Mathematica Workbooks

Conduction Heat Transfer

ME 8341

Bischof - Fall 1999

Contents:

☆ check $C_p = \rho C_p$

or

$C_p = C_p$

```
In[158]:= (* This nb looks at the solution of the 1 D insulated rod with dirichlet
            conditions at the boudaries with an initial condition of 1 and an L of 0.1*)

          L = 0.1
          a = 10^-4
          PI = 22/7
          c = Exp[-a*1*(1*PI/L)^2]
          d = Sin[1*0.1*PI/L]
          t = 10

          Plot3D[(4/PI)*Sum[(1/(2 i - 1))*Exp[-a*t*((2 i - 1)*PI/L)^2]*Sin[(2 i - 1)*x*PI/L],
             {i, 1, 3}], {x, 0, 0.1}, {t, 0, 10} ]
```

Out[158]= 0.1

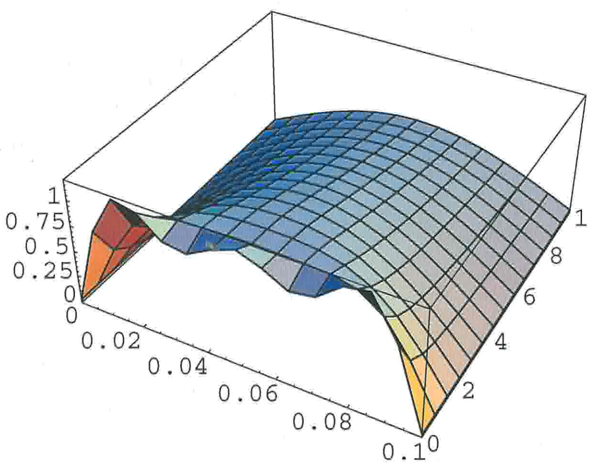
Out[159]= $\frac{1}{10000}$

Out[160]= $\frac{22}{7}$

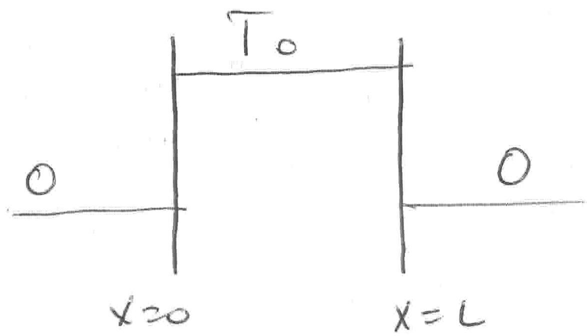
Out[161]= 0.905946

Out[162]= -0.00126449

Out[163]= 10



Out[164]= - SurfaceGraphics -



$$T_{xx} = \frac{1}{\alpha} T_t$$

$x = 0, L \quad T = 0 \quad t > 0$

$t = 0 \quad T = T_o \quad 0 < x < L$

soz N

$$T(x,t) = \underset{(1)}{\frac{4}{\pi}} T_o \sum_{m=1}^{\infty} \frac{1}{(2m-1)} \sin\frac{(2m-1)\pi x}{L} \exp\left(-\alpha\left(\frac{(2m-1)\pi}{L}\right)^2 t\right)$$

```
(* This nb looks at the solution of semi inf
   domain with t^0.5 as the boundary condition at x eq. 0.  This
   closely follows the Duhamel method of Ch. 5 - p. 205 text*)
```

DUHAMEL

$$T(x,t) = \frac{x}{2\sqrt{\pi\alpha}} \int_{\tau=0}^{t} \frac{f(\tau)}{(t-\tau)^{3/2}} \exp\left[\frac{-x^2}{4\alpha(t-\tau)}\right] d\tau$$

where $f(\tau) = \tau^{1/2}$

```
a = 10^-4
PI = 22/7
x = 0.4
t = 500

g = (x/(2*(PI*a)^(0.5)))*
  NIntegrate
   [((TAU^0.5)/(t-TAU)^(1.5))*
    Exp[-x^2/(4*a*(t-TAU))],
   {TAU, 0, t}]

Plot3D[(x/(2*(PI*a)^(0.5)))*
  NIntegrate
   [((TAU^0.5)/(t-TAU)^(1.5))*
    Exp[-x^2/(4*a*(t-TAU))],
   {TAU, 0, t}]
 , {x, 0.4, 1}, {t, 0.1, 1000}]
```

Out[79]= $\frac{1}{10000}$

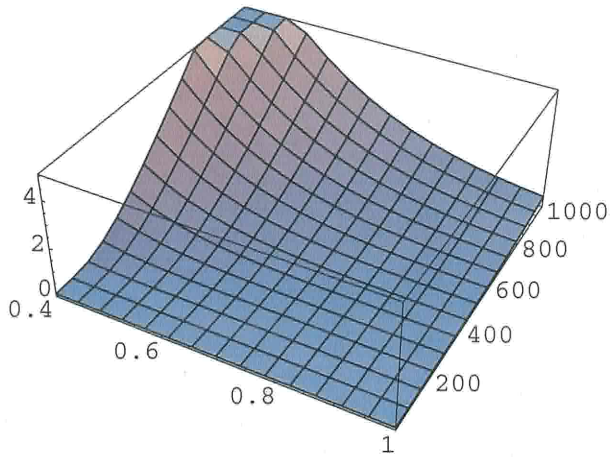
Out[80]= $\frac{22}{7}$

Out[81]= 0.4

Out[82]= 500

Out[83]= 2.74767

```
NIntegrate::ploss :
 Numerical integration stopping due to loss of precision. Achieved neither the requested PrecisionGoal nor
   AccuracyGoal; suspect one of the following: highly oscillatory integrand or the true value of the
   integral is 0. If your integrand is oscillatory try using the option Method->Oscillatory in NIntegrate.

NIntegrate::ploss :
 Numerical integration stopping due to loss of precision. Achieved neither the requested PrecisionGoal nor
   AccuracyGoal; suspect one of the following: highly oscillatory integrand or the true value of the
   integral is 0. If your integrand is oscillatory try using the option Method->Oscillatory in NIntegrate.

NIntegrate::ploss :
 Numerical integration stopping due to loss of precision. Achieved neither the requested PrecisionGoal nor
   AccuracyGoal; suspect one of the following: highly oscillatory integrand or the true value of the
   integral is 0. If your integrand is oscillatory try using the option Method->Oscillatory in NIntegrate.

General::stop : Further output of NIntegrate::ploss will be suppressed during this calculation.
```



Out[84]= - SurfaceGraphics -

In[85]:=
```
(* This nb looks at the solution of semi
   inf domain with t^0.5 as the boundary condition at x=
  0.  This closely follows the Laplace Transform example of Ch. 7 - p. 275 text*)
```

LAPLACE

$$T(x,t) = T_0 \left[t^{1/2} \exp(-x^2/4\alpha t) - \frac{x}{2}\sqrt{\frac{\pi}{\alpha}}\, \mathrm{erfc}\left(\frac{x}{2\sqrt{\alpha t}}\right)\right]$$

(1)

```
a = 10^-4
PI = 22/7
x = 0.5
t = 500

g = t^0.5*
   Exp[-x^2/(4*a*t)] -
  (x/2)*(PI/a)^0.5*
   Erfc[x/(2*(a*t)^0.5)]

Plot3D[t^0.5*
   Exp[-x^2/(4*a*t)] -
  (x/2)*(PI/a)^0.5*
   Erfc[x/(2*(a*t)^0.5)]
 , {x, 0.01
  , 1}, {t, 0.1, 1000}]
```

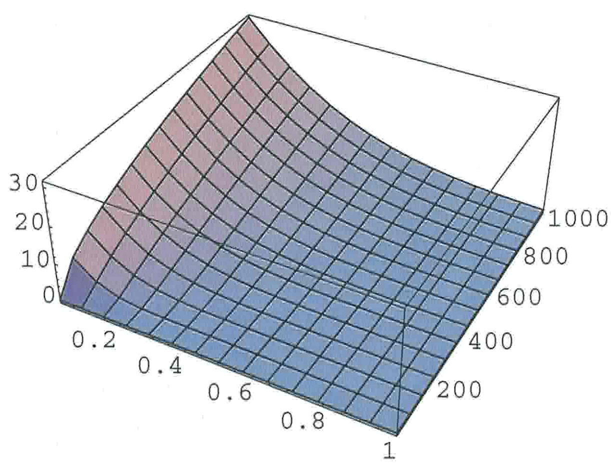
Out[85]= $\frac{1}{10000}$

Out[86]= $\frac{22}{7}$

Out[87]= 0.5

Out[88]= 500

Out[89]= 1.36074



Out[90]= - SurfaceGraphics -

```
Plot3D[t^0.5*
   Exp[-x^2/(4*a*t)] -
  (x/2)*(PI/a)^0.5*
   Erfc[x/(2*(a*t)^0.5)]
 , {x, 0.2
 , 1}, {t, 0.1, 1000}]
```

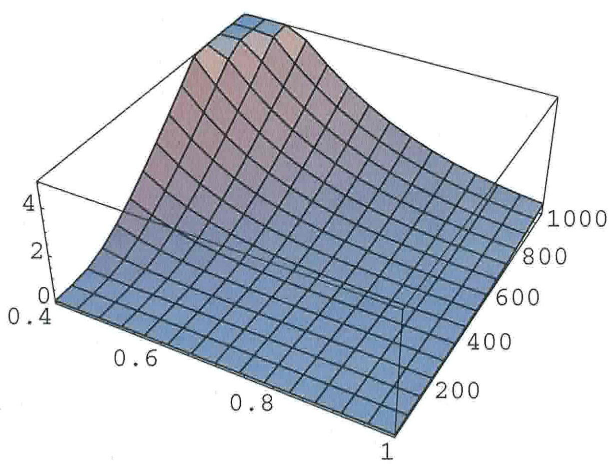
Out[67]= $\frac{1}{10000}$

Out[68]= $\frac{22}{7}$

Out[69]= 1000

Out[70]= 1

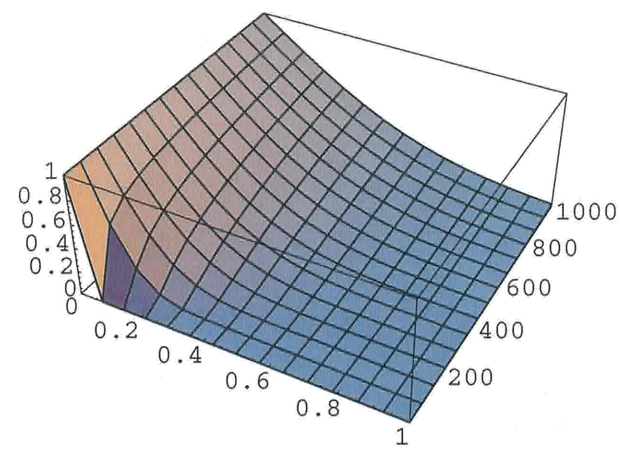
Out[71]= 0.348956



Out[72]= - SurfaceGraphics -

```
In[67]:= a = 10^-4

         Plot3D[Erfc[x / ((4 a * t) ^ (0.5))], {x, 0, 1}, {t, 0.1, 1000}, PlotRange → {0, 1} ]
```

Out[67]= $\frac{1}{10000}$



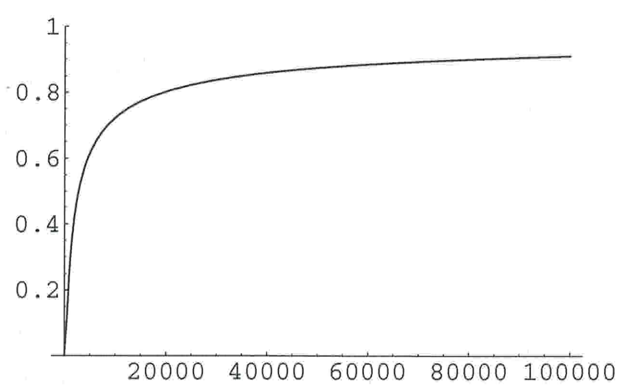
$$\frac{T(x,t)}{T_0} = Erfc\left(\frac{x}{2\sqrt{\alpha t}}\right)$$

or

$$\frac{T(x,t) - T_\infty}{T_0 - \underset{T_i}{\underset{\parallel}{T_\infty}}}$$

Out[68]= - SurfaceGraphics -

```
In[71]:= Plot[Erfc[0.5 / ((4 a * t) ^ (0.5))], {t, 0, 100000}, PlotRange → {0, 1} ]
```



Out[71]= - Graphics -

```
(* This finds the lambda (x) root of
  the transcendental equation for the Neumann problem. *)

kl = 0.5
ks = 2.0
as = 2.0 / (921 * 2100)
al = 0.5 / (999 * 4200)
Cps = 2100 * 921
Tm = 0
To = -100
Ti = 20
L = 335000
PI = 22 / 7

FindRoot[Exp[-x^2] / Erf[x] +
   (kl / ks) * (as / al) ^ (0.5) *
    ((Tm - Ti) / (Tm - To)) * Exp[-x^2 * (as / al)] / Erf[x * (as / al) ^ (0.5)] ==
  (x * L * PI ^ (0.5)) / (Cps * (Tm - To)), {x, 1}]
```

Out[155]= 0.5

Out[156]= 2.

Out[157]= 1.03407×10^{-6}

Out[158]= 1.19167×10^{-7}

Out[159]= 1934100

Out[160]= 0

Out[161]= -100

Out[162]= 20

Out[163]= 335000

Out[164]= $\frac{22}{7}$

Out[165]= {x → 2.23256}

p 408 TEXT

$$\frac{\exp -\lambda^2}{\operatorname{erf} \lambda} + \frac{k_\ell}{k_s}\left(\frac{\alpha_s}{\alpha_\ell}\right)^{1/2} \frac{T_m - T_i}{T_m - T_o} \frac{\exp(-\lambda^2 \alpha_s/\alpha_\ell)}{\operatorname{erfc}\left[\lambda(\alpha_s/\alpha_\ell)^{1/2}\right]}$$

$$= \frac{\lambda L \sqrt{\pi}}{C_{ps}(T_m - T_o)}$$

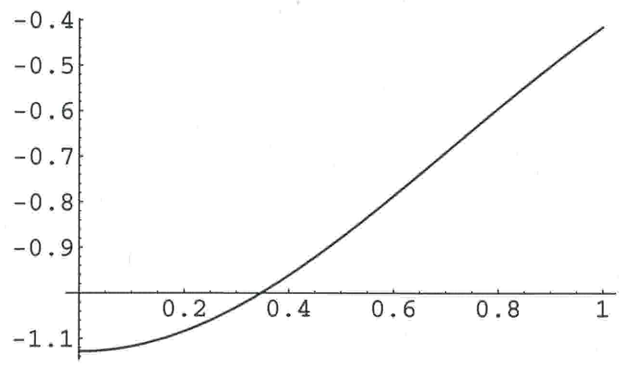
```
In[236]:= x = 1
          g = Derivative[1][Erfc][x]
          h = Derivative[1][Erfc][x]

          Plot[ Derivative[1][Erfc][x], {x, 0, 1} ]
          Plot[ Derivative[1][Erf][x], {x, 0, 1} ]

          Plot3D[ Derivative[1][Erfc][x / ((4 a * t) ^ (0.5))], {t, 0.1, 100}, {x, 0, 1} ]

          Plot3D[ Derivative[1][Erf][x / ((4 a * t) ^ (0.5))], {t, 0.1, 100}, {x, 0, 1} ]
```

$$\left.\frac{\partial\, \mathrm{erfc}\, x}{\partial x}\right|_{x=1}$$

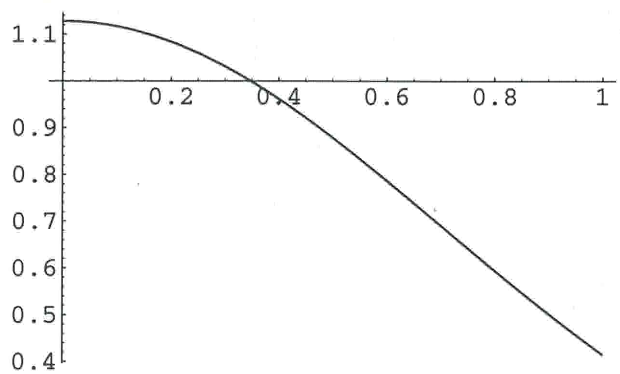
Out[236]= 1

Out[237]= $-\frac{2}{e\sqrt{\pi}}$

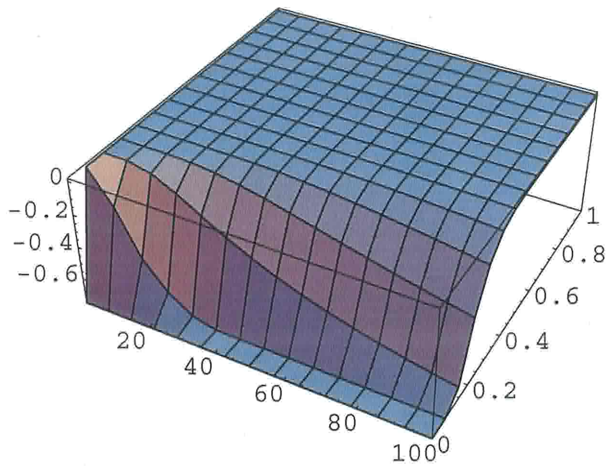
Out[238]= $-\frac{2}{e\sqrt{\pi}}$



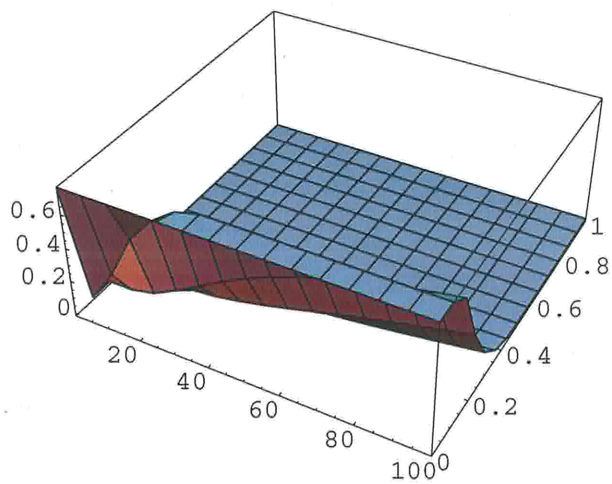
Out[239]= - Graphics -



Out[240]= - Graphics -




```
Out[241]=  - SurfaceGraphics -
```



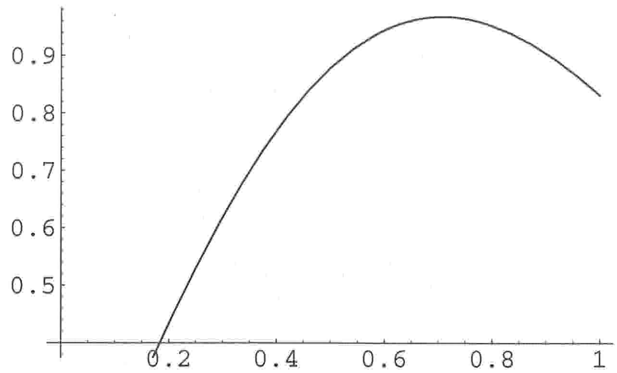

```
Out[242]=  - SurfaceGraphics -

In[243]:=  x = 1
           g = Derivative[2][Erfc][x]
           h = Derivative[2][Erfc][x]

           Plot[ Derivative[2][Erfc][x], {x, 0, 1} ]
           Plot[ Derivative[2][Erf][x], {x, 0, 1} ]

           Plot3D[ Derivative[2][Erfc][x / ((4 a * t) ^ (0.5))], {t, 0.1, 100}, {x, 0, 1} ]

           Plot3D[ Derivative[2][Erf][x / ((4 a * t) ^ (0.5))], {t, 0.1, 100}, {x, 0, 1} ]
```

$$\left.\frac{\partial^2 \operatorname{erf} x}{\partial x^2}\right|_{x=1}$$

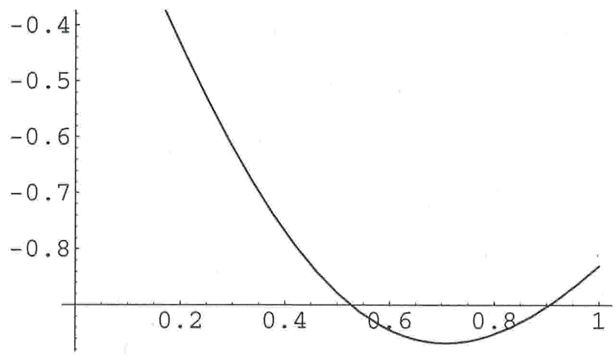
Out[243]= 1

Out[244]= $\frac{4}{e\sqrt{\pi}}$

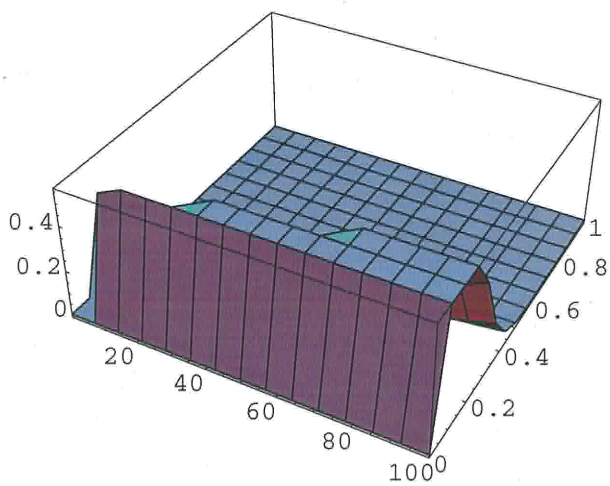
Out[245]= $\frac{4}{e\sqrt{\pi}}$



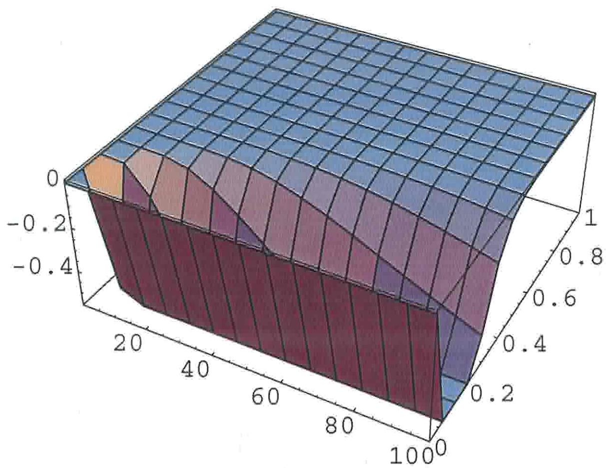
Out[246]= - Graphics -



Out[247]= - Graphics -



Out[248]= - SurfaceGraphics -



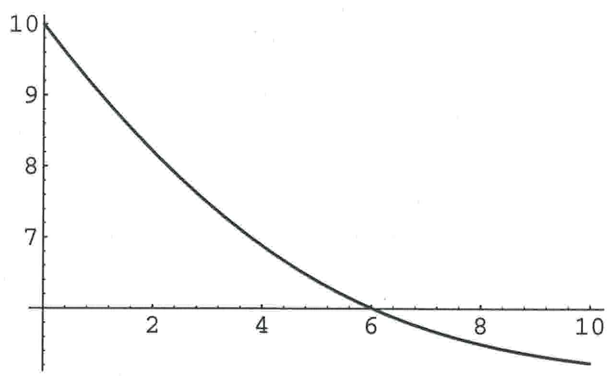
Out[249]= - SurfaceGraphics -

```
m={{-9,19,4},{-3,7,1},{-7,17,2}}

{{-9, 19, 4}, {-3, 7, 1}, {-7, 17, 2}}

Eigenvalues[m]

{0, -I, I}

Inverse[m]

LinearSolve::nosol:
   Linear equation encountered which has no solution.

Inverse[{{-9, 19, 4}, {-3, 7, 1}, {-7, 17, 2}}]

m={{1,-3,0,-2},{3,-12,-2,-6},{-2,10,2,5},{-1,6,1,3}}

{{1, -3, 0, -2}, {3, -12, -2, -6}, {-2, 10, 2, 5},
  {-1, 6, 1, 3}}

Inverse[m]

{{0, 1, 0, 2}, {1, -1, -2, 2}, {0, 1, 3, -3}, {-2, 2, 3, -2}}

m={{1,-1,4},{3,2,-1},{2,1,-1}}

{{1, -1, 4}, {3, 2, -1}, {2, 1, -1}}

Eigenvalues[m]

{-2, 1, 3}

n={{3,2,4},{2,0,2},{4,2,3}}

{{3, 2, 4}, {2, 0, 2}, {4, 2, 3}}

Eigenvalues[n]

{-1, -1, 8}

m•n

m•n

m.n

{{17, 10, 14}, {9, 4, 13}, {4, 2, 7}}

MatrixForm[m]

1    -1    4
3    2     -1
2    1     -1
```

NUM. SOL'N TO
NON-LINEAR DE

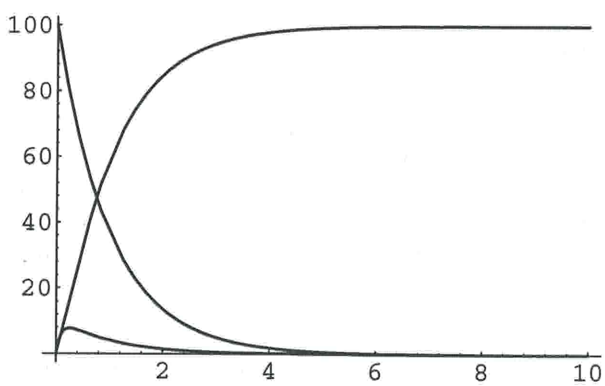

```
sol = NDSolve[
{x'[t]== 1(10/x[t] - 2),

x[0]==10},

{x}, {t,100}]

{{x -> InterpolatingFunction[{0., 100.}, <>]}}
```

```
Plot[Evaluate[{x[t]} /. sol], {t,0,10}]
```




```
-Graphics-
```

```
eqns = Join[
{x'[t]== -x[t], y'[t]== x[t] - y[t],

z'[t]== y[t],

x[0]==100,y[0]==0, z[0]==0},

{x,y,z}, {t,10}]

{x'[t] == -x[t], y'[t] == x[t] - y[t], z'[t] == y[t], x[0] == 100,
  y[0] == 0, z[0] == 0, x, y, z, t, 10}

sol = NDSolve[
{x'[t]== -x[t], y'[t]== x[t] - 10y[t],

z'[t]== 10y[t],

x[0]==100, y[0]==0, z[0]==0},

{x,y,z}, {t,100}]

{{x -> InterpolatingFunction[{0., 100.}, <>],
   y -> InterpolatingFunction[{0., 100.}, <>],
   z -> InterpolatingFunction[{0., 100.}, <>]}}

Plot[Evaluate[{x[t],y[t],z[t]} /. sol], {t,0,10}]
```



-Graphics-

$$X \xrightarrow{k_1} y \xrightarrow{k_2} Z$$

chemical kinetics

$$\frac{dX}{dt} = -k_1 X$$

$$\frac{dy}{dt} = +k_1 x - k_2 y$$

$$\frac{dz}{dt} = k_2 y$$

above kinetic coeffs.

$k_1 = 1 \frac{1}{s}$

$k_2 = 10 \frac{1}{s}$