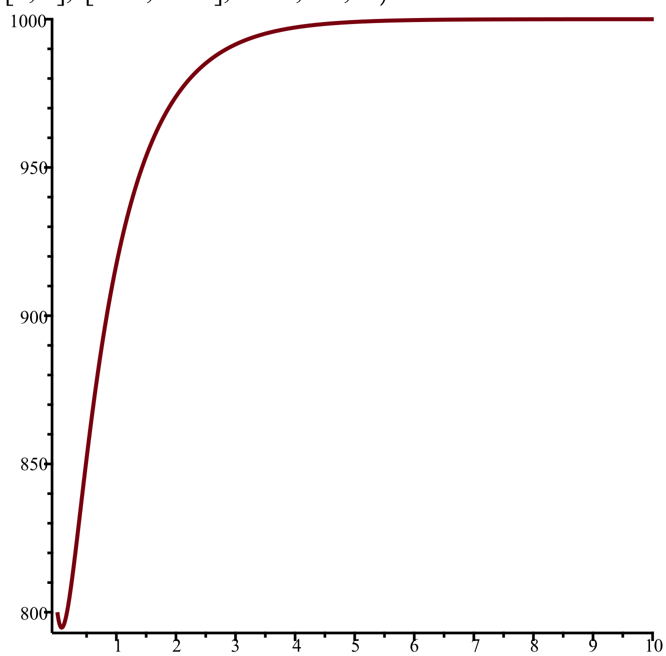
> $EquP(F\_ii3, [s, i])$

$$\{[170.9401709, 621.7948718], [1000., 0.]\} \quad (25)$$

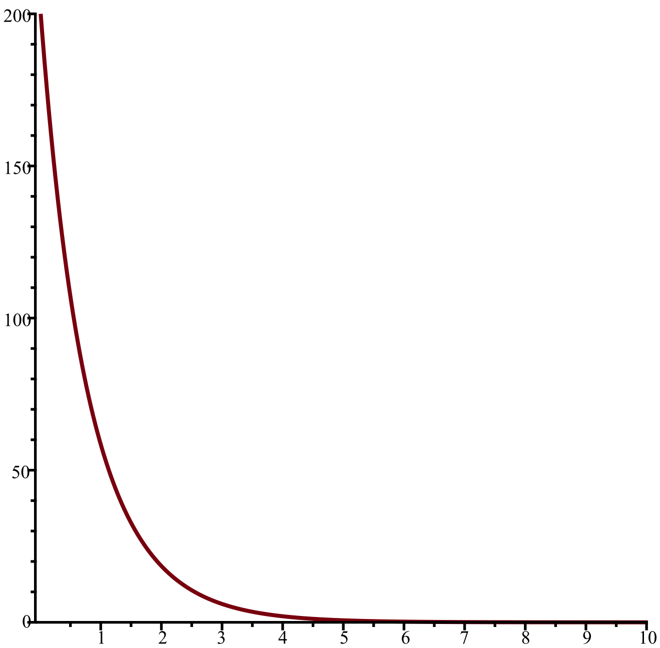
> $SEquP(F\_ii3, [s, i])$

$$\{[170.9401709, 621.7948718]\} \quad (26)$$

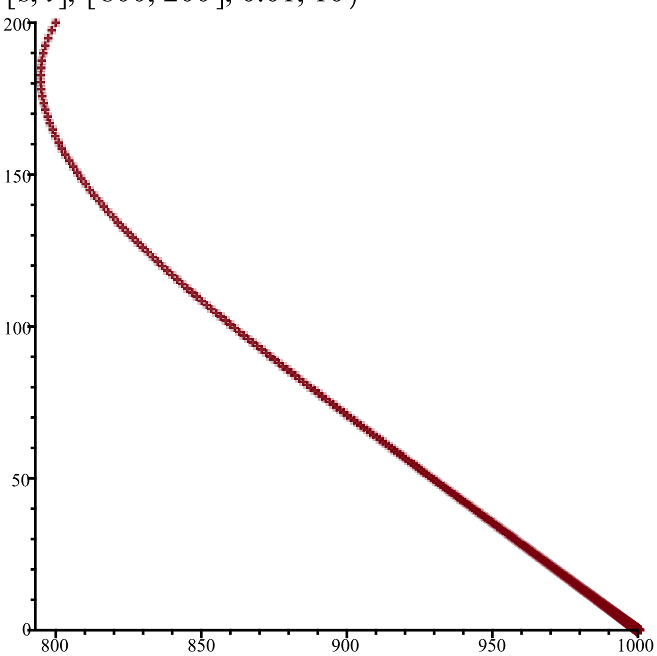
> $TimeSeries(F\_ii1, [s, i], [800, 200], 0.01, 10, 1)$



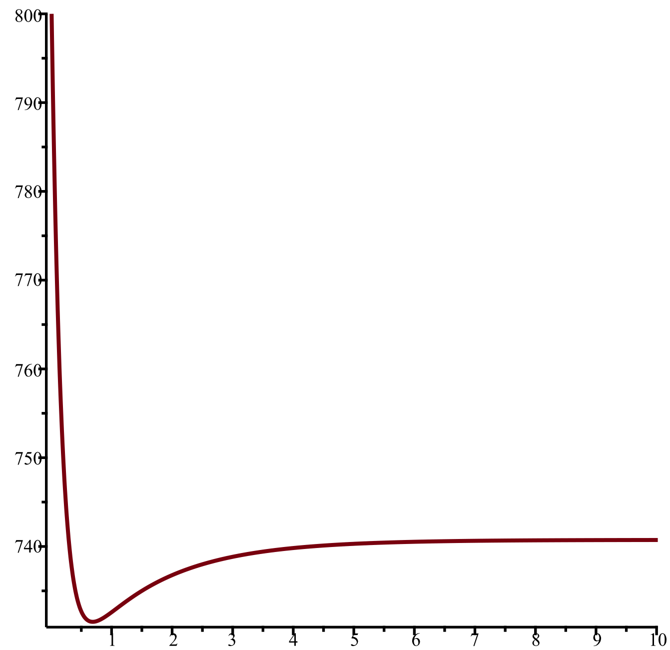
> $TimeSeries(F\_ii1, [s, i], [800, 200], 0.01, 10, 2)$



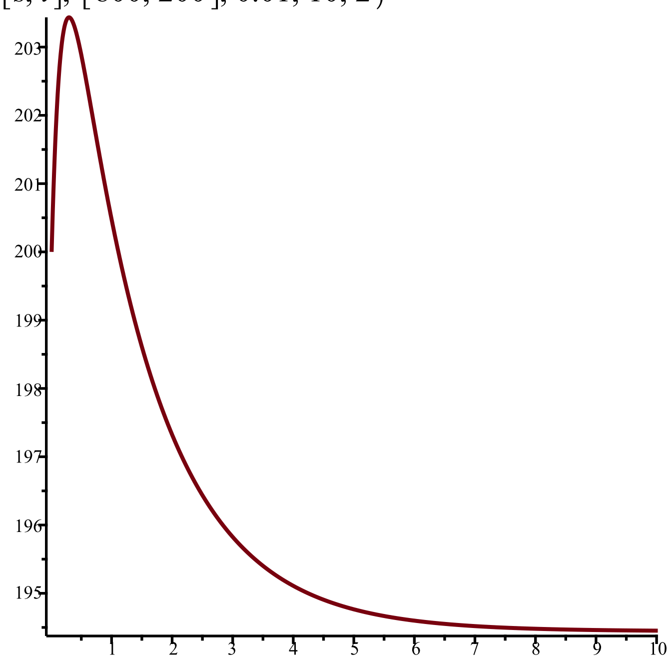
> *PhaseDiag*(*F_ii1*, [*s*, *i*], [800, 200], 0.01, 10)



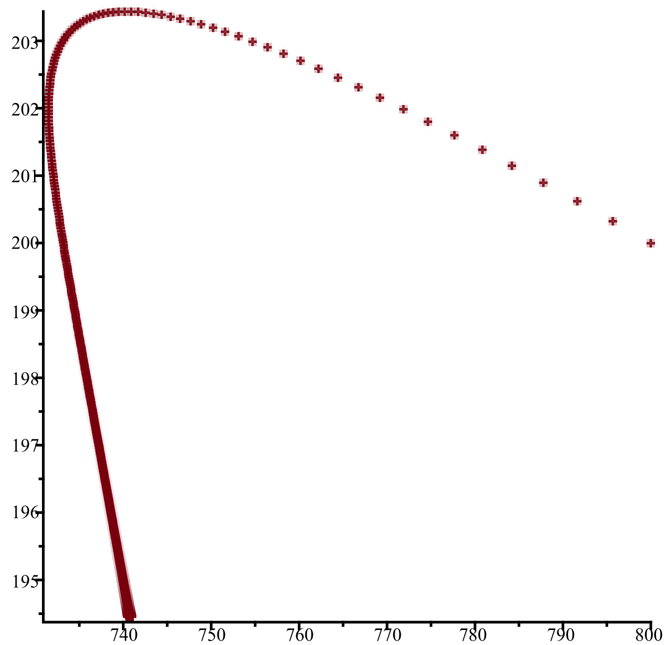
> *TimeSeries*(*F_ii2*, [*s*, *i*], [800, 200], 0.01, 10, 1)



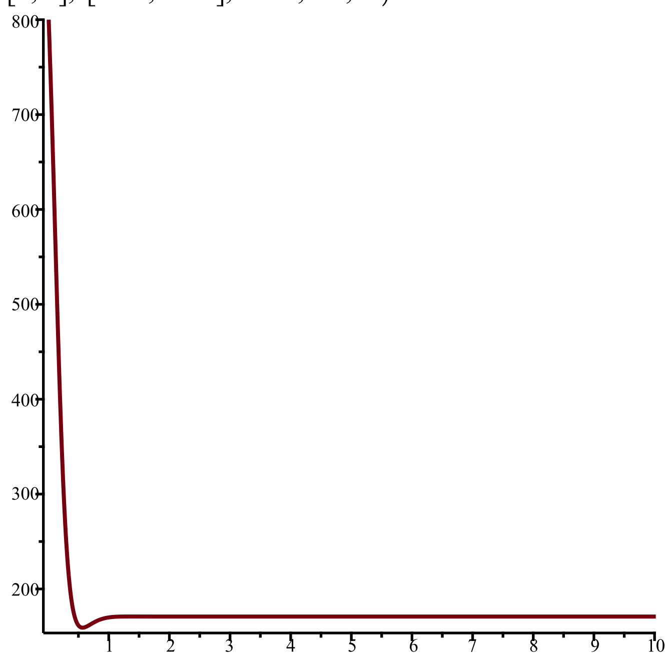
> *TimeSeries*(*F_ii2*, [*s*, *i*], [800, 200], 0.01, 10, 2)



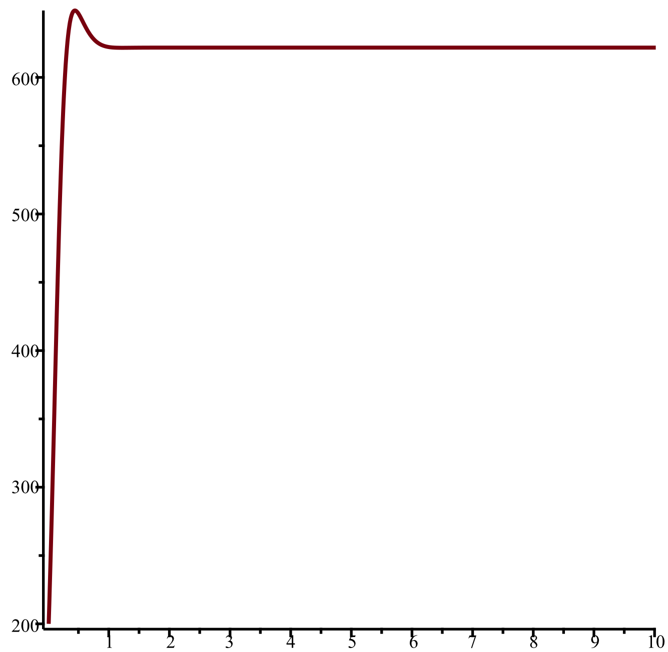
> *PhaseDiag*(*F_ii2*, [*s*, *i*], [800, 200], 0.01, 10)



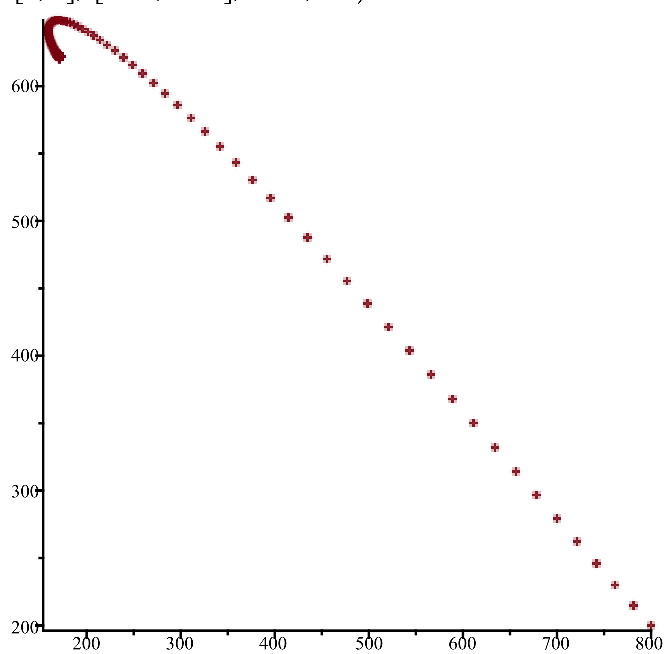
> $TimeSeries(F\_ii3, [s, i], [800, 200], 0.01, 10, 1)$



> $TimeSeries(F\_ii3, [s, i], [800, 200], 0.01, 10, 2)$



> $PhaseDiag(F\_ii3, [s, i], [800, 200], 0.01, 10)$



> #Part (iii)

> $F\_iii1 := SIRS\left(s, i, \frac{0.3 \cdot 4}{1000}, 1, 2, 1000\right)$

$$F\_iii1 := [-0.001200000000\, s\, i + 1000 - s - i, 0.001200000000\, s\, i - 2\, i] \quad (27)$$

> $F\_iii2 := SIRS\left(s, i, \frac{0.9 \cdot 4}{1000}, 1, 2, 1000\right)$

$$F\_iii2 := [-0.003600000000\, s\, i + 1000 - s - i, 0.003600000000\, s\, i - 2\, i] \quad (28)$$

> $F\_iii3 := SIRS\left(s, i, \frac{3.9 \cdot 4}{1000}, 1, 2, 1000\right)$

$$F\_iii3 := [-0.01560000000\, s\, i + 1000 - s - i, 0.01560000000\, s\, i - 2\, i] \quad (29)$$

> $EquP(F\_iii1, [s, i])$

$$\{[1000., 0.], [1666.666667, -222.2222222]\} \quad (30)$$

> $SEquP(F\_iii1, [s, i])$

$$\{[1000., 0.]\} \quad (31)$$

> $EquP(F\_iii2, [s, i])$

$$\{[555.5555556, 148.1481481], [1000., 0.]\} \quad (32)$$

> $SEquP(F\_iii2, [s, i])$

$$\{[555.5555556, 148.1481481]\} \quad (33)$$

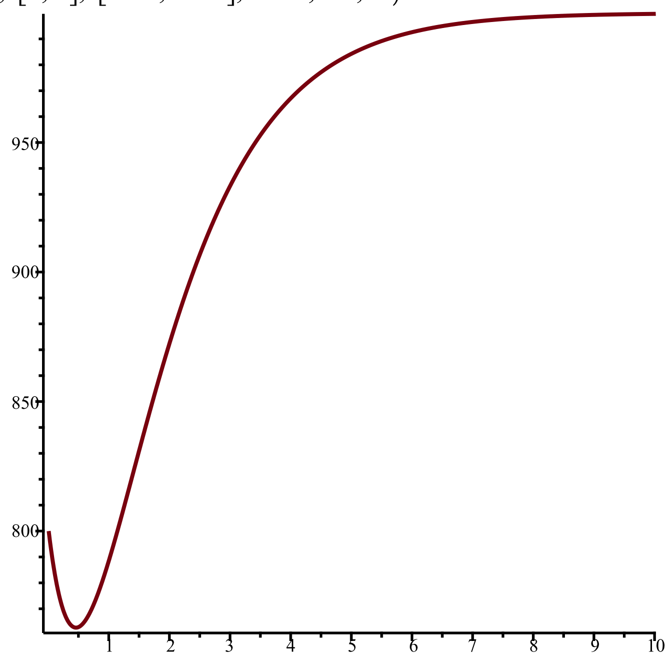
> $EquP(F\_iii3, [s, i])$

$$\{[128.2051282, 290.5982906], [1000., 0.]\} \quad (34)$$

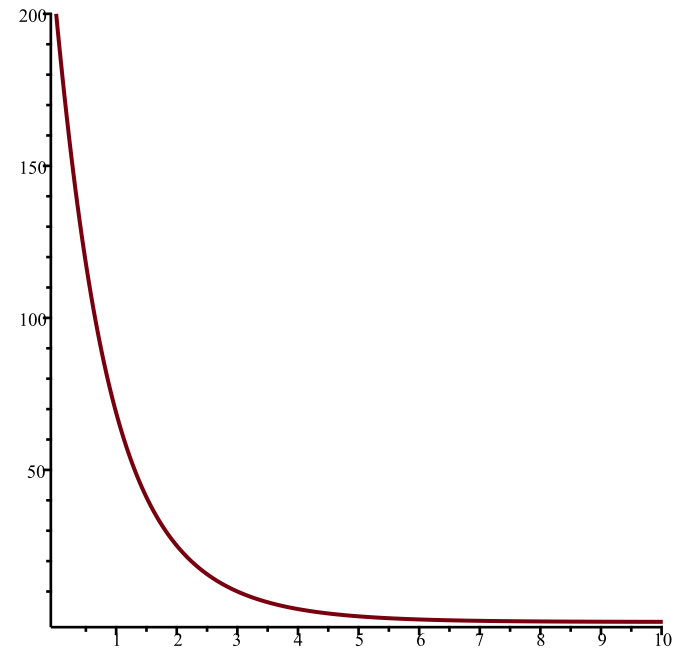
> $SEquP(F\_iii3, [s, i])$

$$\{[128.2051282, 290.5982906]\} \quad (35)$$

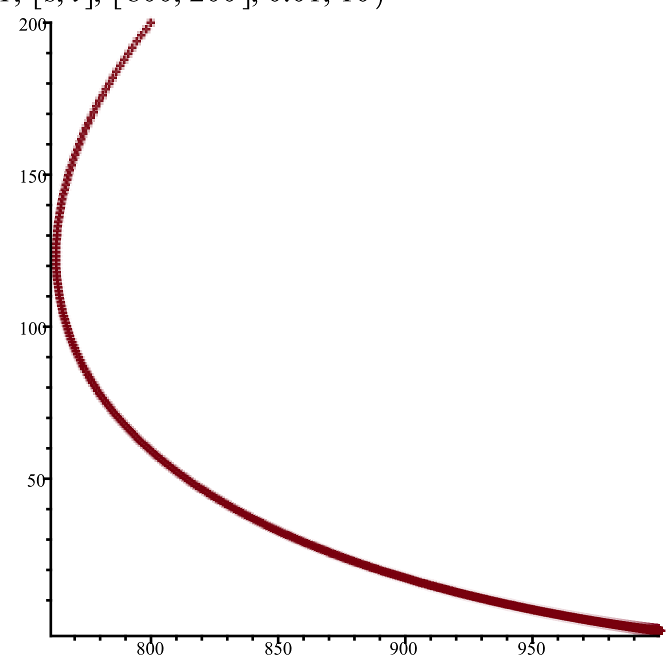
> $TimeSeries(F\_iii1, [s, i], [800, 200], 0.01, 10, 1)$



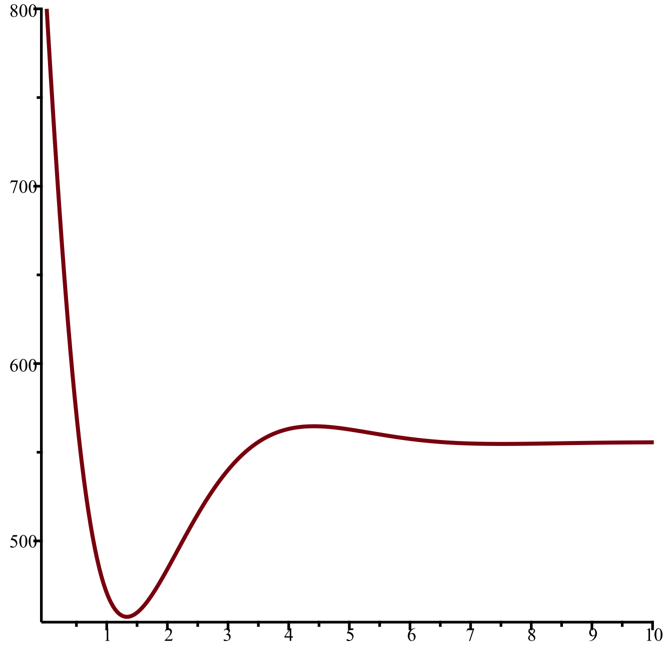
> $TimeSeries(F\_iii1, [s, i], [800, 200], 0.01, 10, 2)$



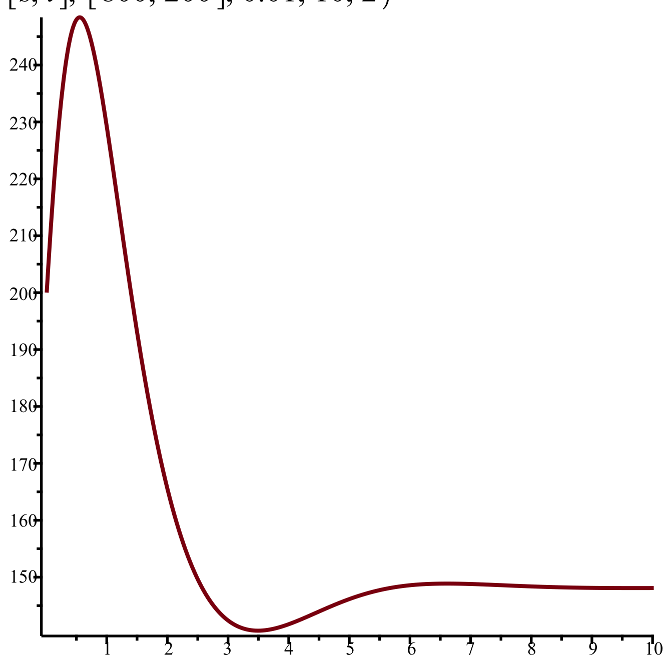
> *PhaseDiag*(*F_iii1*, [*s*, *i*], [800, 200], 0.01, 10)



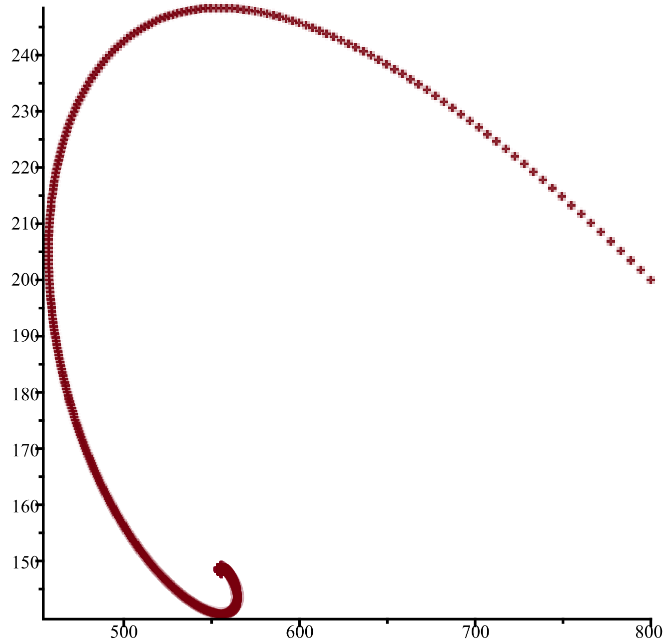
> *TimeSeries*(*F_iii2*, [*s*, *i*], [800, 200], 0.01, 10, 1)



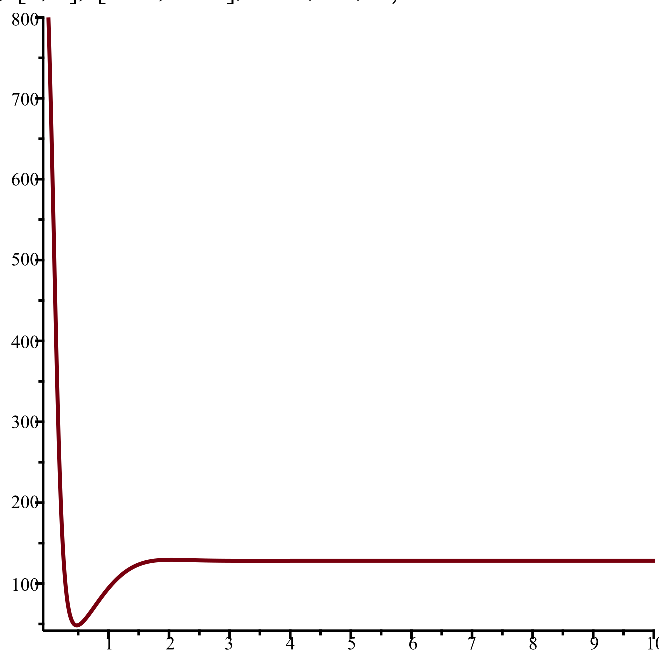
> *TimeSeries*(*F_iii2*, [*s*, *i*], [800, 200], 0.01, 10, 2)



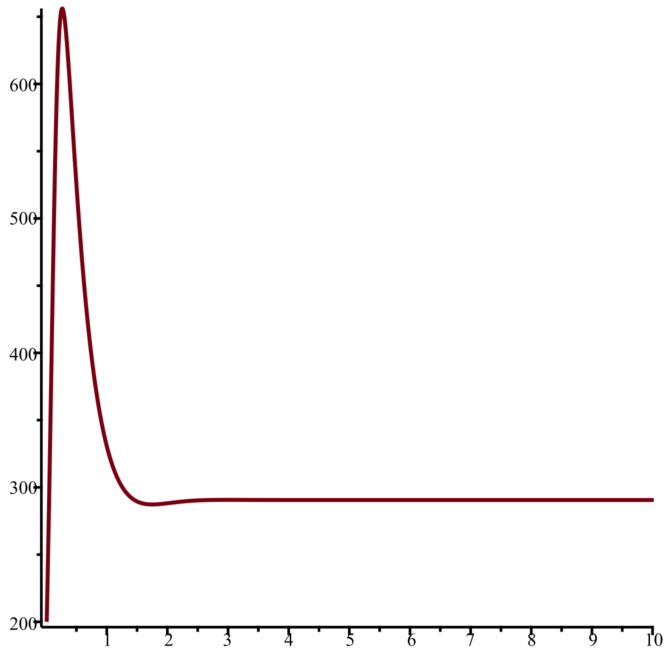
> *PhaseDiag*(*F_iii2*, [*s*, *i*], [800, 200], 0.01, 10)



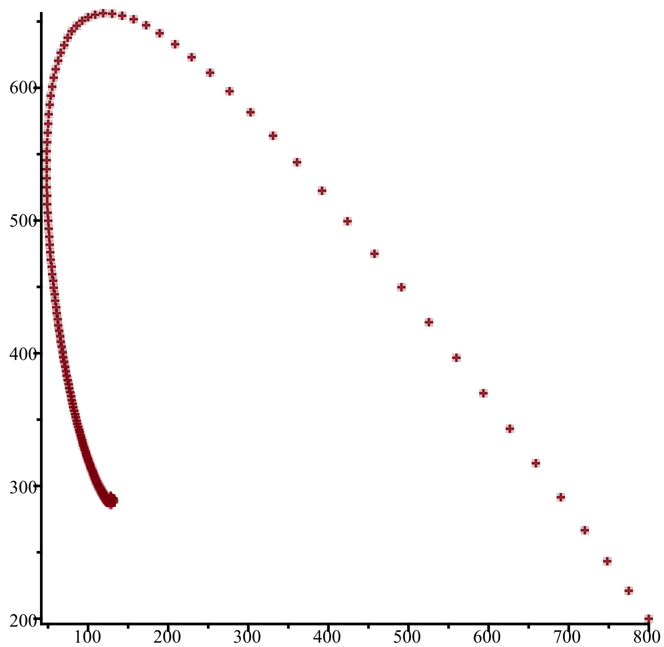
> *TimeSeries*(*F_iii3*, [*s*, *i*], [800, 200], 0.01, 10, 1)



> *TimeSeries*(*F_iii3*, [*s*, *i*], [800, 200], 0.01, 10, 2)



> $PhaseDiag(F\_iii3, [s, i], [800, 200], 0.01, 10)$



>

> *#Part (iv)*

> $F\_iv1 := SIRS\left(s, i, \frac{0.3 \cdot 7}{1000}, 10, 2, 1000\right)$

$$F\_iv1 := [-0.002100000000\, s\, i + 10000 - 10\, s - 10\, i, 0.002100000000\, s\, i - 2\, i] \tag{36}$$

> $F\_iv2 := SIRS\left(s, i, \frac{0.9 \cdot 7}{1000}, 10, 2, 1000\right)$

$$F\_iv2 := [-0.006300000000\, s\, i + 10000 - 10\, s - 10\, i, 0.006300000000\, s\, i - 2\, i] \tag{37}$$

> $F\_iv3 := SIRS\left(s, i, \frac{3.9 \cdot 7}{1000}, 10, 2, 1000\right)$

$$F\_iv3 := [-0.02730000000\, s\, i + 10000 - 10\, s - 10\, i,\ 0.02730000000\, s\, i - 2\, i] \quad \mathbf{(38)}$$

> $EquP(F\_iv1, [s, i])$

$$\{[952.3809524, 39.68253968], [1000., 0.]\} \quad \mathbf{(39)}$$

> $SEquP(F\_iv1, [s, i])$

$$\{[952.3809524, 39.68253968]\} \quad \mathbf{(40)}$$

> $EquP(F\_iv2, [s, i])$

$$\{[317.4603175, 568.7830688], [1000., 0.]\} \quad \mathbf{(41)}$$

> $SEquP(F\_iv2, [s, i])$

$$\{[317.4603175, 568.7830688]\} \quad \mathbf{(42)}$$

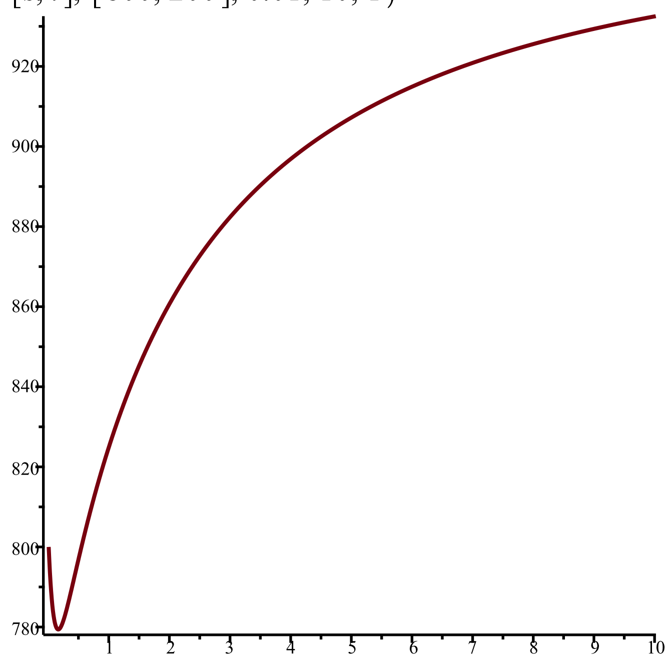
> $EquP(F\_iv3, [s, i])$

$$\{[73.26007326, 772.2832723], [1000., 0.]\} \quad \mathbf{(43)}$$

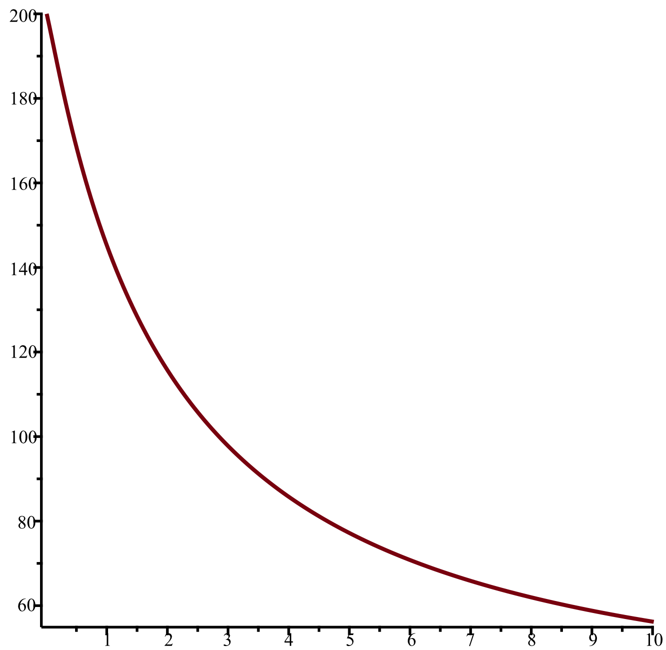
> $SEquP(F\_iv3, [s, i])$

$$\{[73.26007326, 772.2832723]\} \quad \mathbf{(44)}$$

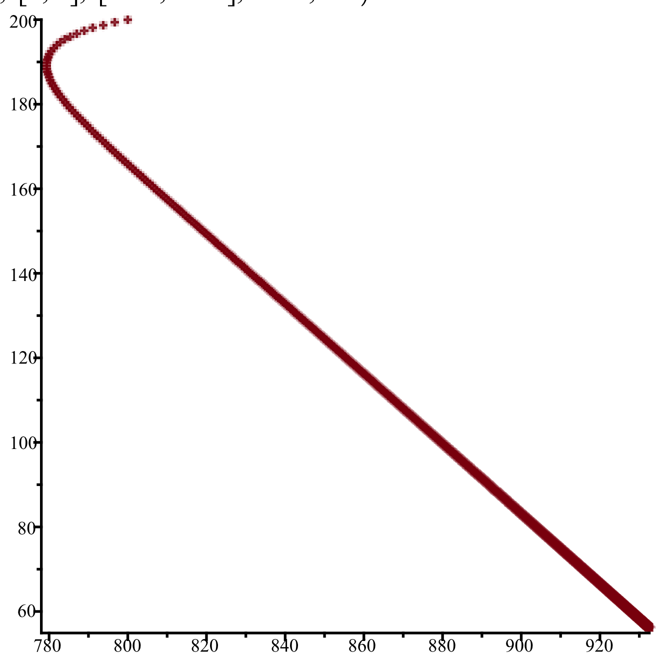
> $TimeSeries(F\_iv1, [s, i], [800, 200], 0.01, 10, 1)$



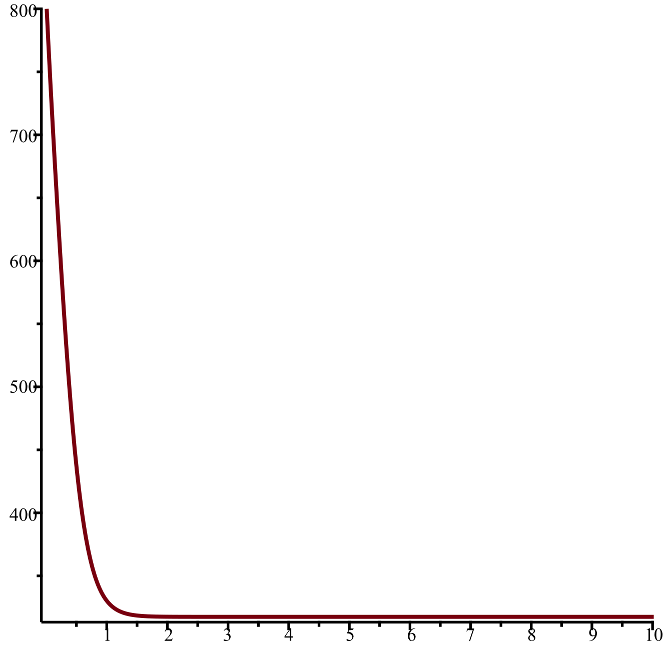
> $TimeSeries(F\_iv1, [s, i], [800, 200], 0.01, 10, 2)$



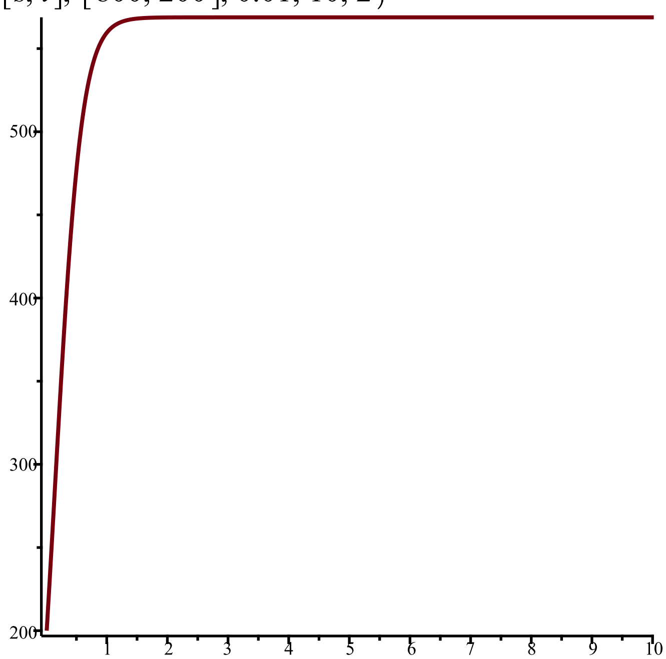
> *PhaseDiag*(*F_iv1*, [*s*, *i*], [800, 200], 0.01, 10)



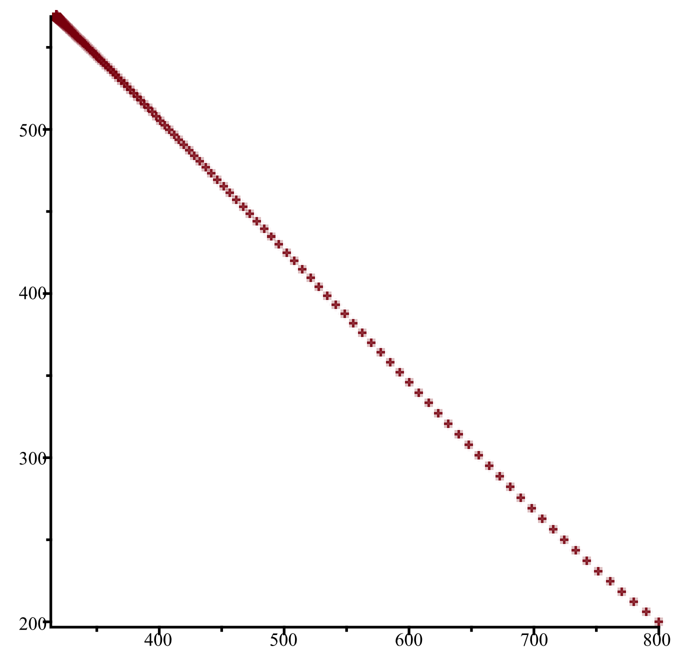
> *TimeSeries*(*F_iv2*, [*s*, *i*], [800, 200], 0.01, 10, 1)



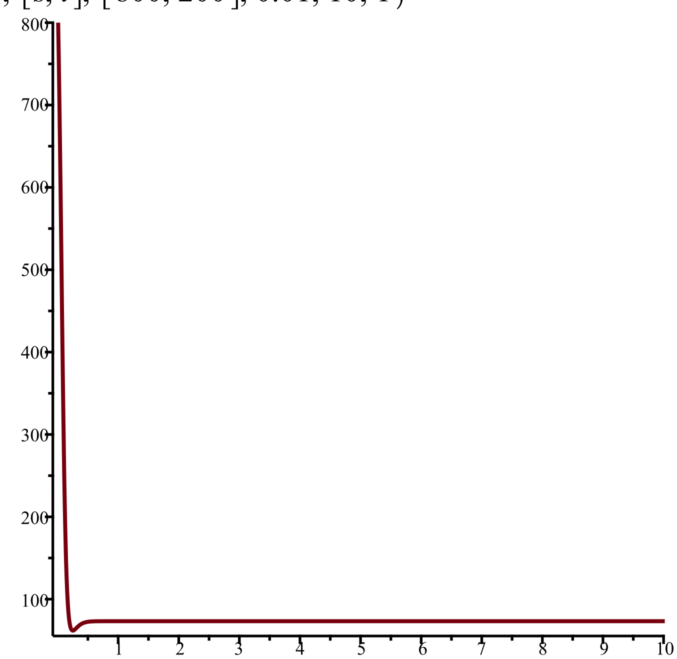
> $TimeSeries(F\_iv2, [s, i], [800, 200], 0.01, 10, 2)$



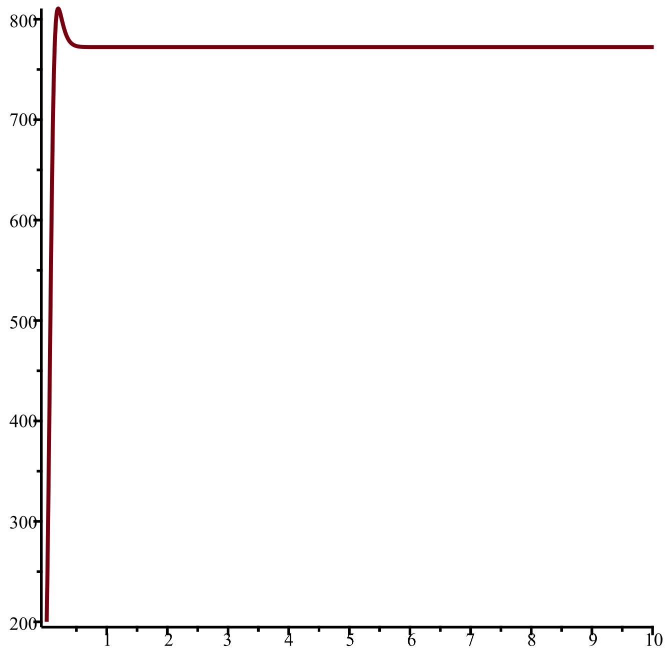
> $PhaseDiag(F\_iv2, [s, i], [800, 200], 0.01, 10)$



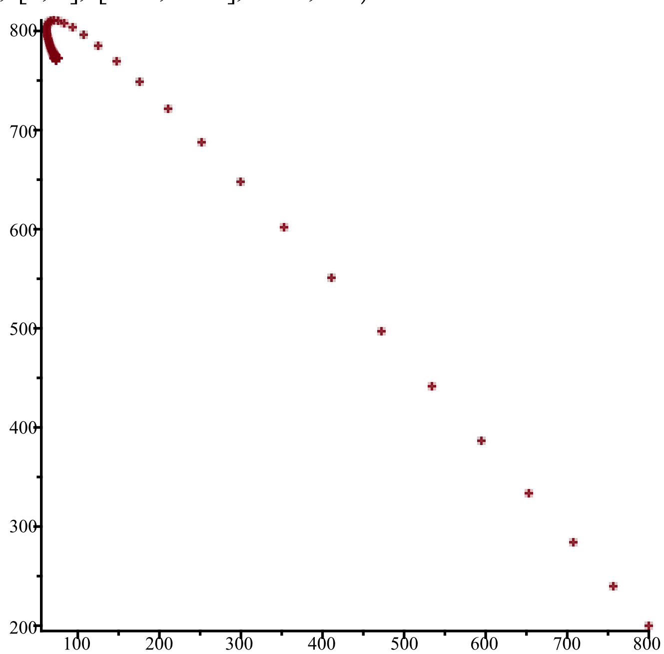
> $TimeSeries(F\_iv3, [s, i], [800, 200], 0.01, 10, 1)$



> $TimeSeries(F\_iv3, [s, i], [800, 200], 0.01, 10, 2)$



> $PhaseDiag(F\_iv3, [s, i], [800, 200], 0.01, 10)$



> *#For all of these, the number of susceptible and number of infected in the long run align with the stable equilibrium points*

>

>

>

> *#Problem 2*

> $F1 := RandNice([x, y], 3)$

$$F1 := [(1 - x - 3y)(1 - 3x - 2y), (3 - x - 2y)(1 - 3x - y)] \quad \textbf{(45)}$$

> $F2 := RandNice([x, y], 3)$

$$F2 := [(2 - 2x - 3y)(3 - 2x - 2y), (1 - x - 3y)(1 - 2x - y)] \quad \textbf{(46)}$$

> $F3 := RandNice([x, y], 3)$

$$F3 := [(2 - 3x - 3y)(3 - 2x - y), (3 - 3x - 3y)(1 - 3x - 3y)] \quad (47)$$

> #For F1

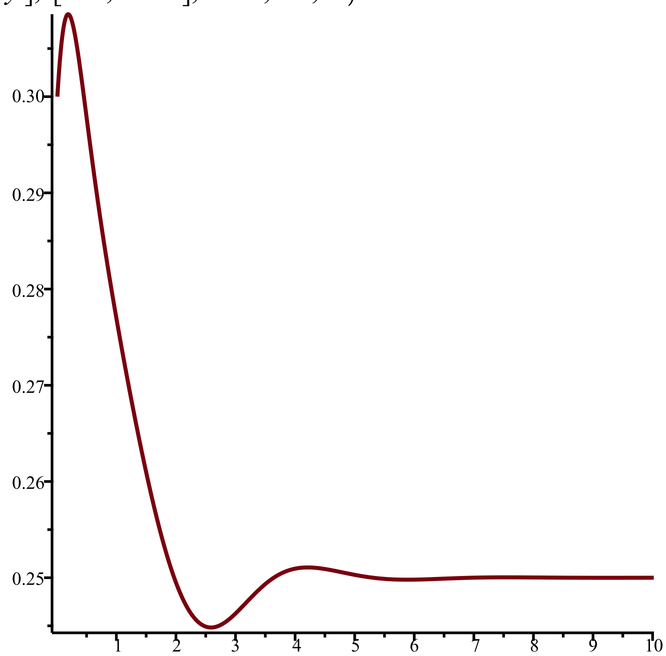
> $EquP(F1, [x, y])$

$$\left\{[-1, 2], [7, -2], \left[\frac{1}{3}, 0\right], \left[\frac{1}{4}, \frac{1}{4}\right]\right\} \quad (48)$$

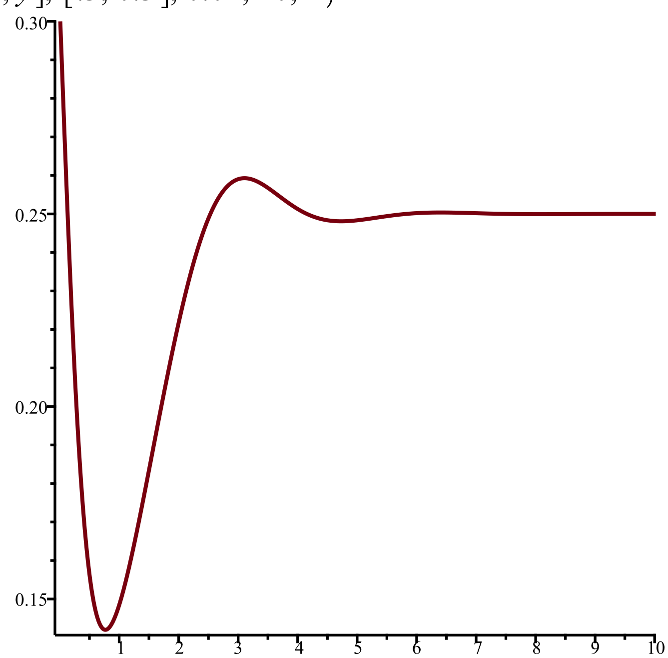
> $SEquP(F1, [x, y])$

$$\{[0.2500000000, 0.2500000000]\} \quad (49)$$

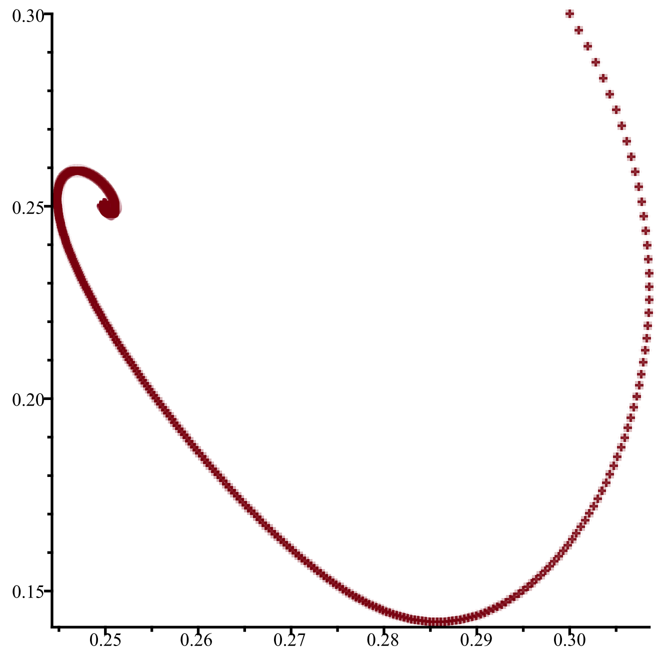
> $TimeSeries(F1, [x, y], [.30, 0.30], 0.01, 10, 1)$



> $TimeSeries(F1, [x, y], [.3, 0.3], 0.01, 10, 2)$



> $PhaseDiag(F1, [x, y], [0.30, 0.30], 0.01, 10)$



>

> *#For F2*

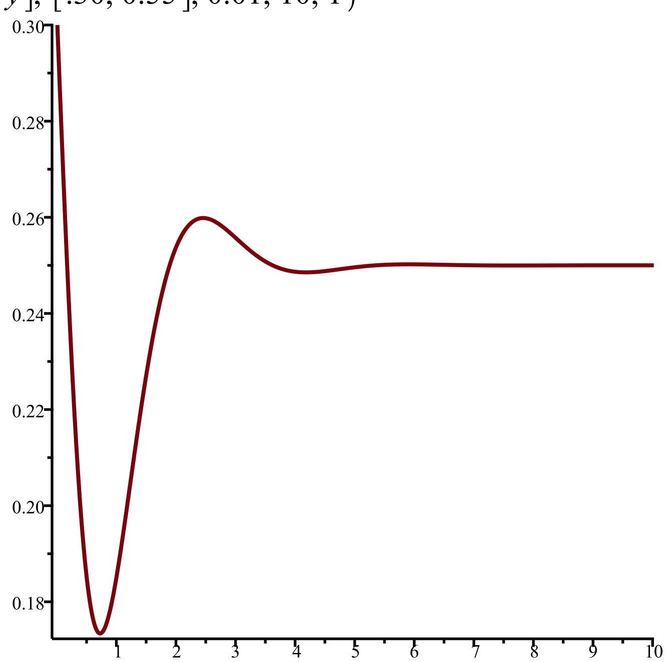
> $EquP(F2, [x, y])$

$$\left\{[1, 0], \left[-\frac{1}{2}, 2\right], \left[\frac{1}{4}, \frac{1}{2}\right], \left[\frac{7}{4}, -\frac{1}{4}\right]\right\} \quad \textbf{(50)}$$

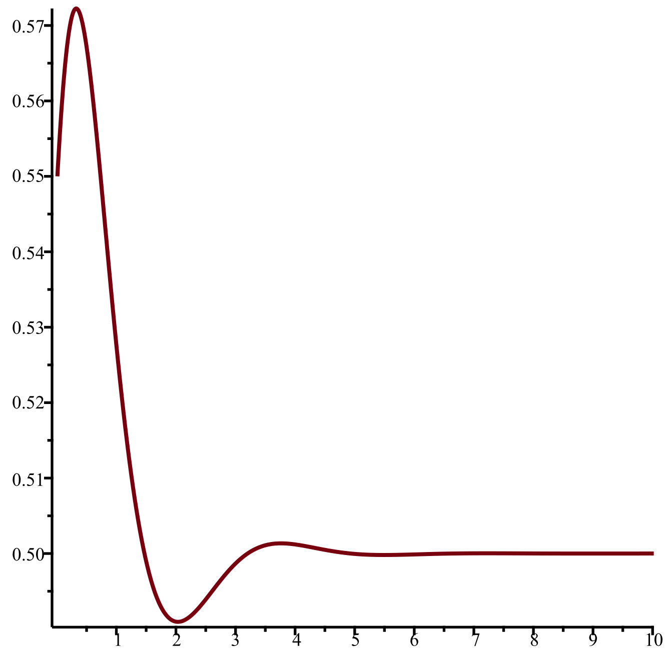
> $SEquP(F2, [x, y])$

$$\{[0.2500000000, 0.5000000000]\} \quad \textbf{(51)}$$

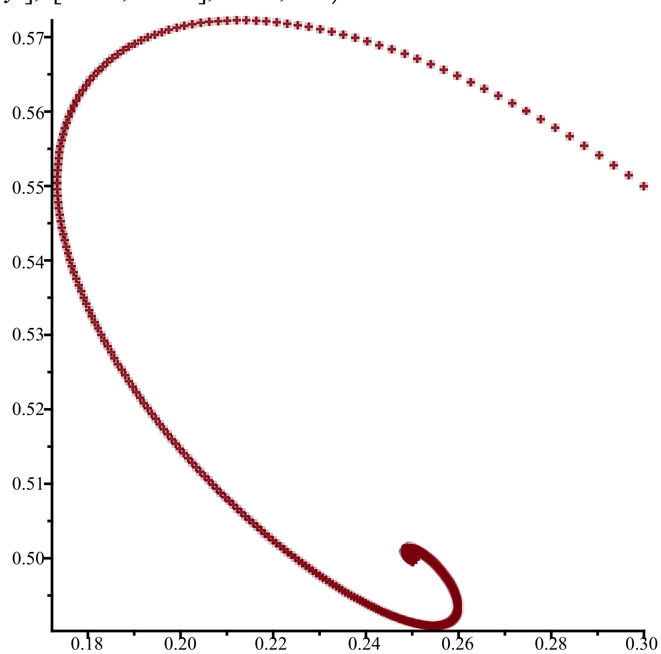
> $TimeSeries(F2, [x, y], [.30, 0.55], 0.01, 10, 1)$



> $TimeSeries(F2, [x, y], [.3, 0.55], 0.01, 10, 2)$



> *PhaseDiag*(*F2*, [*x*, *y*], [0.30, 0.55], 0.01, 10)



> *#For F3*

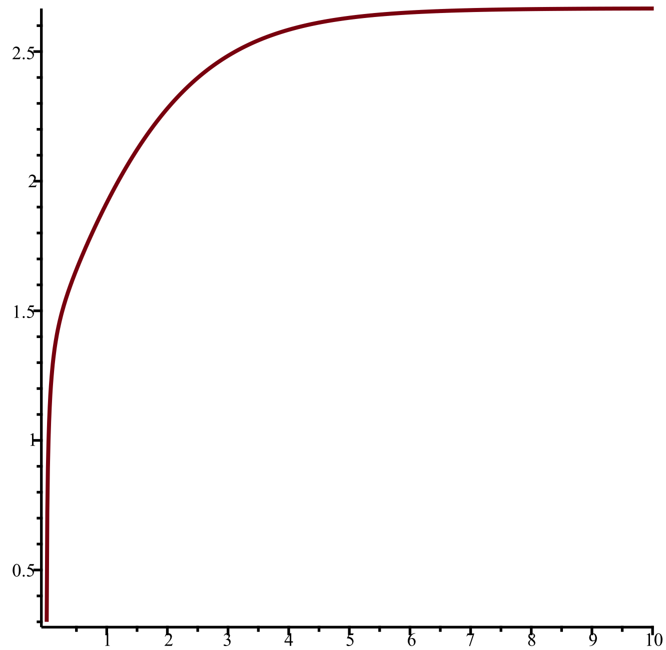
> *EquP*(*F3*, [*x*, *y*])

$$\left\{[2, -1], \left[\frac{8}{3}, -\frac{7}{3}\right]\right\} \quad \textbf{(52)}$$

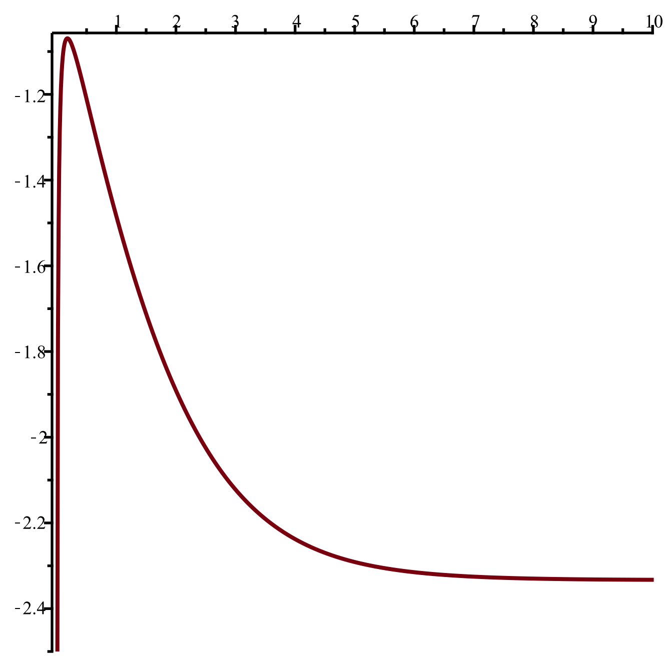
> *SEquP*(*F3*, [*x*, *y*])

$$\{[2.666666667, -2.333333333]\} \quad \textbf{(53)}$$

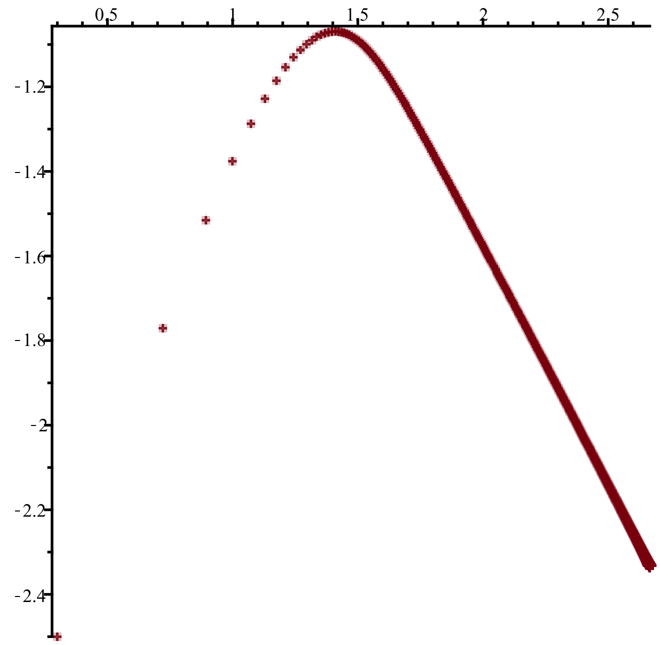
> *TimeSeries*(*F3*, [*x*, *y*], [.30, -2.5], 0.01, 10, 1)



> *TimeSeries*(*F3*, [*x*, *y*], [.3, −2.5], 0.01, 10, 2)



> *PhaseDiag*(*F3*, [*x*, *y*], [0.30, −2.5], 0.01, 10)



> *#The horizontal asymptotes for each align with the stable eq points*

>

>

> *#Problem 3*

>

> *Help*(*Orbk*)

Orbk(k,z,f,INI,K1,K2): Given a positive integer k, a letter (symbol), z, an expression f of z[1], ..., z[k] (representing a multi-variable function of the variables z[1],...,z[k]
a vector INI representing the initial values [x[1],..., x[k]], and (in applications) positive integres K1 and K2, outputs the
values of the sequence starting at n=K1 and ending at n=K2. of the sequence satisfying the difference equation

x[n]=f(x[n-1],x[n-2],..., x[n-k+1]):

This is a generalization to higher-order difference equation of procedure Orb(f,x,x0,K1,K2). For example, try:

*Orbk(1,z,5/2*z[1]*(1-z[1]),[0.5],1000,1010);*

To get the Fibonacci sequence, type:

Orbk(2,z,z[1]+z[2],[1,1],1000,1010);

To get the part of the orbit between n=1000 and n=1010, of the 3rd order recurrence given in Eq. (4) of the Ladas-Amleh paper

https://sites.math.rutgers.edu/~zeilberg/Bio21/AmlehLadas.pdf

with initial conditions x(0)=1, x(1)=3, x(2)=5, Type:

Orbk(3,z,z[2]/(z[2]+z[3]),[1.,3.,5.],1000,1010);

To get the part of the orbit between n=1000 and n=1010, of the 3rd order recurrence given in Eq. (5) of the Ladas-Amleh paper
with initial conditions x(0)=1, x(1)=3, x(2)=5, Type:
Orbk(3,z,(z[1]+z[3])/z[2],[1.,3.,5.],1000,1010);

To get the part of the orbit between n=1000 and n=1010, of the 3rd order recurrence given in Eq. (6) of the Ladas-Amleh paper
with initial conditions x(0)=1, x(1)=3, x(2)=5, Type:
Orbk(3,z,(1+z[3])/z[1],[1.,3.,5.],1000,1010);

To get the part of the orbit between n=1000 and n=1010, of the 3rd order recurrence given in Eq. (7) of the Ladas-Amleh paper
with initial conditions x(0)=1, x(1)=3, x(2)=5, Type:
Orbk(3,z,(1+z[1])/(z[2]+z[3]),[1.,3.,5.],1000,1010); **(54)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [1.5, 1, 1, 1], 1000, 1010\right)$

[1.506309924, 2.239026873, 1.506309924, 2.239026873, 1.506309924, 2.239026873, 1.506309924, 2.239026873, 1.506309924, 2.239026873, 1.506309924] **(55)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [1.5, 1.5, 1, 1], 1000, 1010\right)$

[1.623484305, 2.057654159, 1.623484305, 2.057654159, 1.623484305, 2.057654159, 1.623484305, 2.057654159, 1.623484305, 2.057654159, 1.623484305] **(56)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [1.5, 1.5, 1.5, 1.5], 1000, 1010\right)$

[1.734194427, 1.917928943, 1.734194427, 1.917928943, 1.734194427, 1.917928943, 1.734194427, 1.917928943, 1.734194427, 1.917928943, 1.734194427] **(57)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [2., 2., 2., 2.], 1000, 1010\right)$

[1.866943730, 1.780228265, 1.866943730, 1.780228265, 1.866943730, 1.780228265, 1.866943730, 1.780228265, 1.866943730, 1.780228265, 1.866943730] **(58)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [2.5, 2.5, 2.5, 2.5], 1000, 1010\right)$

[1.978191241, 1.683879292, 1.978191241, 1.683879292, 1.978191241, 1.683879292, 1.978191241, 1.683879292, 1.978191241, 1.683879292, 1.978191241] **(59)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [1.75, 1.75, 1.75, 1.75], 1000, 1010\right)$

[1.803873399, 1.842154845, 1.803873399, 1.842154845, 1.803873399, 1.842154845, 1.803873399, 1.842154845, 1.803873399, 1.842154845, 1.803873399] **(60)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [1.7, 1.7, 1.7, 1.7], 1000, 1010\right)$

[1.790515025, 1.856047753, 1.790515025, 1.856047753, 1.790515025, 1.856047753, 1.790515025, 1.856047753, 1.790515025, 1.856047753, 1.790515025] **(61)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [1.8, 1.8, 1.8, 1.8], 1000, 1010\right)$

[1.816968812, 1.828808992, 1.816968812, 1.828808992, 1.816968812, 1.828808992, 1.816968812, 1.828808992, 1.816968812, 1.828808992, 1.816968812] **(62)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [1.81, 1.81, 1.81, 1.81], 1000, 1010\right)$

[1.819557373, 1.826202283, 1.819557373, 1.826202283, 1.819557373, 1.826202283, 1.819557373, 1.826202283, 1.819557373, 1.826202283, 1.819557373] **(63)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [1.82, 1.82, 1.82, 1.82], 1000, 1010\right)$

[1.822135971, 1.823615754, 1.822135971, 1.823615754, 1.822135971, 1.823615754, 1.822135971, 1.823615754, 1.822135971, 1.823615754, 1.822135971] **(64)**

> *#Stable equilibrium around 1.82*

>

> *Help*(*ToSys*)

ToSys(k,z,f): converts the kth order difference equation x(n)=f(x[n-1],x[n-2],...x[n-k]) to a first-order system

x1(n)=F(x1(n-1),x2(n-1), ...,xk(n-1)), it gives the unerlying transormation, followed by the set of variables

Try:

ToSys(2,z,z[1]+z[2]); **(65)**

> $p3 := ToSys\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}\right)$

$$p3 := \left[\frac{3+z_2+z_3+z_4}{1+z_1+z_3}, z_1, z_2, z_3\right], [z_1, z_2, z_3, z_4]$$ **(66)**

> *Help*(*SFP*)

SFP(F,x): Given a transformation F in the list of variables finds all the STABLE fixed point of the transformation x->F(x), i.e. the set of solutions of

the system {x[1]=F[1], ..., x[k]=F[k]}} that are stable. Try:

*SFP([5/2*x*(1-x)],[x]]);*

*SFP([(1+x+y)/(2+3*x+y), (3+x+2*y)/(5+x+3*y)],[x,y]]);* **(67)**

> *SFP*(*p3*)

{[1.822875656, 1.822875656, 1.822875656, 1.822875656]} **(68)**

> $Orbk\left(4, z, \frac{(3+z[2]+z[3]+z[4])}{1+z[1]+z[3]}, [1.822875656, 1.822875656, 1.822875656,\right.$

1.822875656], 1000, 1010)

[1.822875656, 1.822875655, 1.822875656, 1.822875655, 1.822875656, 1.822875655, 1.822875656, 1.822875655, 1.822875656, 1.822875655, 1.822875656] **(69)**

>

>

>

>

>

>

>

>

>