

Scilab Textbook Companion for
Numerical Methods
by B. Ram¹

Created by
Saurav Suman
B.Tech
Others
NIT Jamshedpur
College Teacher
Self
Cross-Checked by
K. V. P. Pradeep

May 27, 2016

¹Funded by a grant from the National Mission on Education through ICT, <http://spoken-tutorial.org/NMEICT-Intro>. This Textbook Companion and Scilab codes written in it can be downloaded from the "Textbook Companion Project" section at the website <http://scilab.in>

Book Description

Title: Numerical Methods

Author: B. Ram

Publisher: Pearson

Edition: 1

Year: 2010

ISBN: 9788131732212

Scilab numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

AP Appendix to Example(Scilab Code that is an Appednix to a particular Example of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means a scilab code whose theory is explained in Section 2.3 of the book.

Contents

List of Scilab Codes	4
1 Preliminaries	6
2 Non Linear Equations	8
3 Linear System of Equations	40
4 Eigenvalues and Eigenvectors	69
5 Finite Differences and Interpolation	86
6 Curve Fitting	123
7 Numerical Differentiation	136
8 Numerical Quadrature	151
9 Difference Equations	167
10 Ordinary Differential Equations	178
11 Partial Differential Equations	205

List of Scilab Codes

Exa 1.1	Rounding off Numbers	6
Exa 1.2	Relative Maximum Error	6
Exa 1.3	Absolute Error	7
Exa 2.1	Bisection Method	8
Exa 2.2	Bisection Method	9
Exa 2.3	Regula Falsi Method	9
Exa 2.4	Regula Falsi Method	10
Exa 2.5	Regula Falsi Method	11
Exa 2.6	Secant Method	12
Exa 2.7	Newton Raphson Method	13
Exa 2.8	Newton Raphson Method	14
Exa 2.9	Newton Raphson Method	14
Exa 2.10	Newton Raphson Method	15
Exa 2.11	Newton Raphson Method	16
Exa 2.12	Iteration Formula	17
Exa 2.13	Iteration Formula	18
Exa 2.14	Iteration Formula	18
Exa 2.15	Newton Raphshon Method	19
Exa 2.16	Newton Raphshon Method	20
Exa 2.17	Newton Raphshon Method	21
Exa 2.18	Newton Raphshon Method for simultaneous equations	22
Exa 2.19	Newton Raphshon Method for simultaneous equations	23
Exa 2.20	Graeffe Method	24
Exa 2.21	Graeffe Method	25
Exa 2.22	Graeffe Method	27
Exa 2.23	Mullers Method	28
Exa 2.24	Mullers Method	30
Exa 2.25	Mullers Method	33

Exa 2.26	Mullers Method	35
Exa 2.27	Bairstow Method	37
Exa 3.1	Direct Method	40
Exa 3.2	Gaussian Elimination Method	40
Exa 3.3	Gaussian Elimination Method	41
Exa 3.4	Gaussian Elimination Method	43
Exa 3.5	Gauss Jordan Method	44
Exa 3.6	Gauss Jordan Method	45
Exa 3.7	Gauss Jordan Method	46
Exa 3.8	Triangularization Method	48
Exa 3.9	Triangularization Method	49
Exa 3.10	Triangularization Method	51
Exa 3.11	Triangularization Method	52
Exa 3.12	Triangularization Method	54
Exa 3.13	Crout Method	56
Exa 3.14	Crout Method	57
Exa 3.15	Crout Method	59
Exa 3.16	Jacobi Method	60
Exa 3.17	Gauss Seidel Method	61
Exa 3.18	Gauss Seidel Method	62
Exa 3.19	Gauss Seidel Method	62
Exa 3.20	Gauss Seidel Method	63
Exa 3.21	Relaxation Method	63
Exa 3.22	Relaxation Method	65
Exa 3.23	Relaxation Method	67
Exa 4.1	Power Method	69
Exa 4.2	Power Method	70
Exa 4.3	Power Method	71
Exa 4.4	Power Method	72
Exa 4.5	Jacobi Method	73
Exa 4.6	Jacobi Method	74
Exa 4.7	Jacobi Method	75
Exa 4.8	Jacobi Method	76
Exa 4.9	Givens Method	77
Exa 4.10	Givens Method	78
Exa 4.11	Givens Method	79
Exa 4.12	Givens Method	80
Exa 4.13	House Holder Transformation	81

Exa 4.14	House Holder Transformation	82
Exa 4.15	Strum Sequence	83
Exa 4.16	Strum Sequence	84
Exa 4.17	Gerschgorin Circles	85
Exa 5.1	Backward Difference Formula	86
Exa 5.3	Factorial Notation Method	87
Exa 5.5	Finite Differences	88
Exa 5.6	Finite Differences	89
Exa 5.11	Finite Differences	89
Exa 5.16	Finite Differences	90
Exa 5.17	Error Propagation	91
Exa 5.18	Error Propagation	92
Exa 5.20	Newtons Forward Difference Formula	93
Exa 5.21	Newtons Forward Difference Formula	95
Exa 5.22	Newtons Forward Difference Formula	96
Exa 5.23	Newtons Forward Difference Formula	97
Exa 5.24	Central Difference Derivatives	98
Exa 5.27	Central Difference Derivatives	99
Exa 5.28	Divided Difference Interpolation	100
Exa 5.29	Divided Difference Interpolation	101
Exa 5.30	Maximum Error in Interpolation	102
Exa 5.32	Divided Difference Interpolation	103
Exa 5.36	Lagranges Interpolation Method	104
Exa 5.37	Lagranges Interpolation Method	105
Exa 5.38	Lagranges Interpolation Method	106
Exa 5.39	Hermite Interpolation Method	107
Exa 5.40	Hermite Interpolation Method	107
Exa 5.41	Piecewise Cubic Hermite Interpolation Method	108
Exa 5.43	Inverse Interpolation using Newtons Forward Difference Formula	110
Exa 5.44	Inverse Interpolation using Everett Formula	112
Exa 5.45	Inverse Lagrange Method	113
Exa 5.46	Newtons Divided Difference Interpolation	113
Exa 5.47	Bessel Interpolation	114
Exa 5.48	Chebyshev Polynomial	116
Exa 5.50	Spline Interpolation	116
Exa 5.51	Spline Interpolation	117
Exa 5.52	Spline Interpolation	118

Exa 5.53	Spline Interpolation	120
Exa 5.54	Spline Interpolation	121
Exa 6.2	Least Line Square Approximation	123
Exa 6.3	Least Square Method	124
Exa 6.4	Least Square Method	125
Exa 6.5	Power Fit Method	126
Exa 6.6	Least Square Method	127
Exa 6.7	Parabola Best Fit	128
Exa 6.8	Parabola Best Fit	129
Exa 6.9	Least Square Fit	130
Exa 6.10	Least Square Fit	131
Exa 6.11	Least Square Fit	132
Exa 6.12	Least Square Fit	134
Exa 7.1	Differentiation	136
Exa 7.2	Differentiation	136
Exa 7.3	Richardson Extrapolation	137
Exa 7.4	Differentiation	138
Exa 7.5	Stirlings Central Difference Derivatives	138
Exa 7.6	Stirlings Central Difference Derivatives	140
Exa 7.7	Stirlings Central Difference Derivatives	141
Exa 7.8	Stirlings Central Difference Derivatives	143
Exa 7.9	Newtons Backward Formula	144
Exa 7.10	Lagranges Differentiation	145
Exa 7.11	Newtons Divided Difference Interpolation	146
Exa 7.12	Stirlings Central Difference Derivatives	148
Exa 7.13	Newtons Backward Formula	149
Exa 7.14	Newtons Divided Difference	150
Exa 8.1	Simpsons 1 3rd Rule	151
Exa 8.2	Simpsons 1 3rd Rule and Richardson Extrapolation	152
Exa 8.6	Simpsons 1 3rd Rule and Bessels Quadrature	154
Exa 8.7	Simpsons 1 3rd Rule	155
Exa 8.8	Rombers Method	156
Exa 8.9	Rombers Method	157
Exa 8.10	Rombers Method	158
Exa 8.11	Integration by Various Methods	160
Exa 8.12	Euler Maclaurin Methods	162
Exa 8.13	Trapezoidal and Simpsons Rule	163
Exa 8.14	Trapezoidal Rule	165

Exa 9.1	Recurrence formula	167
Exa 9.2	Recurrence formula	167
Exa 9.3	Recurrence formula	168
Exa 9.4	Recurrence formula	168
Exa 9.5	Difference Equation	169
Exa 9.6	Difference Equation	169
Exa 9.8	Recurrence formula	170
Exa 9.9	Particular Solution	171
Exa 9.10	Particular Solution	171
Exa 9.11	Particular Solution	172
Exa 9.12	Particular Solution	173
Exa 9.13	Particular Solution	174
Exa 9.14	Particular Solution	174
Exa 9.15	Particular Solution	175
Exa 9.16	Particular Solution	176
Exa 9.17	Particular Solution	176
Exa 9.18	Particular Solution	177
Exa 10.1	Taylor Method	178
Exa 10.2	Taylor Method	179
Exa 10.3	Taylor Method	180
Exa 10.4	Euler Method	181
Exa 10.5	Euler Method	181
Exa 10.6	Euler and Modified Euler Method	182
Exa 10.7	Modified Euler Method	182
Exa 10.8	Picard Method	183
Exa 10.9	Picard Method	183
Exa 10.10	Picard Method	184
Exa 10.11	Heun Method	185
Exa 10.12	Third Order Runge Kutta Method	185
Exa 10.13	Fourth Order Runge Kutta Method	186
Exa 10.14	Fourth Order Runge Kutta Method for higher order equations	186
Exa 10.15	Fourth Order Runge Kutta Method	187
Exa 10.16	Fourth Order Runge Kutta Method	188
Exa 10.17	Fourth Order Runge Kutta Method for system of 1st order equations	188
Exa 10.18	Fourth Order Runge Kutta Method for higher order equations	189

Exa 10.19	Adams Basforth formula	190
Exa 10.20	Adams Moulton formula	191
Exa 10.21	Adams formula	191
Exa 10.22	Milne Simpson Predictor Corrector Method	192
Exa 10.23	Milne Simpson Predictor Corrector Method	194
Exa 10.24	Milne Simpson Predictor Corrector Method	194
Exa 10.25	Milne Simpsons formula	195
Exa 10.26	Mullers Method	196
Exa 10.33	Numerov Method	197
Exa 10.34	Numerov Method	197
Exa 10.36	Finite Difference Method	198
Exa 10.37	Finite Difference Method	199
Exa 10.38	Finite Difference Method	200
Exa 10.39	Finite Difference Method	200
Exa 10.40	Formula Method	201
Exa 10.41	Eigenvalue Problem	203
Exa 11.1	Gauss Seidel Method	205
Exa 11.2	Gauss Seidel Method	206
Exa 11.3	Gauss Seidel Method	207
Exa 11.4	Gauss Seidel Method	208
Exa 11.5	Gauss Seidel Method	209
Exa 11.6	Gaussian Elimination Method	211
Exa 11.7	Relaxation Method	212
Exa 11.8	Relaxation Method	214
Exa 11.9	Gauss Seidel Method	216
Exa 11.10	Gauss Seidel Method	217
Exa 11.11	Eigenvalue Problem	217
Exa 11.12	Eigenvalue Problem	218
Exa 11.13	Eigenvalue Problem	218
Exa 11.14	Crank Nicolson Method	219
Exa 11.15	Bender Schmidt Method	222
Exa 11.16	Crank Nicolson Method	223
Exa 11.17	Bender Schmidt Method	226
Exa 11.18	Bender Schmidt Method	227
Exa 11.19	Gauss Seidel Method	228
Exa 11.20	Finite Difference Method	229
Exa 11.21	Finite Difference Method	230

List of Figures

5.1 Spline Interpolation	118
------------------------------------	-----

Chapter 1

Preliminaries

Scilab code Exa 1.1 Rounding off Numbers

```
1 //Example 1.1
2 //Rounding off Numbers
3 //Page no. 2
4 clc;clear;close;
5 a=[81.9773;48.365;21.385;12.865;27.553]
6 for i=1:5
7     printf('\n%g becomes %.4g\n',a(i),a(i))
8 end
```

Scilab code Exa 1.2 Relative Maximum Error

```
1 //Example 1.2
2 //Relative Maximum Error
3 //Page no. 5
4 clc;clear;close;
5 h=0.001;
6 x=1;y=1;z=1;dx=0.001;dy=0.001;dz=0.001;
7 deff('u=f(x,y,z)','u=(5*x*y^2)/z^3')
```

```

8 du=abs(f(x+h,y,z)-f(x,y,z))*dx+abs(f(x,y+h,z)-f(x,y,
    z))*dy+abs(f(x,y,z+h)-f(x,y,z))*dz;
9 du=du/h;
10 Er=du/f(x,y,z)
11 printf('\nMaximum Error = %.3f\n\nRelative maximum
    error = %.3f',du,Er)

```

Scilab code Exa 1.3 Absolute Error

```

1 //Example 1.3
2 //Absolute Error
3 //Page no. 6
4 clc;clear;close;
5 a=10;b=0.0356;c=15300;d=62000;
6 ea=0.05;eb=0.0002;ec=100;ed=500;
7 e=ea+eb+ec+ed;
8 printf('\nMaximum Absolute Error of a+b+c+d = %f\n\n
    ',e)
9 E=(c+2*ec)^3-(c+ec)^3
10 printf('
    Absolute Error of c = %f',E)

```

Chapter 2

Non Linear Equations

Scilab code Exa 2.1 Bisection Method

```
1 //Example 2.1
2 //Bisection Method
3 //Page no. 14
4 clc; clear; close;
5 deff( 'y=f(x)', 'y=x^3+x^2-1' )
6 x1=0.5; x2=1; e=0.0001; i=0;
7 printf( 'Iteration \tx1\t\ttx2\t\ttz\t\tf(z)\n' )
8 printf( '
          n' )
9 while abs(x1-x2)>e
10     z=(x1+x2)/2
11     printf( '          %i\t\t%f\t%f\t%f\t%f\n', i, x1, x2, z, f
          (z))
12     if f(z)*f(x1)>0
13         x1=z
14     else
15         x2=z
16     end
17     i=i+1
18 end
```



```

1 //Example 2.3
2 //Regula Falsi Method
3 //Page no. 17
4 clc;clear;close;
5 deff('y=f(x)', 'y=x^3-5*x-7')
6 x1=2;x2=3;e=0.01
7 printf('n\tx1\t\tf(x1)\t\tx2\t\tf(x2)\t\tx3\t\tf(x3)
      ')
8 printf('\n
      n')
9 for i=0:19
10     x3=x2*f(x1)/(f(x1)-f(x2))+x1*f(x2)/(f(x2)-f(x1))
11     printf(' %i\t%f\t%f\t%f\t%f\t%f\n',i,x1,f(x1)
            ),x2,f(x2),x3,f(x3))
12     if f(x1)*f(x3)>0 then
13         x1=x3
14     else
15         x2=x3
16     end
17     if abs(f(x3))<e then
18         break
19     end
20 end
21 printf('\n\nThus the root is %.3f correct upto three
      places of decimal',x3)

```

Scilab code Exa 2.4 Regula Falsi Method

```

1 //Example 2.4
2 //Regula Falsi Method
3 //Page no. 18
4 clc;clear;close;

```



```

5 deff('y=f(x)', 'y=x*log10(x)-1.2')
6 x1=2; x2=3; e=0.000001
7 printf('n\tx1\t\tf(x1)\t\tx2\t\tf(x2)\t\tx3\t\tf(x3)
      ')
8 printf('\n
      n')
9 for i=0:19
10     x3=x2*f(x1)/(f(x1)-f(x2))+x1*f(x2)/(f(x2)-f(x1))
11     printf(' %i\t%f\t%f\t%f\t%f\t%f\t%f\n', i, x1, f(x1)
      ), x2, f(x2), x3, f(x3))
12     if f(x1)*f(x3)>0 then
13         x1=x3
14     else
15         x2=x3
16     end
17     if abs(f(x3))<e then
18         break
19     end
20 end
21 printf('\n\nThus the root is %.3f correct upto three
      places of decimal', x3)

```

Scilab code Exa 2.5 Regula Falsi Method

```

1 //Example 2.5
2 //Regula Falsi Method
3 //Page no. 19
4 clc; clear; close;
5 deff('y=f(x)', 'y=log10(x)-cos(x)')
6 x1=1; x2=1.5; e=0.00000001
7 printf('n\tx1\t\tf(x1)\t\tx2\t\tf(x2)\t\tx3\t\tf(x3)
      ')

```

```

8 printf( '\n
    n ')
9 for i=0:19
10     x3=x2*f(x1)/(f(x1)-f(x2))+x1*f(x2)/(f(x2)-f(x1))
11     printf( ' %i\t%f\t%f\t%f\t%f\t%f\t%f\n', i, x1, f(x1)
    ), x2, f(x2), x3, f(x3))
12     if f(x1)*f(x3)>0 then
13         x1=x3
14     else
15         x2=x3
16     end
17     if abs(f(x3))<e then
18         break
19     end
20 end
21 printf( '\n\nThus the root is %.4f correct upto four
    places of decimal', x3)

```

Scilab code Exa 2.6 Secant Method

```

1 //Example 2.6
2 //Secant Method
3 //Page no. 19
4 clc;clear;close;
5 deff( 'x=f(x)', 'x=cos(x)-x*exp(x)')
6 deff( 'x=f1(x)', 'x=-sin(x)-exp(x)-x*exp(x)')
7 printf( 'n\txn\t\tf(xn)\t\tXn+1\t\tf(Xn+1)\t\tXn+2\t\t
    tError\n')
8 printf( '
    n ')
9 x0=0; x1=1; e=0.00001

```

```

10 for i=1:6
11     x2=x1-f(x1)*(x1-x0)/(f(x1)-f(x0))
12     e1=abs(x0-x2)
13     printf(' %i\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\n',i-1,x0,f(x0),x1,f(x1),x2,e1)
14     x0=x1;
15     x1=x2
16     if abs(x0)<e then
17         break;
18     end
19 end
20 printf('\n\nTherefore, the root is %.4f correct upto
    4 decimal places ',x2)

```

Scilab code Exa 2.7 Newton Raphson Method

```

1 //Example 2.7
2 //Newton Raphson Method
3 //Page no. 21
4 clc;clear;close;
5 deff('x=f(x)', 'x=x^3-5*x+3')
6 deff('x=f1(x)', 'x=3*x^2-5')
7 printf('n\txn\t\t\tf(xn)\t\tf1(xn)\t\tXn+1\t\tError\n
    ')
8 printf('
    n')
9 x0=1;e=0.00001
10 for i=1:6
11     x1=x0-f(x0)/f1(x0)
12     e1=abs(x0-x1)
13     printf(' %i\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\n
        ',i-1,x0,f(x0),f1(x0),x1,e1)
14     x0=x1;
15     if abs(x0)<e then

```

```

16         break;
17     end
18 end
19 printf('\n\nTherefore, this is convergent')

```

Scilab code Exa 2.8 Newton Raphson Method

```

1 //Example 2.8
2 //Newton Raphson Method
3 //Page no. 21
4 clc;clear;close;
5 deff('x=f(x)', 'x=x^4-3*x^3+2*x^2+2*x-7')
6 deff('x=f1(x)', 'x=4*x^3-9*x^2+4*x+2')
7 printf('n\txn\t\t\tf(xn)\t\tf1(xn)\t\tXn+1\t\tError\n
      ')
8 printf('
      n')
9 x0=2.1;e=0.00001
10 for i=1:6
11     x1=x0-f(x0)/f1(x0)
12     e1=abs(x0-x1)
13     printf(' %i\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\n
      ',i-1,x0,f(x0),f1(x0),x1,e1)
14     x0=x1;
15     if abs(x0)<e then
16         break;
17     end
18 end
19 printf('\n\nTherefore, this is convergent and root =
      %.8f',x0)

```

Scilab code Exa 2.9 Newton Raphson Method

```

1 //Example 2.9
2 //Newton Raphson Method
3 //Page no. 22
4 clc;clear;close;
5 deff('x=f(x)', 'x=exp(x)-5*x')
6 deff('x=f1(x)', 'x=exp(x)-5')
7 printf('n\txn\t\t\tf(xn)\t\tf1(xn)\t\tXn+1\t\tError\n
      ')
8 printf('
      n')
9 x0=0.4;e=0.00001
10 for i=1:5
11     x1=x0-f(x0)/f1(x0)
12     e1=abs(x0-x1)
13     printf(' %i\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\n
      ',i-1,x0,f(x0),f1(x0),x1,e1)
14     x0=x1;
15     if abs(x0)<e then
16         break;
17     end
18 end
19 printf('\n\nTherefore, this is convergent and the
      root is %.10f',x0)

```

Scilab code Exa 2.10 Newton Raphson Method

```

1 //Example 2.10
2 //Newton Raphson Method
3 //Page no. 22
4 clc;clear;close;
5 deff('x=f(x)', 'x=3*x-cos(x)-1')
6 deff('x=f1(x)', 'x=3+sin(x)')
7 printf('n\txn\t\t\tf(xn)\t\tf1(xn)\t\tXn+1\t\tError\n
      ')

```

```

8 printf('
    n')
9 x0=0.6;e=0.00001
10 for i=1:3
11     x1=x0-f(x0)/f1(x0)
12     e1=abs(x0-x1)
13     printf(' %i\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\n
            ',i-1,x0,f(x0),f1(x0),x1,e1)
14     x0=x1;
15     if abs(x0)<e then
16         break;
17     end
18 end
19 printf('\n\nTherefore, this is convergent and root =
        %.4f',x0)

```

Scilab code Exa 2.11 Newton Raphson Method

```

1 //Example 2.11
2 //Newton Raphson Method
3 //Page no. 23
4 clc;clear;close;
5 deff('x=f(x)', 'x=x*sin(x)+cos(x)')
6 deff('x=f1(x)', 'x=x*cos(x)')
7 printf('n\txn\t\t\tf(xn)\t\tf1(xn)\t\tXn+1\t\tError\n
        ')
8 printf('
    n')
9 x0=%pi;e=0.00001
10 for i=1:6
11     x1=x0-f(x0)/f1(x0)

```

```

12     e1=abs(x0-x1)
13     printf(' %i\t%.10 f\t%.10 f\t%.10 f\t%.10 f\t%.10 f\n
           ',i-1,x0 ,f(x0) ,f1(x0) ,x1 ,e1)
14     x0=x1;
15     if abs(x0)<e then
16         break;
17     end
18 end
19 printf('\n\nTherefore , the root is %.4 f',x0)

```

Scilab code Exa 2.12 Iteration Formula

```

1 //Example 2.12
2 //Iteration Formula
3 //Page no. 28
4 clc;clear;close;
5 deff('x=f(x)', 'x=1/sqrt(1+x)')
6 printf('n\txn\t\t\tf(xn)\t\tXn+1\t\tError\n')
7 printf('
           n')
8 x0=0.75;e=0.00001
9 for i=1:8
10     x1=f(x0)
11     e1=abs(x0-x1)
12     printf(' %i\t%.10 f\t%.10 f\t%.10 f\t%.10 f\n',i-1,
           x0 ,f(x0) ,x1 ,e1)
13     x0=x1;
14     if abs(x0)<e then
15         break;
16     end
17 end
18 printf('\n\nTherefore , the root is %.6 f',x0)

```

Scilab code Exa 2.13 Iteration Formula

```
1 //Example 2.13
2 //Iteration Formula
3 //Page no. 28
4 clc;clear;close;
5 deff('x=f(x)', 'x=(log10(x)+7)/2')
6 printf('n\txn\t\t\tf(xn)\t\tXn+1\t\tError\n')
7 printf('
-----
n')
8 x0=3.8;e=0.00001
9 for i=1:6
10     x1=f(x0)
11     e1=abs(x0-x1)
12     printf(' %i\t%.10f\t%.10f\t%.10f\t%.10f\n',i-1,
13           x0,f(x0),x1,e1)
14     x0=x1;
15     if abs(x0)<e then
16         break;
17     end
18 printf('\n\nTherefore, the root is %.6f',x0)
```

Scilab code Exa 2.14 Iteration Formula

```
1 //Example 2.14
2 //Iteration Formula
3 //Page no. 29
4 clc;clear;close;
5 deff('x=f(x)', 'x=exp(x)/5')
6 printf('n\txn\t\t\tf(xn)\t\tXn+1\t\tError\n')
```



```

7 printf('
      n')
8 x0=0.3;e=0.00001
9 for i=1:11
10     x1=f(x0)
11     e1=abs(x0-x1)
12     printf(' %i\t%.10f\t%.10f\t%.10f\t%.10f\n',i-1,
            x0,f(x0),x1,e1)
13     x0=x1;
14     if abs(x0)<e then
15         break;
16     end
17 end
18 printf('\n\nTherefore, the root is %.6f',x0)

```

Scilab code Exa 2.15 Newton Raphshon Method

```

1 //Example 2.15
2 //Newton Raphshon Method
3 //Page no. 30
4 clc;clear;close;
5 h=0.001
6 deff('x=f(x)', 'x=x^4-5*x^3-12*x^2+76*x-79')
7 deff('x=f1(x)', 'x=(f(x+h)-f(x))/h')
8 deff('x=f2(x)', 'x=(f1(x+h)-f1(x))/h')
9 printf('n\txn\t\t\tf(xn)\t\tXn+1\t\tError\n')
10 printf('
      n')
11 x=2;e=0.00001
12 e2=sqrt((-2*f(x))/f2(x))
13 x0=x+e2
14 for i=1:4
15     x1=x0-f(x0)/f1(x0)

```

```

16     e1=abs(x0-x1)
17     printf(' %i\t%.10 f\t%.10 f\t%.10 f\t%.10 f\n',i-1,
           x0,f(x0),x1,e1)
18     x0=x1;
19     if abs(x0)<e then
20         break;
21     end
22 end
23 printf('\n\nTherefore, the root is %.4f\n\n\n\n',x0)
24
25 x0=x-e2
26 for i=1:4
27     x1=x0-f(x0)/f1(x0)
28     e1=abs(x0-x1)
29     printf(' %i\t%.10 f\t%.10 f\t%.10 f\t%.10 f\n',i-1,
           x0,f(x0),x1,e1)
30     x0=x1;
31     if abs(x0)<e then
32         break;
33     end
34 end
35 printf('\n\nTherefore, the root is %.4f',x0)

```

Scilab code Exa 2.16 Newton Raphshon Method

```

1 //Example 2.16
2 //Newton Raphshon Method
3 //Page no. 31
4 clc;clear;close;
5 h=0.001
6 deff('x=f(x)', 'x=x^3-5*x^2+8*x-4')
7 deff('x=f1(x)', 'x=(f(x+h)-f(x))/h')
8 deff('x=f2(x)', 'x=(f1(x+h)-f1(x))/h')
9 printf('n\txn\t\t\tf(xn)\t\tXn+1\t\tError\n')
10 printf('

```

```

n')
11 x0=1.8;e=0.00001
12 for i=1:10
13     x1=x0-2*f(x0)/f1(x0)
14     e1=abs(x0-x1)
15     printf(' %i\t%.10f\t%.10f\t%.10f\t%.10f\n',i-1,
            x0,f(x0),x1,e1)
16     x0=x1;
17     if abs(x0)<e then
18         break;
19     end
20 end
21 printf('\n\nTherefore, the root is %.4f\n\n\n\n',x0)

```

Scilab code Exa 2.17 Newton Raphshon Method

```

1 //Example 2.17
2 //Newton Raphshon Method
3 //Page no. 32
4 clc;clear;close;
5 h=0.001
6 deff('x=f(x)', 'x=x^3-x^2-x+1')
7 deff('x=f1(x)', 'x=(f(x+h)-f(x))/h')
8 deff('x=f2(x)', 'x=(f1(x+h)-f1(x))/h')
9 printf('n\txn\t\t\tf(xn)\t\tXn+1\t\tError\n')
10 printf('
n')
11 x0=0.8;e=0.00001
12 for i=1:10
13     x1=x0-2*f(x0)/f1(x0)
14     e1=abs(x0-x1)
15     printf(' %i\t%.10f\t%.10f\t%.10f\t%.10f\n',i-1,
            x0,f(x0),x1,e1)

```

```

16     x0=x1;
17     if abs(x0)<e then
18         break;
19     end
20 end
21 printf('\n\nTherefore , the root is %.4f\n\n\n\n',x0)

```

Scilab code Exa 2.18 Newton Raphshon Method for simultaneous equations

```

1 //Example 2.18
2 //Newton Raphshon Method for simultaneous equations
3 //Page no. 33
4 clc;clear;close;
5
6 deff('y=f1(x,y)', 'y=x+3*log10(x)-y^2');
7 deff('y=f2(x,y)', 'y=2*x^2-x*y-5*x+1');
8 h=0.01;
9 function u=f3(x,y,z)
10     if z==1 then
11         u=(f1(x+h,y)-f1(x,y))/h
12     elseif z==2
13         u=(f1(x,y+h)-f1(x,y))/h
14     elseif z==3
15         u=(f2(x+h,y)-f2(x,y))/h
16     else
17         u=(f2(x,y+h)-f2(x,y))/h
18     end
19 endfunction
20 x=3.4;y=2.2;
21 for i=1:4
22     printf('\n\tx%i = %g\t\ty%i = %g\n',i-1,x,i-1,y)
23     printf('\nfi(x0,y0) = %g',f1(x,y));
24     printf('\nomega(x0,y0) = %g',f2(x,y));
25     printf('\nd(fi)/dx = %g',f3(x,y,1));

```

```

26 printf( '\nd(fi)/dy = %g', f3(x,y,2));
27 printf( '\nd(omega)/dx = %g', f3(x,y,3));
28 printf( '\nd(omega)/dy = %g', f3(x,y,4));
29 A=[f3(x,y,1), f3(x,y,2); f3(x,y,3), f3(x,y,4)];
30 B=[-f1(x,y); -f2(x,y)];
31 C=inv(A)*B;
32 x=x+C(1);
33 y=y+C(2);
34 printf( '\n\n\th%i = %g\t\tk%i = %g\n\n', i, C(1), i, C
      (2));
35 end
36 printf( '\n\tx%i = %g\t\tty%i = %g\n\n\n\nNote :
      Computational Errors in Book', i, x, i, y)

```

Scilab code Exa 2.19 Newton Raphshon Method for simultaneous equations

```

1 //Example 2.19
2 //Newton Raphshon Method for simultaneous equations
3 //Page no. 35
4 clc; clear; close;
5
6 deff( 'y=f1(x,y)', 'y=1+x^2-y^2' );
7 deff( 'y=f2(x,y)', 'y=2*x*y' );
8 h=0.01;
9 function u=f3(x,y,z)
10     if z==1 then
11         u=(f1(x+h,y)-f1(x,y))/h
12     elseif z==2
13         u=(f1(x,y+h)-f1(x,y))/h
14     elseif z==3
15         u=(f2(x+h,y)-f2(x,y))/h
16     else
17         u=(f2(x,y+h)-f2(x,y))/h
18     end

```

```

19 endfunction
20 x=0.5;y=0.5;
21 for i=1:3
22     printf('\n\tx%i = %g\t\ty%i = %g\n',i-1,x,i-1,y)
23     printf('\nfi(x0,y0) = %g',f1(x,y));
24 printf('\nomega(x0,y0) = %g',f2(x,y));
25 printf('\nd(fi)/dx = %g',f3(x,y,1));
26 printf('\nd(fi)/dy = %g',f3(x,y,2));
27 printf('\nd(omega)/dx = %g',f3(x,y,3));
28 printf('\nd(omega)/dy = %g',f3(x,y,4));
29 A=[f3(x,y,1),f3(x,y,2);f3(x,y,3),f3(x,y,4)];
30 B=[-f1(x,y);-f2(x,y)];
31 C=inv(A)*B;
32 x=x+C(1);
33 y=y+C(2);
34 printf('\n\n\th%i = %g\t\tk%i = %g\n\n',i,C(1),i,C
(2));
35 end
36 printf('\n\tx%i = %g\t\ty%i = %g\n',i,x,i,y)

```

Scilab code Exa 2.20 Graeffe Method

```

1 //Example 2.20
2 //Graeffe Method
3 //Page no. 38
4 clc;clear;close;
5
6 a=[1,-5,-17,20]
7 k=0;
8 for k=2:6
9     for i=1:4
10        a(k,i)=(-1)^(i-1)*(a(k-1,i))^2
11        j=1;
12        while i+j<5 & i+j>2
13            a(k,i)=a(k,i)+(-1)^(i-j-1)*2*(a(k-1,i-j)

```



```

        ,2))^(1/(2^(i-1))),a(i,4),abs(a(i,4)/a(i,3))
        ^(1/(2^(i-1))))
23 end
24 for i=5:6
25     printf(' %i\t%g\t%.4g\t%.5g\t\t%.9g\t%.8g\t%.7g\t
        t%.10g\n',i-1,a(i,1),a(i,2),abs(a(i,2)/a(i,1)
        )^(1/(2^(i-1))),a(i,3),abs(a(i,3)/a(i,2))
        ^(1/(2^(i-1))),a(i,4),abs(a(i,4)/a(i,3))
        ^(1/(2^(i-1))))
26 end
27 printf('\n\nThe Absolute Values of the roots are %g,
        %.8g and %g',abs(a(i,2)/a(i,1))^(1/(2^(i-1))),
        abs(a(i,3)/a(i,2))^(1/(2^(i-1))),abs(a(i,4)/a(i
        ,3))^(1/(2^(i-1))))

```

Scilab code Exa 2.23 Mullers Method

```

1 //Example 2.23
2 //Mullers Method
3 //Page no. 41
4 clc;clear;close;
5
6 deff('y=f(x)', 'y=x^3-x-1')
7 zi=[1;2;3];
8 s=["i","z2","z0","z1","f2","f0","f1","a0","a1","a2",
    "zr+","zr-"]
9 li(1)=(zi(3,1)-zi(2,1))/(zi(2,1)-zi(1,1))
10 hi(1)=zi(3,1)-zi(2,1);
11 for i=2:6
12     for j=1:3
13         fz(j,i-1)=f(zi(j,i-1))
14     end
15     di(i-1)=1+li(i-1)
16     gi(i-1)=fz(1,i-1)*li(i-1)^2-fz(2,i-1)*di(i-1)^2+
        fz(3,i-1)*(li(i-1)+di(i-1))

```

```

17     D1(i-1)=gi(i-1)+sqrt(gi(i-1)^2-4*fz(3,i-1)*di(i
        -1)*li(i-1)*(fz(1,i-1)*li(i-1)-fz(2,i-1)*di(i
        -1)+fz(3,i-1)))
18     D2(i-1)=gi(i-1)-sqrt(gi(i-1)^2-4*fz(3,i-1)*di(i
        -1)*li(i-1)*(fz(1,i-1)*li(i-1)-fz(2,i-1)*di(i
        -1)+fz(3,i-1)))
19     if abs(D1(i-1))>abs(D2(i-1)) then
20         li(i)=-2*fz(3,i-1)*di(i-1)/D1(i-1)
21     else
22         li(i)=-2*fz(3,i-1)*di(i-1)/D2(i-1)
23     end
24     hi(i)=li(i)*hi(i-1);
25     z(i-1)=zi(3,i-1)+hi(i)
26     for j=1:2
27         zi(j,i)=zi(j+1,i-1)
28     end
29     zi(3,i)=z(i-1)
30 end
31 for i=1:12
32     if i==1 then
33         printf(s(i))
34         for j=1:5
35             printf('\t\t\t%i',j-1)
36         end
37     elseif i<=4
38         printf('\n %s',s(i))
39         for j=1:5
40             printf('\t\t%.10f',zi(i-1,j))
41         end
42     elseif i<=7
43         printf('\n %s',s(i))
44         for j=1:5
45             printf('\t\t%.10f',fz(i-4,j))
46         end
47     elseif i<=8
48         printf('\n %s',s(i))
49         for j=1:5
50             printf('\t\t%.10f',li(j))

```

```

51         end
52     elseif i<=9
53         printf('\n %s',s(i))
54         for j=1:5
55             printf('\t\t%.10f',di(j))
56         end
57     elseif i<=10
58         printf('\n %s',s(i))
59         for j=1:5
60             printf('\t\t%.10f',gi(j))
61         end
62     elseif i<=11
63         printf('\n %s',s(i))
64         for j=1:5
65             printf('\t\t%.10f',z(j))
66         end
67     elseif i<=12
68         printf('\n %s',s(i))
69         for j=1:5
70             printf('\t\t%.10f',zi(j))
71         end
72     end
73 end
74 printf('\n\nAt the end of the %i iteration , the root
of the equation is %.10f',j-2,z(j))

```

Scilab code Exa 2.24 Mullers Method

```

1 //Example 2.24
2 //Mullers Method
3 //Page no. 42
4 clc;clear;close;
5
6 deff('y=f(x)', 'y=x^3-x-2')
7 zi=[1.4;1.5;1.6];

```

```

8 s=["i","z2","z0","z1","f2","f0","f1","a0","a1","a2",
    "zr+","zr-"]
9 li(1)=(zi(3,1)-zi(2,1))/(zi(2,1)-zi(1,1))
10 hi(1)=zi(3,1)-zi(2,1);
11 for i=2:6
12     for j=1:3
13         fz(j,i-1)=f(zi(j,i-1))
14     end
15     di(i-1)=1+li(i-1)
16     gi(i-1)=fz(1,i-1)*li(i-1)^2-fz(2,i-1)*di(i-1)^2+
        fz(3,i-1)*(li(i-1)+di(i-1))
17     D1(i-1)=gi(i-1)+sqrt(gi(i-1)^2-4*fz(3,i-1)*di(i-1)*li(i-1)*(fz(1,i-1)*li(i-1)-fz(2,i-1)*di(i-1)+fz(3,i-1)))
18     D2(i-1)=gi(i-1)-sqrt(gi(i-1)^2-4*fz(3,i-1)*di(i-1)*li(i-1)*(fz(1,i-1)*li(i-1)-fz(2,i-1)*di(i-1)+fz(3,i-1)))
19     if abs(D1(i-1))>abs(D2(i-1)) then
20         li(i)=-2*fz(3,i-1)*di(i-1)/D1(i-1)
21     else
22         li(i)=-2*fz(3,i-1)*di(i-1)/D2(i-1)
23     end
24     hi(i)=li(i)*hi(i-1);
25     z(i-1)=zi(3,i-1)+hi(i)
26     for j=1:2
27         zi(j,i)=zi(j+1,i-1)
28     end
29     zi(3,i)=z(i-1)
30 end
31 for i=1:12
32     if i==1 then
33         printf(s(i))
34         for j=1:5
35             printf('\t\t\t%i',j-1)
36         end
37     elseif i<=4
38         printf('\n %s',s(i))
39         for j=1:5

```

```

40         printf('\t\t%.10f',zi(i-1,j))
41     end
42 elseif i<=7
43     printf('\n %s',s(i))
44     for j=1:5
45         printf('\t\t%.10f',fz(i-4,j))
46     end
47 elseif i<=8
48     printf('\n %s',s(i))
49     for j=1:5
50         printf('\t\t%.10f',li(j))
51     end
52 elseif i<=9
53     printf('\n %s',s(i))
54     for j=1:5
55         printf('\t\t%.10f',di(j))
56     end
57 elseif i<=10
58     printf('\n %s',s(i))
59     for j=1:5
60         printf('\t\t%.10f',gi(j))
61     end
62 elseif i<=11
63     printf('\n %s',s(i))
64     for j=1:5
65         printf('\t\t%.10f',z(j))
66     end
67 elseif i<=12
68     printf('\n %s',s(i))
69     for j=1:5
70         printf('\t\t%.10f',zi(j))
71     end
72 end
73 end
74 printf('\n\nAt the end of the %i iteration , the root
of the equation is %.10f',j-2,z(j))

```

Scilab code Exa 2.25 Mullers Method

```
1 //Example 2.25
2 //Mullers Method
3 //Page no. 43
4 clc;clear;close;
5
6 deff('y=f(x)', 'y=cos(x)-x*exp(x)')
7 zi=[-1;0;1];
8 s=["i", "z2", "z0", "z1", "f2", "f0", "f1", "a0", "a1", "a2",
    "zr+", "zr-"]
9 li(1)=(zi(3,1)-zi(2,1))/(zi(2,1)-zi(1,1))
10 hi(1)=zi(3,1)-zi(2,1);
11 for i=2:7
12     for j=1:3
13         fz(j,i-1)=f(zi(j,i-1))
14     end
15     di(i-1)=1+li(i-1)
16     gi(i-1)=fz(1,i-1)*li(i-1)^2-fz(2,i-1)*di(i-1)^2+
        fz(3,i-1)*(li(i-1)+di(i-1))
17     D1(i-1)=gi(i-1)+sqrt(gi(i-1)^2-4*fz(3,i-1)*di(i-1)
        *li(i-1)*(fz(1,i-1)*li(i-1)-fz(2,i-1)*di(i-1)
        +fz(3,i-1)))
18     D2(i-1)=gi(i-1)-sqrt(gi(i-1)^2-4*fz(3,i-1)*di(i-1)
        *li(i-1)*(fz(1,i-1)*li(i-1)-fz(2,i-1)*di(i-1)
        +fz(3,i-1)))
19     if abs(D1(i-1))>abs(D2(i-1)) then
20         li(i)=-2*fz(3,i-1)*di(i-1)/D1(i-1)
21     else
22         li(i)=-2*fz(3,i-1)*di(i-1)/D2(i-1)
23     end
24     hi(i)=li(i)*hi(i-1);
25     z(i-1)=zi(3,i-1)+hi(i)
26     for j=1:2
```

```

27         zi(j,i)=zi(j+1,i-1)
28     end
29     zi(3,i)=z(i-1)
30 end
31 for i=1:12
32     if i==1 then
33         printf(s(i))
34         for j=1:6
35             printf('\t\t\t%i',j-1)
36         end
37     elseif i<=4
38         printf('\n %s',s(i))
39         for j=1:6
40             printf('\t\t%.10f',zi(i-1,j))
41         end
42     elseif i<=7
43         printf('\n %s',s(i))
44         for j=1:6
45             printf('\t\t%.10f',fz(i-4,j))
46         end
47     elseif i<=8
48         printf('\n %s',s(i))
49         for j=1:6
50             printf('\t\t%.10f',li(j))
51         end
52     elseif i<=9
53         printf('\n %s',s(i))
54         for j=1:6
55             printf('\t\t%.10f',di(j))
56         end
57     elseif i<=10
58         printf('\n %s',s(i))
59         for j=1:6
60             printf('\t\t%.10f',gi(j))
61         end
62     elseif i<=11
63         printf('\n %s',s(i))
64         for j=1:6

```



```

65         printf('\t\t%.10f',z(j))
66     end
67 elseif i<=12
68     printf('\n %s',s(i))
69     for j=1:6
70         printf('\t\t%.10f',zi(j))
71     end
72 end
73 end
74 printf('\n\nAt the end of the %i iteration , the root
of the equation is %.10f',j-2,z(j))

```

Scilab code Exa 2.26 Mullers Method

```

1 //Example 2.26
2 //Mullers Method
3 //Page no. 44
4 clc;clear;close;
5
6 deff('y=f(x)', 'y=x^3-x^2-x-1')
7 zi=[0;1;2];
8 s=["i", "z2", "z0", "z1", "f2", "f0", "f1", "a0", "a1", "a2",
    "zr+", "zr-"]
9 li(1)=(zi(3,1)-zi(2,1))/(zi(2,1)-zi(1,1))
10 hi(1)=zi(3,1)-zi(2,1);
11 for i=2:7
12     for j=1:3
13         fz(j,i-1)=f(zi(j,i-1))
14     end
15     di(i-1)=1+li(i-1)
16     gi(i-1)=fz(1,i-1)*li(i-1)^2-fz(2,i-1)*di(i-1)^2+
        fz(3,i-1)*(li(i-1)+di(i-1))
17     D1(i-1)=gi(i-1)+sqrt(gi(i-1)^2-4*fz(3,i-1)*di(i-1)*li(i-1)*(fz(1,i-1)*li(i-1)-fz(2,i-1)*di(i-1)+fz(3,i-1)))

```

```

18     D2(i-1)=gi(i-1)-sqrt(gi(i-1)^2-4*fz(3,i-1)*di(i
        -1)*li(i-1)*(fz(1,i-1)*li(i-1)-fz(2,i-1)*di(i
        -1)+fz(3,i-1)))
19     if abs(D1(i-1))>abs(D2(i-1)) then
20         li(i)=-2*fz(3,i-1)*di(i-1)/D1(i-1)
21     else
22         li(i)=-2*fz(3,i-1)*di(i-1)/D2(i-1)
23     end
24     hi(i)=li(i)*hi(i-1);
25     z(i-1)=zi(3,i-1)+hi(i)
26     for j=1:2
27         zi(j,i)=zi(j+1,i-1)
28     end
29     zi(3,i)=z(i-1)
30 end
31 for i=1:12
32     if i==1 then
33         printf(s(i))
34         for j=1:6
35             printf('\t\t\t%i',j-1)
36         end
37     elseif i<=4
38         printf('\n %s',s(i))
39         for j=1:6
40             printf('\t\t%.10f',zi(i-1,j))
41         end
42     elseif i<=7
43         printf('\n %s',s(i))
44         for j=1:6
45             printf('\t\t%.10f',fz(i-4,j))
46         end
47     elseif i<=8
48         printf('\n %s',s(i))
49         for j=1:6
50             printf('\t\t%.10f',li(j))
51         end
52     elseif i<=9
53         printf('\n %s',s(i))

```

```

54         for j=1:6
55             printf( '\t\t%.10f', di(j))
56         end
57     elseif i<=10
58         printf( '\n %s', s(i))
59         for j=1:6
60             printf( '\t\t%.10f', gi(j))
61         end
62     elseif i<=11
63         printf( '\n %s', s(i))
64         for j=1:6
65             printf( '\t\t%.10f', z(j))
66         end
67     elseif i<=12
68         printf( '\n %s', s(i))
69         for j=1:6
70             printf( '\t\t%.10f', zi(j))
71         end
72     end
73 end
74 printf( '\n\nAt the end of the %i iteration , the root
of the equation is %.10f', j-2, z(j))

```

Scilab code Exa 2.27 Bairstow Method

```

1 //Example 2.27
2 //Bairstow Method
3 //Page no. 48
4 clc;clear;close;
5 deff( 'y=f(x,p,q)', 'y=x^4+5*x^3+p*x^2-5*x-9')
6 n=4;
7 a=[1,5,3,-5,-9];
8 p0=a(3);q0=a(4);
9 b(1)=0;b(2)=1;c(1)=0;c(2)=1;
10 for j=1:4

```

```

11     for i=1:5
12     printf('\t\t%i',a(i))
13 end
14     for k=3:6
15         b(k)=a(k-1)-p0*b(k-1)-q0*b(k-2)
16         c(k)=b(k)-p0*c(k-1)-q0*c(k-2)
17     end
18
19
20 printf('\n  %.4f\t',-p0);
21 for i=1:4
22     printf('\t\t%.4f',-p0*b(i+1))
23 end
24 printf('\n  %.4f\t\t\t',-q0);
25 for i=1:3
26     printf('\t\t%.4f',-q0*b(i+1))
27 end
28 printf('\n
    n')
29 for i=1:5
30     printf('\t\t%.2f=b%i',b(i+1),i-1)
31 end
32 printf('\n  %.4f\t',-p0);
33 for i=1:3
34     printf('\t\t%.4f',-p0*c(i+1))
35 end
36 printf('\n  %.4f\t\t\t',-q0);
37 for i=1:2
38     printf('\t\t%.4f',-q0*c(i+1))
39 end
40 printf('\n
    n')
41 for i=1:4
42     printf('\t\t%.2f=c%i',c(i+1),i-1)
43 end
44 printf('\n\n')

```

```

45  cb=c(n+1)-b(n+1);
46      dp=-(b(n+2)*c(n-1)-b(n+1)*c(n))/(c(n)^2-cb*c(n
      -1))
47      dq=-(b(n+1)*cb-b(n+2)*c(n))/(c(n)^2-cb*c(n-1))
48  p0=p0+dp;
49  q0=q0+dq;
50  printf('\n dp = %.6f\t\tdq = %.6f\n p%i = %.6f\t
      t\tq%i = %.6f\n\n\n',dp,dq,j,p0,j,q0);
51  end

```

Chapter 3

Linear System of Equations

Scilab code Exa 3.1 Direct Method

```
1 //Example 3.1
2 //Direct Method
3 //Page no. 3.1
4 clc;clear;close;
5
6 A=[1,1,2;1,2,3;2,3,1];           //Parameter Matrix
7 B=[1;1;2]
8 C=inv(A)*B;
9 disp(C,"Solution Matrix = ")
```

Scilab code Exa 3.2 Gaussian Elimination Method

```
1 //Example 3.2
2 //Gaussian Elimination Method
3 //Page no. 54
4 clc;clear;close;
5
6 A=[2,4,-6,-4;1,5,3,10;1,3,2,5]; //
   augmented matrix
```

```

7
8 //triangularization
9 for i=1:3
10     for j=1:4
11         if i==1 then
12             B(i,j)=A(i,j)
13         elseif i==2
14             B(i,j)=A(i,j)-A(i,1)*A(i-1,j)/A(1,1)
15             B(i+1,j)=A(i+1,j)-A(i+1,1)*A(i-1,j)/A
                (1,1)
16         elseif i==3
17             if j==1 then
18                 A=B
19             end
20             B(i,j)=B(i,j)-A(i,2)*B(i-1,j)/B(2,2)
21         end
22     end
23 end
24 disp(A, 'Augmented Matrix=')
25 disp(B, 'Triangulated Matrix=')
26 //back substitution
27 x(3)=B(3,4)/B(3,3);
28 printf( '\nx(3)= %i\n', x(3))
29 for i=2:-1:1
30     k=0
31     for j=i+1:3
32         k=k+B(i,j)*x(j)
33     end
34     x(i)=(1/B(i,i))*(B(i,4)-k)
35     printf( '\nx(%i)= %i\n', i, x(i))
36 end

```

Scilab code Exa 3.3 Gaussian Elimination Method

```
1 //Example 3.3
```

```

2 //Gaussian Elimination Method
3 //Page no. 54
4 clc;clear;close;
5
6 A=[10,-7,3,5,6;-6,8,-1,-4,5;3,1,4,11,2;5,-9,-2,4,7];
   //augmented matrix
7 disp(A,'Augmented Matrix=')
8 C=A;
9 //triangularization
10 for i=1:4
11     for j=1:5
12         if i==1 then
13             B(i,j)=A(i,j)
14         elseif i==2
15             B(i,j)=A(i,j)-A(i,1)*A(i-1,j)/A(1,1)
16             B(i+1,j)=A(i+1,j)-A(i+1,1)*A(i-1,j)/A
               (1,1)
17             B(i+2,j)=A(i+2,j)-A(i+2,1)*A(i-1,j)/A
               (1,1)
18         elseif i==3
19             if j==1 then
20                 C=B
21             else
22                 B(i,j)=B(i,j)-C(i,2)*B(i-1,j)/B(2,2)
23                 B(i+1,j)=C(i+1,j)-C(i+1,2)*C(i-1,j)/
                   C(2,2)
24             end
25         else
26             if j==1 then
27                 C=B
28             end
29                 B(i,j)=B(i,j)-C(i,3)*B(i-1,j)/B(3,3)
30         end
31     end
32 end
33
34 disp(B,'Triangulated Matrix=')
35 //back substitution

```



```

36 x(4)=B(4,5)/B(4,4);
37 printf('\nx(4) = %.0f\n',x(4))
38 for i=3:-1:1
39     k=0
40     for j=i+1:4
41         k=k+B(i,j)*x(j)
42     end
43     x(i)=(1/B(i,i))*(B(i,5)-k)
44     printf('\nx(%i) = %.0f\n',i,x(i))
45 end

```

Scilab code Exa 3.4 Gaussian Elimination Method

```

1 //Example 3.4
2 //Gaussian Elimination Method
3 //Page no. 55
4 clc;clear;close;
5
6 A=[2,1,1,10;3,2,3,18;1,4,9,16]; //
   augmented matrix
7 disp(A,'Augmented Matrix=')
8 //triangularization
9 for i=1:3
10     for j=1:4
11         if i==1 then
12             B(i,j)=A(i,j)
13         elseif i==2
14             B(i,j)=A(i,j)-A(i,1)*A(i-1,j)/A(1,1)
15             B(i+1,j)=A(i+1,j)-A(i+1,1)*A(i-1,j)/A
               (1,1)
16         elseif i==3
17             if j==1 then
18                 A=B
19             end
20             B(i,j)=B(i,j)-A(i,2)*B(i-1,j)/B(2,2)

```

```

21         end
22     end
23 end
24
25 disp(B, 'Triangulated Matrix=')
26 //back substitution
27 x(3)=B(3,4)/B(3,3);
28 printf( '\nx(3)= %i\n', x(3))
29 for i=2:-1:1
30     k=0
31     for j=i+1:3
32         k=k+B(i,j)*x(j)
33     end
34     x(i)=(1/B(i,i))*(B(i,4)-k)
35     printf( '\nx(%i)= %i\n', i, x(i))
36 end

```

Scilab code Exa 3.5 Gauss Jordan Method

```

1 //Example 3.5
2 //Gauss-Jordan Method
3 //Page no. 57
4
5 clc;clear;close;
6
7 A=[1,2,1,8;2,3,4,20;4,3,2,16]; //augmented
   matrix
8
9 for i=1:3
10     j=i
11     while (A(i,i)==0 & j<=3)
12         for k=1:4
13             B(1,k)=A(j+1,k)
14             A(j+1,k)=A(i,k)
15             A(i,k)=B(1,k)

```

```

16         end
17         disp(A)
18         j=j+1
19     end
20     for k=4:-1:i
21         A(i,k)=A(i,k)/A(i,i)
22     end
23     disp(A)
24     for k=1:3
25         if(k~=i) then
26             l=A(k,i)/A(i,i)
27             for m=i:4
28                 A(k,m)=A(k,m)-l*A(i,m)
29             end
30         end
31     end
32     end
33     disp(A)
34 end
35
36 for i=1:3
37     printf('\nx(%i) = %g\n',i,A(i,4))
38 end

```

Scilab code Exa 3.6 Gauss Jordan Method

```

1 //Example 3.6
2 //Gauss-Jordan Method
3 //Page no. 57
4
5 clc;clear;close;
6
7 A=[10,1,1,12;1,10,1,12;1,1,10,12]; //augmented
   matrix
8

```

```

9  for i=1:3
10     j=i
11     while (A(i,i)==0 & j<=3)
12         for k=1:4
13             B(1,k)=A(j+1,k)
14             A(j+1,k)=A(i,k)
15             A(i,k)=B(1,k)
16         end
17         disp(A)
18         j=j+1
19     end
20     for k=4:-1:i
21         A(i,k)=A(i,k)/A(i,i)
22     end
23     disp(A)
24     for k=1:3
25         if(k~=i) then
26             l=A(k,i)/A(i,i)
27             for m=i:4
28                 A(k,m)=A(k,m)-l*A(i,m)
29             end
30         end
31     end
32     disp(A)
33 end
34
35
36 for i=1:3
37     printf('\nx(%i) = %g\n',i,A(i,4))
38 end

```

Scilab code Exa 3.7 Gauss Jordan Method

```

1 //Example 3.7
2 //Gauss-Jordan Method

```

```

3 //Page no. 58
4
5 clc;clear;close;
6
7 A=[1,1,1,9;2,-3,4,13;3,4,5,40]; //augmented
   matrix
8
9 for i=1:3
10     j=i
11     while (A(i,i)==0 & j<=3)
12         for k=1:4
13             B(1,k)=A(j+1,k)
14             A(j+1,k)=A(i,k)
15             A(i,k)=B(1,k)
16         end
17         disp(A)
18         j=j+1
19     end
20     for k=4:-1:i
21         A(i,k)=A(i,k)/A(i,i)
22     end
23     disp(A)
24     for k=1:3
25         if(k~=i) then
26             l=A(k,i)/A(i,i)
27             for m=i:4
28                 A(k,m)=A(k,m)-l*A(i,m)
29             end
30         end
31     end
32     disp(A)
33 end
34
35
36 for i=1:3
37     printf('\nx(%i) = %g\n',i,A(i,4))
38 end

```

Scilab code Exa 3.8 Triangularization Method

```
1 //Example 3.8
2 //Triangularization Method
3 //Page no. 60
4 clc;clear;close;
5
6 A=[1,2,3;2,5,2;3,1,5];
7 B=[14;18;20];
8 printf('A can be factorizaed as follows:\n')
9 printf('\tL\t\t * \t\tU\t\t =\t\tA')
10 U(2,1)=0;U(3,1)=0;U(3,2)=0;
11 L(1,2)=0;L(1,3)=0;L(2,3)=0;
12 for i=1:3
13     L(i,i)=1
14 end
15 for i=1:3
16     U(1,i)=A(1,i)
17 end
18 L(2,1)=A(1,2)/U(1,1);
19 for i=2:3
20     U(2,i)=A(2,i)-U(1,i)*L(2,1);
21 end
22 L(3,1)=A(1,3)/U(1,1);
23 L(3,2)=(A(3,2)-U(1,2)*L(3,1))/U(2,2);
24 U(3,3)=A(3,3)-U(1,3)*L(3,1)-U(2,3)*L(3,2);
25 printf('\n')
26 for i=1:3
27     for j=1:3
28         printf('% .2 f\t',L(i,j))
29     end
30
31     if(i==2)
32         printf(' *          ')
```

```

33     else
34         printf( '\t ' )
35     end
36
37     for j=1:3
38         printf( '%.2 f\t ', U(i, j))
39     end
40     if(i==2)
41         printf( ' =      ' )
42     else
43         printf( '\t ' )
44     end
45     for j=1:3
46         printf( '%.2 f\t ', A(i, j))
47     end
48     printf( '\n ' )
49 end
50 printf( '\nY=U*X ' )
51     Y=inv(L)*B
52     X=inv(U)*Y
53 printf( '\n\nX=' )
54 for i=1:3
55     printf( '\n    %i ', X(i, 1))
56 end

```

Scilab code Exa 3.9 Triangularization Method

```

1 //Example 3.9
2 //Triangularization Method
3 //Page no. 61
4 clc;clear;close;
5
6 A=[1,2,3;2,5,2;3,1,5];
7 B=[14;18;20];
8 printf( 'A =\n ' )

```

```

9
10 U(2,1)=0;U(3,1)=0;U(3,2)=0;
11 L(1,2)=0;L(1,3)=0;L(2,3)=0;
12 for i=1:3
13     L(i,i)=1
14 end
15 for i=1:3
16     U(1,i)=A(1,i)
17 end
18 L(2,1)=A(1,2)/U(1,1);
19 for i=2:3
20     U(2,i)=A(2,i)-U(1,i)*L(2,1);
21 end
22 L(3,1)=A(1,3)/U(1,1);
23 L(3,2)=(A(3,2)-U(1,2)*L(3,1))/U(2,2);
24 U(3,3)=A(3,3)-U(1,3)*L(3,1)-U(2,3)*L(3,2);
25 printf('\n')
26 for i=1:3
27     for j=1:3
28         printf('%.2f\t',L(i,j))
29     end
30
31     if(i==2)
32         printf(' * ')
33     else
34         printf('\t')
35     end
36
37     for j=1:3
38         printf('%.2f\t',U(i,j))
39     end
40     printf('\n')
41 end
42
43     Y=inv(L)*B
44     X=inv(U)*Y
45 printf('\n\nX=')
46 for i=1:3

```



```

47     printf( '\n    %i ', X(i,1))
48 end

```

Scilab code Exa 3.10 Triangularization Method

```

1 //Example 3.10
2 //Triangularization Method
3 //Page no. 62
4 clc;clear;close;
5
6 A=[2,4,-6;1,5,3;1,3,2];
7 B=[-4;10;5];
8 printf('A can be factorizaed as follows:\n')
9 printf('\tL\t\t * \t\tU\t\t =\t\tA')
10 U(2,1)=0;U(3,1)=0;U(3,2)=0;
11 L(1,2)=0;L(1,3)=0;L(2,3)=0;
12 for i=1:3
13     L(i,i)=1
14 end
15 for i=1:3
16     U(1,i)=A(1,i)
17 end
18 L(2,1)=1/U(1,1);
19 for i=2:3
20     U(2,i)=A(2,i)-U(1,i)*L(2,1);
21 end
22 L(3,1)=1/U(1,1);
23 L(3,2)=(A(3,2)-U(1,2)*L(3,1))/U(2,2);
24 U(3,3)=A(3,3)-U(1,3)*L(3,1)-U(2,3)*L(3,2);
25 printf('\n')
26 for i=1:3
27     for j=1:3
28         printf( '%.2f\t ',L(i,j))
29     end
30

```

```

31     if(i==2)
32         printf(' * ')
33     else
34         printf('\t')
35     end
36
37     for j=1:3
38         printf('%.2f\t',U(i,j))
39     end
40     if(i==2)
41         printf(' = ')
42     else
43         printf('\t')
44     end
45     for j=1:3
46         printf('%.2f\t',A(i,j))
47     end
48     printf('\n')
49 end
50 printf('\nY=U*X')
51     Y=inv(L)*B
52     X=inv(U)*Y
53 printf('\n\nX=')
54 for i=1:3
55     printf('\n %i',X(i,1))
56 end

```

Scilab code Exa 3.11 Triangularization Method

```

1 //Example 3.11
2 //Triangularization Method
3 //Page no. 63
4 clc;clear;close;
5
6 A=[1,3,8;1,4,3;1,3,4];

```

```

7 B=[4;-2;1];
8 printf('A can be factorizaed as follows:\n')
9 printf('\tL\t\t * \t\tU\t\t =\t\tA ')
10 U(2,1)=0;U(3,1)=0;U(3,2)=0;
11 L(1,2)=0;L(1,3)=0;L(2,3)=0;
12 for i=1:3
13     L(i,i)=1
14 end
15 for i=1:3
16     U(1,i)=A(1,i)
17 end
18 L(2,1)=1/U(1,1);
19 for i=2:3
20     U(2,i)=A(2,i)-U(1,i)*L(2,1);
21 end
22 L(3,1)=1/U(1,1);
23 L(3,2)=(A(3,2)-U(1,2)*L(3,1))/U(2,2);
24 U(3,3)=A(3,3)-U(1,3)*L(3,1)-U(2,3)*L(3,2);
25 printf('\n')
26 for i=1:3
27     for j=1:3
28         printf('%0.2f\t',L(i,j))
29     end
30
31     if(i==2)
32         printf(' * ')
33     else
34         printf('\t')
35     end
36
37     for j=1:3
38         printf('%0.2f\t',U(i,j))
39     end
40     if(i==2)
41         printf(' = ')
42     else
43         printf('\t')
44     end

```

```

45     for j=1:3
46         printf( '%.2f\t', A(i, j))
47     end
48     printf( '\n')
49 end
50 printf( '\nY=U*X')
51     Y=inv(L)*B
52     X=inv(U)*Y
53 printf( '\n\nX=')
54 for i=1:3
55     printf( '\n    %.2f', X(i, 1))
56 end

```

Scilab code Exa 3.12 Triangularization Method

```

1 //Example 3.12
2 //Triangularization Method
3 //Page no. 63
4 clc;clear;close;
5
6 A=[4, -1, 2; -1, 5, 3; 2, 3, 6];
7 B=[12; 10; 18];
8 printf( 'A can be factorizaed as follows:\n')
9 printf( '\tL\t\t * \t\tU\t\t =\t\tA')
10 U(2,1)=0;U(3,1)=0;U(3,2)=0;
11 L(1,2)=0;L(1,3)=0;L(2,3)=0;
12 for i=1:3
13     L(i, i)=1
14 end
15 for i=1:3
16     U(1, i)=A(1, i)
17 end
18 L(2,1)=1/U(1,1);
19 for i=2:3
20     U(2, i)=A(2, i)-U(1, i)*L(2,1);

```

```

21 end
22 L(3,1)=1/U(1,1);
23 L(3,2)=(A(3,2)-U(1,2)*L(3,1))/U(2,2);
24 U(3,3)=A(3,3)-U(1,3)*L(3,1)-U(2,3)*L(3,2);
25 printf('\n')
26 for i=1:3
27     for j=1:3
28         printf('%.2f\t',L(i,j))
29     end
30
31     if(i==2)
32         printf(' * ')
33     else
34         printf('\t')
35     end
36
37     for j=1:3
38         printf('%.2f\t',U(i,j))
39     end
40     if(i==2)
41         printf(' = ')
42     else
43         printf('\t')
44     end
45     for j=1:3
46         printf('%.2f\t',A(i,j))
47     end
48     printf('\n')
49 end
50 printf('\nY=U*X')
51     Y=inv(L)*B
52     X=inv(U)*Y
53 printf('\n\nX=')
54 for i=1:3
55     printf('\n    %.2f',X(i,1))
56 end

```

Scilab code Exa 3.13 Crout Method

```
1 //Example 3.13
2 //Crout Method
3 //Page no. 67
4 clc;clear;close;
5
6 A=[1,2,3,1;3,1,1,0;2,1,1,0]
7 for i=1:3
8     for j=1:4
9         if j==1 then
10            M(i,j)=A(i,j)
11        elseif i==1
12            M(i,j)=A(i,j)/A(1,1)
13        elseif j==2
14            M(i,j)=A(i,j)-M(1,j)*M(i,j-1)
15        elseif i==2
16            M(i,j)=(A(i,j)-M(i,1)*M(i-1,j))/M(i,2)
17        elseif j==3
18            M(i,j)=A(i,j)-(M(i,j-2)*M(i-2,j)+M(i,j-1)*M(i-1,j))
19        else
20            M(i,j)=(A(i,j)-(M(i,j-3)*M(i-2,j)+M(i,j-2)*M(i-1,j)))/M(i,j-1)
21        end
22    end
23 end
24 disp(M, 'M = ')
25 for i=1:3
26     for j=1:4
27         if j~=4 then
28             U1(i,j)=M(i,j)
29         else
30             Y(i,1)=M(i,j)
```

```

31         end
32     end
33 end
34 U=eye(3,3)
35 for i=1:3
36     for j=1:3
37         if j>i then
38             U(i,j)=U1(i,j)
39         end
40     end
41 end
42 disp(U, 'U = ')
43 disp(Y, 'Y = ')
44 X=inv(U)*Y
45 printf('\n\nHence, the solution is : \t')
46 for i=1:3
47     printf('x%i = %i\t\t',i,X(i))
48 end

```

Scilab code Exa 3.14 Crout Method

```

1 //Example 3.14
2 //Crout Method
3 //Page no. 68
4 clc;clear;close;
5
6 A=[2,1,4,12;8,-3,2,20;4,11,-1,33]
7 for i=1:3
8     for j=1:4
9         if j==1 then
10            M(i,j)=A(i,j)
11        elseif i==1
12            M(i,j)=A(i,j)/A(1,1)
13        elseif j==2
14            M(i,j)=A(i,j)-M(1,j)*M(i,j-1)

```

```

15         elseif i==2
16             M(i,j)=(A(i,j)-M(i,1)*M(i-1,j))/M(i,2)
17         elseif j==3
18             M(i,j)=A(i,j)-(M(i,j-2)*M(i-2,j)+M(i,j
                -1)*M(i-1,j))
19         else
20             M(i,j)=(A(i,j)-(M(i,j-3)*M(i-2,j)+M(i,j
                -2)*M(i-1,j)))/M(i,j-1)
21         end
22     end
23 end
24 disp(M, 'M = ')
25 for i=1:3
26     for j=1:4
27         if j~=4 then
28             U1(i,j)=M(i,j)
29         else
30             Y(i,1)=M(i,j)
31         end
32     end
33 end
34 U=eye(3,3)
35 for i=1:3
36     for j=1:3
37         if j>i then
38             U(i,j)=U1(i,j)
39         end
40     end
41 end
42 disp(U, 'U = ')
43 disp(Y, 'Y = ')
44 X=inv(U)*Y
45 printf('\n\nHence, the solution is : \t')
46 for i=1:3
47     printf('x%i = %i\t\t',i,X(i))
48 end

```

Scilab code Exa 3.15 Crout Method

```
1 //Example 3.15
2 //Crout Method
3 //Page no. 69
4 clc;clear;close;
5
6 A
   =[1,2,-12,8,27;5,4,7,-2,4;-3,7,9,5,11;6,-12,-8,3,49]
7 for i=1:4
8     for j=1:5
9         if j==1 then
10            M(i,j)=A(i,j)
11        elseif i==1
12            M(i,j)=A(i,j)/A(1,1)
13        elseif j==2
14            M(i,j)=A(i,j)-M(1,j)*M(i,j-1)
15        elseif i==2
16            M(i,j)=(A(i,j)-M(i,1)*M(i-1,j))/M(i,2)
17        elseif j==3
18            M(i,j)=A(i,j)-(M(i,j-2)*M(1,j)+M(i,j-1)*
19                M(2,j))
20        elseif i==3
21            M(i,j)=(A(i,j)-(M(i,1)*M(i-2,j)+M(i,2)*M
22                (i-1,j)))/M(i,3)
23        elseif j==4
24            M(i,j)=A(i,j)-(M(i,j-2)*M(i-2,j)+M(i,j
25                -1)*M(i-1,j)+M(i,j-3)*M(i-3,j))
26        else
27            M(i,j)=(A(i,j)-(M(i,j-2)*M(i-1,j)+M(i,j
28                -3)*M(i-2,j)+M(i,j-4)*M(i-3,j)))/M(i,
29                j-1)
30        end
```

```

26     end
27 end
28 disp(M, 'M = ')
29 for i=1:4
30     for j=1:5
31         if j~=5 then
32             U1(i,j)=M(i,j)
33         else
34             Y(i,1)=M(i,j)
35         end
36     end
37 end
38 U=eye(4,4)
39 for i=1:4
40     for j=1:4
41         if j>i then
42             U(i,j)=U1(i,j)
43         end
44     end
45 end
46 disp(U, 'U = ')
47 disp(Y, 'Y = ')
48 X=inv(U)*Y
49 printf('\n\nHence, the solution is : \t')
50 for i=1:4
51     printf('x%i = %i \t',i,X(i))
52 end

```

Scilab code Exa 3.16 Jacobi Method

```

1 //Example 3.16
2 //Jacobi Method
3 //Page no. 72
4 clc;clear;close;
5

```

```

6 x0=0;y0=0;z0=0;
7 deff('x=f1(y,z)', 'x=(y-z+10)/5')
8 deff('y=f2(x,z)', 'y=(-2*x+z+11)/8')
9 deff('z=f3(x,y)', 'z=(x-y+3)/4')
10 for i=1:13
11     x1=f1(y0,z0);
12     y1=f2(x0,z0);
13     z1=f3(x0,y0);
14     printf('\tx(%i) = %g\n\n\ty(%i) = %g\n\n\tz(%i)
           = %g\n\n\n\n', i, x1, i, y1, i, z1)
15     x0=x1;y0=y1;z0=z1;
16 end
17 printf('Thus we find that solution converges to %g,
        %g and %g', x0, y0, z0)

```

Scilab code Exa 3.17 Gauss Seidel Method

```

1 //Example 3.17
2 //Gauss Seidel Method
3 //Page no. 73
4 clc;clear;close;
5
6 x0=0;y0=0;z0=0;
7 deff('x=f1(y,z)', 'x=(y-z+10)/5')
8 deff('y=f2(x,z)', 'y=(-2*x+z+11)/8')
9 deff('z=f3(x,y)', 'z=(x-y+3)/4')
10 for i=1:8
11     x0=f1(y0,z0);
12     y0=f2(x0,z0);
13     z0=f3(x0,y0);
14     printf('\tx(%i) = %g\n\n\ty(%i) = %g\n\n\tz(%i)
           = %g\n\n\n\n', i, x0, i, y0, i, z0)
15 end
16 printf('Thus we find that solution converges to %g,
        %g and %g', x0, y0, z0)

```

Scilab code Exa 3.18 Gauss Seidel Method

```
1 //Example 3.18
2 //Gauss Seidel Method
3 //Page no. 74
4 clc;clear;close;
5
6 x0=0;y0=0;z0=0;
7 deff('x=f1(y,z)', 'x=(110-y-z)/54')
8 deff('y=f2(x,z)', 'y=(72-2*x-6*z)/15')
9 deff('z=f3(x,y)', 'z=(85+x-6*y)/27')
10 for i=1:5
11     x0=f1(y0,z0);
12     y0=f2(x0,z0);
13     z0=f3(x0,y0);
14     printf('\tx(%i) = %g\n\n\ty(%i) = %g\n\n\tz(%i)
           = %g\n\n\n\n',i,x0,i,y0,i,z0)
15 end
16 printf('Thus we find that solution converges to %.3f
       , %.3f and %.3f',x0,y0,z0)
```

Scilab code Exa 3.19 Gauss Seidel Method

```
1 //Example 3.19
2 //Gauss Seidel Method
3 //Page no. 75
4 clc;clear;close;
5
6 x0=0;y0=0;z0=0;
7 deff('x=f1(y,z)', 'x=(32-4*y+z)/28')
8 deff('y=f2(x,z)', 'y=(35-2*x-4*z)/17')
```

```

9  deff('z=f3(x,y)', 'z=(24-x-3*y)/10')
10 for i=1:6
11     x0=f1(y0,z0);
12     y0=f2(x0,z0);
13     z0=f3(x0,y0);
14     printf('\tx(%i) = %g\n\n\ty(%i) = %g\n\n\tz(%i)
           = %g\n\n\n', i, x0, i, y0, i, z0)
15 end
16 printf('Thus we find that solution converges to %.4f
         , %.4f and %.4f', x0, y0, z0)

```

Scilab code Exa 3.20 Gauss Seidel Method

```

1  //Example 3.20
2  //Gauss Seidel Method
3  //Page no. 75
4  clc;clear;close;
5
6  x0=0;y0=0;z0=0;
7  deff('x=f1(y,z)', 'x=(17-y+2*z)/20')
8  deff('y=f2(x,z)', 'y=(-18-3*x+z)/20')
9  deff('z=f3(x,y)', 'z=(25-3*x+3*y)/20')
10 for i=1:3
11     x0=f1(y0,z0);
12     y0=f2(x0,z0);
13     z0=f3(x0,y0);
14     printf('\tx(%i) = %g\n\n\ty(%i) = %g\n\n\tz(%i)
           = %g\n\n\n', i, x0, i, y0, i, z0)
15 end
16 printf('Thus we find that solution converges to %.1g
         , %.1g and %.1g', x0, y0, z0)

```

Scilab code Exa 3.21 Relaxation Method

```

1 //Example 3.21
2 //Relaxation Method
3 //Page no. 79
4 clc;clear;close;
5
6 A=[10,-2,-2,-6;-1,10,-2,-7;-1,-1,10,-8]
7 deff('y=R(i,x,y,z)', 'y=A(i,1)*x+A(i,2)*y+A(i,3)*z+A(
      i,4)')
8 printf('dx\tdy\tdz\tdR1\tdR2\tdR3\n
      _____')
9 I=eye(3,3)
10 for i=1:3
11     printf('\n')
12     for j=1:3
13         printf(' %g\t',I(i,j))
14     end
15     for j=1:3
16         printf(' %g\t',A(j,i))
17     end
18 end
19 printf('\n\n\n\n\n xi\tyi\tzi\tdR1\tdR2\tdR3\n
      _____\n')
20 I1=[0,0,0;0,0,1;0,1,0;1,0,0]
21 for i=1:4
22     for j=1:3
23         l=0;
24         for k=1:i
25             l=1+I1(k,j)
26         end
27         I(i,j)=l
28     end
29 end
30 X=eye(1,6)-eye(1,6)
31 for i=1:4
32     printf('\n')
33     for j=1:3
34         printf(' %g\t',I1(i,j))
35         X(j)=X(j)+I1(i,j)

```

```

36     end
37     for j=1:3
38         printf( '%g\t', R(j, I(i, 1), I(i, 2), I(i, 3)))
39         if i==4 then
40             X(j+3)=X(j+3)+R(j, I(i, 1), I(i, 2), I(i, 3))
41         end
42     end
43 end
44 printf( '\n
-----\n')
45 for i=1:6
46     printf( ' %g\t', X(i))
47 end
48 printf( '\n\n\nHence the solution is \n\t x = %g\n\t
y = %g\n\t z = %g', X(1), X(2), X(3))
-----

```

Scilab code Exa 3.22 Relaxation Method

```

1 //Example 3.22
2 //Relaxation Method
3 //Page no. 80
4 clc;clear;close;
5
6 A=[10, -2, 1, -12; 1, 9, -1, -10; 2, -1, 11, -20]
7 deff( 'y=R(i, x, y, z)', 'y=A(i, 1)*x+A(i, 2)*y+A(i, 3)*z+A(
i, 4)')
8 printf( 'dx\t dy\t dz\t dR1\t dR2\t dR3\n
-----')
9 I=eye(3,3)
10 for i=1:3
11     printf( '\n')
12     for j=1:3
13         printf( ' %g\t', I(i, j))
14     end
15     for j=1:3

```

```

16         printf( '%g\t', A(j,i))
17     end
18 end
19 printf( '\n\n\n\n\n xi\tyi\tzi\tR1\tR2\tR3\n
-----\n')
20 I1
    =[0,0,0;0,0,2;0,1,0;1,0,0;0,0,-0.3;0.2,0,0;0,0.2,0;0,-0.03,0;-0.0
21 for i=1:10
22     for j=1:3
23         l=0;
24         for k=1:i
25             l=l+I1(k,j)
26         end
27         I(i,j)=l
28     end
29 end
30 X=eye(1,6)-eye(1,6)
31 for i=1:10
32     printf( '\n')
33     for j=1:3
34         printf( ' %g\t', I1(i,j))
35         X(j)=X(j)+I1(i,j)
36     end
37     for j=1:3
38         printf( '%g\t', R(j,I(i,1),I(i,2),I(i,3)))
39         if i==10 then
40             X(j+3)=X(j+3)+R(j,I(i,1),I(i,2),I(i,3))
41         end
42     end
43 end
44 printf( '\n
-----\n')
45 for i=1:6
46     printf( ' %g\t', X(i))
47 end
48 printf( '\n\n\nHence the solution is \n\t x = %g\n\t
y = %g\n\t z = %g', X(1), X(2), X(3))

```

Scilab code Exa 3.23 Relaxation Method

```
1 //Example 3.23
2 //Relaxation Method
3 //Page no. 81
4 clc;clear;close;
5
6 A=[10,-2,-3,-205;-2,10,-2,-154;-2,-1,10,-120]
7 deff('y=R(i,x,y,z)', 'y=A(i,1)*x+A(i,2)*y+A(i,3)*z+A(
8   i,4)')
9 printf('dx\tdy\tdz\tdR1\tdR2\tdR3\n
10 -----')
11 I=eye(3,3)
12 for i=1:3
13     printf('\n')
14     for j=1:3
15         printf(' %g\t',I(i,j))
16     end
17     for j=1:3
18         printf(' %g\t',A(j,i))
19     end
20 end
21 printf('\n\n\n\n\n xi\tyi\tzi\tr1\tr2\tr3\n
22 -----\n')
23 I1
24 =[0,0,0;20,0,0;0,19,0;0,0,18;10,0,0;0,6,0;0,0,2;2,0,0;0,0,1;0,1,0]
25
26 for i=1:10
27     for j=1:3
28         l=0;
29         for k=1:i
30             l=l+I1(k,j)
31         end
32         I(i,j)=l
```

```

28     end
29 end
30 X=eye(1,6)-eye(1,6)
31 for i=1:10
32     printf('\n')
33     for j=1:3
34         printf(' %g\t',I1(i,j))
35         X(j)=X(j)+I1(i,j)
36     end
37     for j=1:3
38         printf(' %g\t',R(j,I(i,1),I(i,2),I(i,3)))
39         if i==10 then
40             X(j+3)=X(j+3)+R(j,I(i,1),I(i,2),I(i,3))
41         end
42     end
43 end
44 printf('\n
-----\n')
45 for i=1:6
46     printf(' %g\t',X(i))
47 end
48 printf('\n\n\nHence the solution is \n\t x = %g\n\t
y = %g\n\t z = %g',X(1),X(2),X(3))
-----

```

Chapter 4

Eigenvalues and Eigenvectors

Scilab code Exa 4.1 Power Method

```
1 //Example 4.1
2 //Power Method
3 //Page no. 89
4 clc;close;clear;
5
6 A=[1,3,-1;3,2,4;-1,4,10];
7 e=0.001;
8 q0=[0;0;1];
9 for i=1:5
10     q1=A*q0;
11     a=max(q1)
12     for j=1:3
13         q2(j)=q1(j)/a;
14     end
15     printf('\nq(%i) = %.4 f      a = %.4 f      Scaled
16         q(%i) = %.3 f\n      %.3 f
           %.3 f\n
           %i\n\n',
           i,q1(1),a,i,q2(1),q1(2),q2(2),q1(3),q2(3))
16     q1=q2;
```

```

17     q0=q1;
18 end
19 printf('Hence the largest eigenvalue is %.2f with
        the corresponding eigenvector as %.3f\n

        %.3f\n

        %i ', a, q0(1), q0(2), q0(3))

```

Scilab code Exa 4.2 Power Method

```

1 //Example 4.2
2 //Power Method
3 //Page no. 90
4 clc; close; clear;
5
6 A=[1, -3, 2; 4, 4, -1; 6, 3, 5];
7 e=0.001;
8 q0=[1; 1; 1];
9 for i=1:9
10     q1=A*q0;
11     a=max(q1)
12     for j=1:3
13         q2(j)=q1(j)/a;
14     end
15     printf('\nq(%i) = %.4f      a = %.4f      Scaled
            q(%i) = %.3f\n      %.3f
            %.3f
            %i\n\n',
            i, q1(1), a, i, q2(1), q1(2), q2(2), q1(3), q2(3))
16     q1=q2;
17     q0=q1;
18 end
19 q0=q0*30

```

```

20 printf('Hence the largest eigenvalue is %.1g with
    the corresponding eigenvector as %.1g\n

    %.1g\n

    %i', a, q0(1), q0(2), q0(3))

```

Scilab code Exa 4.3 Power Method

```

1 //Example 4.3
2 //Power Method
3 //Page no. 91
4 clc; close; clear;
5
6 A=[2, -1, 0; -1, 2, -1; 0, -1, 2];
7 e=0.001;
8 q0=[1; 1; 1];
9 for i=1:6
10     q1=A*q0;
11     a=max(q1)
12     for j=1:3
13         q2(j)=q1(j)/a;
14     end
15     printf('\nq(%i) = %.4f      a = %.4f      Scaled
        q(%i) = %.3f\n      %.3f
        %.3f\n
        %i\n\n',
        i, q1(1), a, i, q2(1), q1(2), q2(2), q1(3), q2(3))
16     q1=q2;
17     q0=q1;
18 end
19 q0=-q0/q0(2)
20 printf('Hence the largest eigenvalue is %.3f with
    the corresponding eigenvector as %.1f\n

```

`% .1g\n`

`% .1f', a, q0(1), q0(2), q0(3)`

Scilab code Exa 4.4 Power Method

```
1 //Example 4.4
2 //Power Method
3 //Page no. 93
4 clc; close; clear;
5
6 A=[3, -1, 0; -1, 2, -1; 0, -1, 3];
7 e=0.001;
8 q0=[1; 1; 1];
9 for i=1:5
10     q1=A*q0;
11     a=max(q1)
12     for j=1:3
13         q2(j)=q1(j)/a;
14     end
15     printf( '\nq(%i) = %.4f      a = %.4f      Scaled
16             q(%i) = %.3f\n      %.3f
17             %.3f\n
18             %i\n\n',
19             i, q1(1), a, i, q2(1), q1(2), q2(2), q1(3), q2(3))
16     q1=q2;
17     q0=q1;
18 end
19 q0=-q0/q0(2)
20 printf( 'Hence the largest eigenvalue is %.1g with
21         the corresponding eigenvector as %.1g\n
22
23         %.1g\n
```

`% .1 g', a, q0(1), q0(2), q0(3)`

Scilab code Exa 4.5 Jacobi Method

```
1 //Example 4.5
2 //Jacobi Method
3 //Page no. 95
4 clc; close; clear;
5
6 A=[10,7,8,7;7,5,6,5;8,6,10,9;7,5,9,10];
7 n=4;
8 for k=1:14
9     max1=0
10    for i=1:n
11        for j=1:n
12            if A(i,j)>max1 & i~=j then
13                max1=A(i,j)
14                i1=i;j1=j;
15            end
16        end
17    end
18    fi=(atan((2*A(i1,j1))/(A(i1,i1)-A(j1,j1)+10^-20)))/2
19    disp(fi,'fi = ')
20    O1=eye(n,n)
21    O1(i1,j1)=-sin(fi)
22    O1(j1,i1)=sin(fi)
23    O1(i1,i1)=cos(fi)
24    O1(j1,j1)=cos(fi)
25    disp(O1,'O1 = ')
26    A=inv(O1)*A*O1
27    disp(A,'A1 = ')
28    end
29    printf('\n\n The eigenvalues are : \n\n')
30    for i=1:n
```

```

31     printf('\t1%i = %g\t',i,A(i,i))
32 end
33 printf('\n\n')
34 l=poly(0,'lb')
35 A=A-l*eye(n,n)
36 disp(det(A),'Characteristic Equation = ')
37 printf("\n\n\n\n\nNote : Computation Errors in some
      parts in calculation performed in book")

```

Scilab code Exa 4.6 Jacobi Method

```

1 //Example 4.6
2 //Jacobi Method
3 //Page no. 97
4 clc;close;clear;
5
6 A=[1,sqrt(2),2;sqrt(2),3,sqrt(2);2,sqrt(2),1];
7 C=A;
8 V=[sqrt(2),0,1/2;sqrt(2),0,1/4;3/(4*sqrt(2)),-1/(4*
      sqrt(2)),2]
9 S=eye(3,3)
10 disp(A,"A =")
11 VI=0;
12 for i=1:3
13     for j=1:3
14         if(i~=j)
15             VI=VI+A(i,j)^2
16                 //initial off diag norm
17         end
18     end
19 VI=sqrt(VI);
20 VF=VI*10^-7; //final threshold
21 V1=VI/3;
22 o=poly(0,"o");

```



```

23 for i=1:3
24 for q=2:3
25     for p=q-1:-1:1
26         if(A(p,q)>V1)
27             a=-A(p,q);
28             b=(A(p,p)-A(q,q))/2
29             if(b~=0)
30                 w=b*abs(1/b)*(a/sqrt(a^2+b^2));
31             else
32                 w=(a/sqrt(a^2+b^2));
33             end
34             sin0=w/sqrt(2*(1+sqrt(1-w^2)));
35             cos0=sqrt(1-sin0^2)
36         end
37         B(p,p)=A(p,p)*cos0^2+A(q,q)*sin0^2-2*A(p,q)*
           sin0*cos0
38         B(q,q)=A(p,p)*sin0^2+A(q,q)*cos0^2+2*A(p
           ,q)*sin0*cos0
39         B(p,q)=(A(p,p)-A(q,q))*sin0*cos0+A(p,q)
           *(cos0^2-sin0^2)
40         S(i,i)=S(i,i)
41         S(i,p)=S(i,p)*cos0-S(i,q)*sin0
42         S(i,q)=S(i,p)*sin0+S(i,q)*cos0
43
44     end
45 end
46 end
47 disp(B,"B =")
48 disp(S,"S =")
49 printf('\n\n\nComputation error in the solution
           provided by book')

```

Scilab code Exa 4.7 Jacobi Method

1 //Example 4.7

```

2 //Jacobi Method
3 //Page no. 99
4 clc;close;clear;
5
6 A=[2,3,1;3,2,2;1,2,1];
7 n=3;
8 for k=1:10
9     max1=0
10    for i=1:n
11        for j=1:n
12            if A(i,j)>max1 & i~=j then
13                max1=A(i,j)
14                i1=i;j1=j;
15            end
16        end
17    end
18    fi=(atan((2*A(i1,j1))/(A(i1,i1)-A(j1,j1)+10^-20)))/2
19    disp(fi,'fi = ')
20    O1=eye(n,n)
21    O1(i1,j1)=-sin(fi)
22    O1(j1,i1)=sin(fi)
23    O1(i1,i1)=cos(fi)
24    O1(j1,j1)=cos(fi)
25    disp(O1,'O1 = ')
26    A=inv(O1)*A*O1
27    disp(A,'A1 = ')
28    end
29    printf('\n\n The eigenvalues are : \n\n')
30    for i=1:n
31        printf('\t1%i = %g\t',i,A(i,i))
32    end

```

Scilab code Exa 4.8 Jacobi Method

```

1 //Example 4.8

```

```

2 //Givens Method
3 //Page no. 103
4 clc;close;clear;
5
6 A=[2,3,1;3,2,2;1,2,1];
7 n=3;
8 for k=1:1
9     max1=0
10    i1=2;j1=3;
11    fi=(atan((2*A(i1,j1))/(A(i1,i1)-A(j1,j1)+10^-20)))/2
12    disp(fi,'fi = ')
13    O1=eye(n,n)
14    O1(i1,j1)=-sin(fi)
15    O1(j1,i1)=sin(fi)
16    O1(i1,i1)=cos(fi)
17    O1(j1,j1)=cos(fi)
18    disp(O1,'O1 = ')
19    A=inv(O1)*A*O1
20    disp(A,'B = ')
21    end
22    printf('\n\n')
23    l=poly(0,'lb')
24    A=A-l*eye(n,n)
25    disp(-det(A),'Characteristic Equation = ')
26    A=roots(det(A))
27    printf('\n\n The approximate roots of characteristic
           equation are: \n\n')
28    for i=1:n
29        printf('\t1%i = %g\t',i,A(i))
30    end

```

Scilab code Exa 4.9 Givens Method

```

1 //Example 4.9
2 //Givens Method

```

```

3 //Page no. 104
4 clc;close;clear;
5
6 A=[3,2,1;2,3,2;1,2,3];
7 n=3;
8 for k=1:1
9     max1=0
10 i1=2;j1=3;
11 fi=(atan((2*A(i1,j1))/(A(i1,i1)-A(j1,j1)+10^-20)))/2
12 disp(fi,'fi = ')
13 O1=eye(n,n)
14 O1(i1,j1)=-sin(fi)
15 O1(j1,i1)=sin(fi)
16 O1(i1,i1)=cos(fi)
17 O1(j1,j1)=cos(fi)
18 disp(O1,'O1 = ')
19 A=inv(O1)*A*O1
20 disp(A,'B = ')
21 end
22 printf('\n\n')
23 l=poly(0,'lb')
24 A=A-l*eye(n,n)
25 disp(-det(A),'Characteristic Equation = ')
26 A=roots(det(A))
27 printf('\n\n The eigenvalues are : \n\n')
28 for i=1:n
29     printf('\t1%i = %g\t',i,A(i))
30 end

```

Scilab code Exa 4.10 Givens Method

```

1 //Example 4.10
2 //Givens Method
3 //Page no. 105
4 clc;close;clear;

```

```

5
6 A=[8,-6,2;-6,7,-4;2,-4,3];
7 n=3;
8 for k=1:1
9     max1=0
10 i1=2;j1=3;
11 fi=(atan((2*A(i1,j1))/(A(i1,i1)-A(j1,j1)+10^-20)))/2
12 disp(fi,'fi = ')
13 O1=eye(n,n)
14 O1(i1,j1)=-sin(fi)
15 O1(j1,i1)=sin(fi)
16 O1(i1,i1)=cos(fi)
17 O1(j1,j1)=cos(fi)
18 disp(O1,'O1 = ')
19 A=inv(O1)*A*O1
20 disp(A,'B = ')
21 end
22 printf('\n\n')
23 l=poly(0,'lb')
24 A=A-l*eye(n,n)
25 disp(det(A),'Characteristic Equation = ')
26 A=roots(det(A))
27 printf('\n\n The eigenvalues are : \n\n')
28 for i=1:n
29     printf('\t1%i = %g\t',i,A(i))
30 end

```

Scilab code Exa 4.11 Givens Method

```

1 //Example 4.11
2 //Givens Method
3 //Page no. 106
4 clc;close;clear;
5
6 A=[1,2,2;2,1,2;2,2,1];

```

```

7 n=3;
8 for k=1:1
9     max1=0
10 i1=2;j1=3;
11 fi=(atan((2*A(i1,j1))/(A(i1,i1)-A(j1,j1)+10^-20)))/2
12 disp(fi,'fi = ')
13 O1=eye(n,n)
14 O1(i1,j1)=-sin(fi)
15 O1(j1,i1)=sin(fi)
16 O1(i1,i1)=cos(fi)
17 O1(j1,j1)=cos(fi)
18 disp(O1,'O1 = ')
19 A=inv(O1)*A*O1
20 disp(A,'B = ')
21 end
22 printf('\n\n')
23 l=poly(0,'lb')
24 A=A-l*eye(n,n)
25 disp(-det(A),'Characteristic Equation = ')
26 A=roots(det(A))
27 printf('\n\n The eigenvalues are : \n\n')
28 for i=1:n
29     printf('\t1%i = %g\t',i,A(i))
30 end

```

Scilab code Exa 4.12 Givens Method

```

1 //Example 4.12
2 //Givens Method
3 //Page no. 107
4 clc;close;clear;
5
6 A=[1,2,2,2;2,1,2,2;2,2,1,3;2,2,3,1];
7 n=4;
8 for k=1:3

```

```

9      max1=0
10     if k==1 then
11         i1=2;j1=3;
12     elseif k==2
13         i1=2;j1=4;
14     else
15         i1=3;j1=4;
16     end
17 fi=(atan((2*A(i1,j1))/(A(i1,i1)-A(j1,j1)+10^-20)))/2
18 disp(fi,'fi = ')
19 O1=eye(n,n)
20 O1(i1,j1)=-sin(fi)
21 O1(j1,i1)=sin(fi)
22 O1(i1,i1)=cos(fi)
23 O1(j1,j1)=cos(fi)
24 disp(O1,'O1 = ')
25 A=inv(O1)*A*O1
26 disp(A,'B = ')
27 end
28 printf('\n\n')
29 l=poly(0,'lb')
30 A=A-l*eye(n,n)
31 disp(-det(A),'Characteristic Equation = ')
32 A=roots(det(A))
33 printf('\n\n The eigenvalues are : \n\n')
34 for i=1:n
35     printf('\t1%i = %g\t',i,A(i))
36 end

```

Scilab code Exa 4.13 House Holder Transformation

```

1 //Example 4.13
2 //House Holder Transformation
3 //Page no. 113
4 clc;clear;close;

```

```

5
6 A=[3,2,1;2,3,2;1,2,3]
7 disp(A, 'A=')
8 k=0;
9 for j=2:3
10     k=k+A(j,1)^2;
11 end
12 a=A(2,1)*abs(1/A(2,1))*sqrt(k);
13 disp(a, 'alpha=')
14 U=[0;a+A(2,1);A(3,1)];
15 disp(U, 'U=')
16 U1=U'*U;
17 disp(U1, 'UT*U=')
18 U2=U*U';
19 disp(U2, 'U*UT=')
20 P=eye(3,3)-(2*U2)/U1;
21 disp(P, 'P=');
22 B=P*A*P;
23 disp(B, 'B=');

```

Scilab code Exa 4.14 House Holder Transformation

```

1 //Example 4.14
2 //House Holder Transformation
3 //Page no. 114
4 clc;clear;close;
5
6 A=[1,3,4;3,1,2;4,2,1]
7 disp(A, 'A=')
8 k=0;
9 for j=2:3
10     k=k+A(j,1)^2;
11 end
12 a=A(2,1)*abs(1/A(2,1))*sqrt(k);
13 disp(a, 'alpha=')

```



```

14 U=[0;a+A(2,1);A(3,1)];
15 disp(U,'U=')
16 U1=U'*U;
17 disp(U1,'UT*U=')
18 U2=U*U';
19 disp(U2,'U*UT=')
20 P=eye(3,3)-(2*U2)/U1;
21 disp(P,'P=');
22 B=P*A*P;
23 disp(B,'B=');

```

Scilab code Exa 4.15 Strum Sequence

```

1 //Example 4.15
2 //Strum Sequence
3 //Page no. 116
4 clc;clear;close;
5
6 A=[1,2,2;2,1,2;2,2,1]
7 disp(A,'A=')
8 k=0;
9 for j=2:3
10     k=k+A(j,1)^2;
11 end
12 a=A(2,1)*abs(1/A(2,1))*sqrt(k);
13 U=[0;a+A(2,1);A(3,1)];
14 U1=U'*U;
15 U2=U*U';
16 P=eye(3,3)-(2*U2)/U1;
17 B=P*A*P;
18 disp(B,'Reduced Matrix = ');
19 lb=poly(0,"lb")
20 f01=1; //strum
    sequence
21 f11=(B(1,1)-lb)*f01;

```

```

22 f21=(B(2,2)-1b)*f11-B(1,2)^2*f01
23 f31=(B(3,3)-1b)*f21-B(2,3)^2*f11
24 disp(f31,"f3(lambda) = ")
25 disp(roots(f31),"Therefore the eigenvalues are : ")

```

Scilab code Exa 4.16 Strum Sequence

```

1 //Example 4.16
2 //Strum Sequence
3 //Page no. 117
4 clc;clear;close;
5
6 A=[8,-6,2;-6,7,-4;2,-4,3]
7 disp(A,'A=')
8 k=0;
9 for j=2:3
10     k=k+A(j,1)^2;
11 end
12 a=A(2,1)*abs(1/A(2,1))*sqrt(k);
13 U=[0;a+A(2,1);A(3,1)];
14 U1=U'*U;
15 U2=U*U';
16 P=eye(3,3)-(2*U2)/U1;
17 B=P*A*P;
18 disp(B,'Reduced Matrix = ');
19 1b=poly(0,"1b")
20 f01=1; //strum
    sequence
21 f11=(B(1,1)-1b)*f01;
22 f21=(B(2,2)-1b)*f11-B(1,2)^2*f01
23 f31=(B(3,3)-1b)*f21-B(2,3)^2*f11
24 disp(f31,"f3(lambda) = ")
25 disp(roots(f31),"Therefore the eigenvalues are : ")

```

Scilab code Exa 4.17 Gerschgorin Circles

```
1 //Example 4.17
2 //Gerschgorin Circles
3 //Page no. 118
4 clc;clear;close;
5
6 A=[1,2,3;2,4,6;3,6,1];
7 j=2;
8 k=3;
9 printf('The Gerschgorin Circles are : \n\n A =')
10 for i=1:3
11     printf('\t|z-%i| = |%i| + |%i| = %i\n',A(i,i),A(
        i,j),A(i,k),A(i,j)+A(i,k))
12     if j~=1 then
13         j=j-1
14     end
15     if i==2 then
16         k=k-1
17     end
18 end
```

Chapter 5

Finite Differences and Interpolation

Scilab code Exa 5.1 Backward Difference Formula

```
1 //Example 5.1
2 //Backward Difference Formula
3 //Page no. 124
4 clc;close;clear;
5 printf( '\tx\t\tty\t1st Difference      2nd Difference
           3rd Difference  4th Difference\n')
6 printf( '
           ')
7 h=0.02;
8 z=[-1;0;1;2;3;4;5]
9 deff( 'y=f(x)', 'y=x^3-3*x^2+5*x-7')
10 for i=1:7
11     z(i,2)=f(z(i,1))
12 end
13 for i=3:8
14     for j=1:9-i
15         z(j,i)=z(j+1,i-1)-z(j,i-1)
16     end
```

```

17 end
18 printf('\n')
19 for i=1:7
20     for j=1:6
21         if z(i,j)==0 then
22             printf('\t%i\t',z(i,j))
23         else
24             printf('\t%i\t',z(i,j))
25         end
26     end
27     printf('\n')
28 end

```

Scilab code Exa 5.3 Factorial Notation Method

```

1 //Example 5.3
2 //Factorial Notation Method
3 //Page no. 131
4 clc;close;clear;
5
6 h=0.00000001;h1=0000000.1
7 deff('y=f(x)', 'y=x^3-2*x^2+x-1')
8 deff('y=f1(x)', 'y=x*(x-1)*(x-2)')
9 deff('y=f2(x)', 'y=x*(x-1)')
10 for i=0:2
11     A(i+1,1)=f2(i);
12     A(i+1,2)=i;
13     A(i+1,3)=1
14     B(i+1,1)=f(i)-f1(i)
15 end
16 x=poly(0, 'x')
17 C=inv(A)*B
18 disp(C(3), '+',C(2)*x, '+',C(1)*f2(x), '+',f(x))
19 printf('\n\nf(x) = ')
20 deff('y=f3(x)', 'y=C(3)+C(2)*x+C(1)*f2(x)+f(x)')

```

```

21 disp(f3(x))
22 deff('y=f4(x)', 'y=(f3(x+h)-f3(x))/h') //1st
    derivative
23 disp(f4(x), 'dx = ')
24 deff('y=f5(x)', 'y=(f4(x+h1)-f4(x))/h1') //2nd
    derivative
25 disp(f5(x), 'd2x = ')
26 deff('y=f6(x)', 'y=(f5(x+h1)-f5(x))/h1') //3rd
    derivative
27 disp(f6(x), 'd3x = ')
28 deff('y=f7(x)', 'y=(f6(x+h1)-f6(x))/h1') //4th
    derivative
29 disp(f7(x), 'd4x = ')

```

Scilab code Exa 5.5 Finite Differences

```

1 //Example 5.5
2 //Finite Differences
3 //Page no. 132
4 clc; close; clear;
5 printf('    x\t f(x)\tdf(x)\t    d2f(x)\td3f(x)\t
        d4f(x)\n')
6 printf('
    ')
7 x=[0,1;1,3;2,9;3, poly(0, "y3");4,81]
8 for i=3:6
9     for j=1:7-i
10         x(j,i)=x(j+1,i-1)-x(j,i-1)
11     end
12 end
13 disp(x)
14 disp(roots(x(1,6)), "y3 = ")

```

Scilab code Exa 5.6 Finite Differences

```
1 //Example 5.6
2 //Finite Differences
3 //Page no. 132
4 clc;close;clear;
5 printf('  x\t f(x)\t df(x)\t d2f(x)    d3f(x)
        d4f(x)\n')
6 printf('
    ')
7 x=[0,3;1,12;2,81;3,2000;4,100]
8 for i=3:6
9     for j=1:7-i
10        x(j,i)=x(j+1,i-1)-x(j,i-1)
11    end
12 end
13 disp(x)
14 disp(x(1,6),"d4 y(0) = ")
```

Scilab code Exa 5.11 Finite Differences

```
1 //Example 5.11
2 //Finite Differences
3 //Page no. 136
4 clc;close;clear;
5 printf('  x\t f(x)\tdf(x)  d2f(x)  d3f(x) d4f(x)\n')
6 printf('
    ')
7 x=[0,-5;1,1;2,9;3,25;4,55;5,105]
```

```

8 for i=3:6
9     for j=1:8-i
10        x(j,i)=x(j+1,i-1)-x(j,i-1)
11    end
12 end
13 disp(x)
14 x1=poly(0,"x")
15 fx=x(1,2)+x1*x(1,3)+(x1^2-x1)*x(1,4)/2+(x1^3-3*x1
    ^2+2*x1)*x(1,5)/6
16 disp("is the required polynomial",fx)

```

Scilab code Exa 5.16 Finite Differences

```

1 //Example 5.16
2 //Finite Differences
3 //Page no. 138
4 clc;close;clear;
5
6 printf('  x\tf(x)\tdf(x)  d2f(x)  d3f(x)  d4f(x)\n')
7 printf('
    _____\n'
    )
8 x=[0,1;1,-1;2,1;3,-1;4,1;5,0;6,0;7,0];
9 for i=3:6
10    for j=1:8-i
11        if x(j+1,i-1)~=0 then
12            x(j,i)=x(j+1,i-1)-x(j,i-1)
13        end
14    end
15 end
16 k=-9;
17 for i=1:8
18    printf('      ')
19    for j=1:6
20        if i==j+k then

```



```

21         break
22     elseif x(i,j)==0 & j~=1 & j~=2 then
23         printf('d%i y%i \t',j-1,i-1)
24     elseif x(i,j)==0 & i~=1
25         printf('y%i \t',i-1)
26     else
27         printf('%i \t',x(i,j))
28     end
29 end
30 printf('\n')
31 k=k+2
32 end
33 x1=poly(0,"x")
34 fx=x(1,2)+x1*x(1,3)+(x1^2-x1)*x(1,4)/2+(x1^3-3*x1
    ^2+2*x1)*x(1,5)/6
35 for i=1:3
36     x(1+i,6)=16;
37     printf('\nd5y%i = 16',i)
38 end
39 printf('\nElements should be constant\n\n');
40 i=1;k=2;
41 for j=5:-1:2
42     while i<4
43         x(k+1,j)=x(k,j)+x(k,j+1);
44         if j>2 then
45             printf('\nd%i y%i = %i',j-1,k,x(k+1,j))
46         else
47             printf('\ny%i = %i',k,x(k+1,j))
48         end
49         k=k+1;
50         i=i+1;
51     end
52     i=1;k=k-2;
53 end

```

Scilab code Exa 5.17 Error Propagation

```
1 //Example 5.17
2 //Error Propagation
3 //Page no. 140
4 clc;close;clear;
5 printf('  x\t      y\t\ttdy\ttd2y\t      d3y\t      d4y
        \t d5y\n')
6 printf('


---


7 x
      =[1,1;1.1,1.5191;1.2,2.0736;1.3,2.6611;1.4,3.2816;1.5,3.9375;1.6,
8 for i=3:7
9     for j=1:13-i
10        x(j,i)=x(j+1,i-1)-x(j,i-1)
11    end
12 end
13 disp(x)
14 for i=1:11
15     if abs(x(i,7))<10^-5 then
16         continue
17     else
18         break
19     end
20 end
21 printf("\n\Therefore the error is in the value
        corresponding to %g i.e. %g",x(i+5,1),x(i+5,2))


---


```

Scilab code Exa 5.18 Error Propagation

```
1 //Example 5.18
2 //Error Propagation
3 //Page no. 141
```

```

4 clc;close;clear;
5 printf('  x\t      y\t\ttdy\ttd2y\t      d3y\t      d4y
   \t      d5y\n')
6 printf('
   ')
7 x
   =[0,2;1,5;2,8;3,17;4,38;5,75;6,140;7,233;8,362;9,533;10,752]

8 for i=3:6
9     for j=1:13-i
10        x(j,i)=x(j+1,i-1)-x(j,i-1)
11    end
12 end
13 disp(x)
14 for i=1:11
15     if abs(x(i,6))<10^-5 then
16         continue
17     else
18         break
19     end
20 end
21 printf("\n\Therefore the error is in the value
   corresponding to %g i.e. %g",x(i+4,1),x(i+4,2))

```

Scilab code Exa 5.20 Newtons Forward Difference Formula

```

1 //Example 5.20
2 //Newtons Forward Difference Formula
3 //Page no. 144
4 clc;close;clear;
5 printf(' x\t sin x\t\t 1st\t\t 2nd\t\t 3rd\t\t
   t      4th\t\t 5th\n\t\t\ttdifference\tdifference\
   tdifference\tdifference\tdifference\t')
6 printf('\n

```

```

    ')
7  h=0.2;
8  z
    =[0.5,0.47943;0.7,0.64422;0.9,0.78333;1.1,0.89121;1.3,0.96356;1.5

9  deff('y=f(x,p)', 'y=z(x,2)+p*z(x,3)+p*(p+1)*z(x,4)/2+
    p*(p+1)*(p+2)*z(x,5)/6+p*(p+1)*(p+2)*(p+3)*z(x,6)
    /24')
10 deff('y=f1(x,p)', 'y=z(x,2)+p*z(x,3)+p*(p-1)*z(x,4)
    /2+p*(p-1)*(p-2)*z(x,5)/6+p*(p-1)*(p-2)*(p-3)*z(x
    ,6)/24+p*(p-1)*(p-2)*(p-3)*(p-4)*z(x,7)/120')
11 x01=0.5;x11=0.54;
12 x02=1.3;x12=1.36
13 for i=3:7
14     for j=1:8-i
15         z(j,i)=z(j+1,i-1)-z(j,i-1)
16     end
17 end
18 printf('\n')
19 for i=1:6
20     for j=1:7
21         if z(i,j)==0 then
22             printf(' \t')
23         else
24             if j==1 then
25                 printf(' %.1f\t',z(i,j))
26             else
27                 printf('%.7f\t',z(i,j))
28             end
29         end
30     end
31     printf('\n')
32 end
33 p=(x11-x01)/h;
34 disp(f1(1,p),"fp (0.54) =");
35 p=(x12-x02)/h;
36 disp(f(5,p),"fp (1.36) =");

```

Scilab code Exa 5.21 Newtons Forward Difference Formula

```
1 //Example 5.21
2 //Newton's Forward Difference Formula
3 //Page no. 145
4 clc;close;clear;
5 printf(' x\t f(x)\t\t 1st\t\t 2nd\t\t 3rd\t\t
   \n\t\t\t difference\t difference\t difference\t')
6 printf('\n
   ')
7 h=1;
8 z=[0,-4;1,-1;2,2;3,11;4,32;5,71]
9 deff('y=f1(x,p)', 'y=z(x,2)+p*z(x,3)+p*(p-1)*z(x,4)
   /2+p*(p-1)*(p-2)*z(x,5)/6')
10 x01=0;x11=6;
11 x02=2;x12=2.5
12 for i=3:7
13     for j=1:8-i
14         z(j,i)=z(j+1,i-1)-z(j,i-1)
15     end
16 end
17 printf('\n')
18 for i=1:6
19     for j=1:5
20         if z(i,j)==0 & i~=1 then
21             printf('\t')
22         else
23             if j==1 then
24                 printf(' %.1f\t',z(i,j))
25             else
26                 printf(' %.7f\t',z(i,j))
27             end
28         end
end
```

```

29     end
30     printf('\n')
31 end
32 x=poly(0, 'x')
33 l=z(1,2)+x*z(1,3)+x*(x-1)*z(1,4)/2+x*(x-1)*(x-2)*z
    (1,5)/6
34 disp(1,"The required equation is :")
35 p=(x11-x01)/h;
36 disp(f1(1,p),"fp (6) =");
37 p=(x12-x02)/h;
38 disp(f1(3,p),"fp (2.5) =");

```

Scilab code Exa 5.22 Newtons Forward Difference Formula

```

1 //Example 5.22
2 //Newton's Forward Difference Formula
3 //Page no. 147
4 clc;close;clear;
5 printf(' x\t y\t\t 1st\t\t 2nd\t\t 3rd\t\t\t\n\
    \t\t\t difference\t difference\t difference\t')
6 printf('\n
    ')
7 h=1;
8 z=[0,-3;1,3;2,11;3,27;4,57;5,107]
9 deff('y=f1(x,p)', 'y=z(x,2)+p*z(x,3)+p*(p-1)*z(x,4)
    /2+p*(p-1)*(p-2)*z(x,5)/6')
10 x01=0;x11=6;
11 x02=2;x12=2.5
12 for i=3:7
13     for j=1:8-i
14         z(j,i)=z(j+1,i-1)-z(j,i-1)
15     end
16 end
17 printf('\n')

```

```

18 for i=1:6
19     for j=1:5
20         if z(i,j)==0 & i~=1 then
21             printf(' \t')
22         else
23             if j==1 then
24                 printf(' %.1f\t',z(i,j))
25             else
26                 printf('%.7f\t',z(i,j))
27             end
28         end
29     end
30     printf('\n')
31 end
32 x=poly(0,'x')
33 l=z(1,2)+x*z(1,3)+x*(x-1)*z(1,4)/2+x*(x-1)*(x-2)*z
    (1,5)/6
34 disp(1,"The required equation is :")

```

Scilab code Exa 5.23 Newtons Forward Difference Formula

```

1 //Example 5.23
2 //Newton's Forward Difference Formula
3 //Page no. 147
4 clc;close;clear;
5 printf(' x\t y\t d1\t d2\t d3\t d4\t')
6 printf('\n
    ')
7 h=5;
8 z=[80,5026;85,5674;90,6362;95,7088;100,7854]
9 deff('y=f(x,p)', 'y=z(x,2)+p*z(x-1,3)+p*(p+1)*z(x
    -2,4)/2+p*(p+1)*(p+2)*z(x-3,5)/6+p*(p+1)*(p+2)*(p
    +3)*z(x-4,6)/24')
10 x01=100; x11=105;

```

```

11 for i=3:7
12     for j=1:7-i
13         z(j,i)=z(j+1,i-1)-z(j,i-1)
14     end
15 end
16 printf('\n')
17 for i=1:5
18     for j=1:6
19         if z(i,j)==0 then
20             printf(' \t')
21         else
22             if j==1 then
23                 printf(' %i\t',z(i,j))
24             else
25                 printf(' %i\t',z(i,j))
26             end
27         end
28     end
29     printf('\n')
30 end
31 x=poly(0,'x')
32 l=z(1,2)+x*z(1,3)+x*(x-1)*z(1,4)/2+x*(x-1)*(x-2)*z
    (1,5)/6
33 disp(1,"The required equation is :")
34 p=(x11-x01)/h;
35 disp(f(5,p),"fp (105) =");

```

Scilab code Exa 5.24 Central Difference Derivatives

```

1 //Example 5.24
2 //Central Difference Derivatives
3 //Page no. 160
4 clc;close;clear;
5 printf('    x\t\t\t y\t\t\t d\t\t\t d2\t\t\t d3\t\t\t
    \t\t\t d4\n')

```



```

6  printf('
    ')
7  h=0.01;s=0.5;
8  def f('y=f1(x,p)', 'y=z(x,2)+p*z(x,3)+p*(p-1)*(z(x,4)+
    z(x-1,4))/4')
9  z
    =[0.01,98.4342;0.02,48.4392;0.03,31.7775;0.04,23.4492;0.05,18.454
10 for i=3:6
11     for j=1:7-i
12         z(j,i)=z(j+1,i-1)-z(j,i-1)
13     end
14 end
15 printf('\n')
16 for i=1:5
17     for j=1:6
18         if z(i,j)==0 then
19             printf(' \t')
20         else
21             printf('%0.7f\t',z(i,j))
22         end
23     end
24     printf('\n')
25 end
26 x00=0.03;x01=0.0341;
27 p=(x01-x00)/h
28 printf('\n\nf(0.0341) = %g',f1(3,p))

```

Scilab code Exa 5.27 Central Difference Derivatives

```

1 //Example 5.27
2 //Divided Difference Interpolation
3 //Page no. 165
4 clc;close;clear;

```

```

5
6 x=[-4,-1,0,2,5]
7 y=[1245,33,5,9,1335];
8 y1=y;
9 def f('yi=P(a,b,d,e)', 'yi=(b(d+1)-b(d))/(a(d+e)-a(d))
      ') //function for finding polynomials
10 for i=1:4
11     for j=1:5-i
12         z(j,i)=P(x,y,j,i)
13         y(j)=z(j,i)
14     end
15 end
16 z(6,1)=0;
17 printf('x\t y          f(x0,x1)          f(x0,x1,x3)          f(x0,
      x1,x2,x3)          f(x0,x1,x2,x3,x4)\n')
18 printf('
      n')
19 for j=1:5
20     printf(' %i\t%i \t%i\t\t%i\t\t%i\t\t\t\t\t%i\
      n',x(1,j),y1(1,j),z(j,1),z(j,2),z(j,3),z(
      j,4))
21 end
22 x1=poly(0,'x')
23 fx=y1(1)+(x1-x(1))*z(1,1)+(x1-x(1))*(x1-x(2))*z
      (1,2)+(x1-x(1))*(x1-x(2))*(x1-x(3))*z(1,3)+(
      x1-x(1))*(x1-x(2))*(x1-x(3))*(x1-x(4))*z(1,4)
24 disp(fx,"The Required Equation = ")

```

Scilab code Exa 5.28 Divided Difference Interpolation

```

1 //Example 5.28
2 //Divided Difference Interpolation
3 //Page no. 167
4 clc;close;clear;

```

```

5
6 x=[-1,0,3,6,7]
7 y=[3,-6,39,822,1611];
8 y1=y;
9 def f('yi=P(a,b,d,e)', 'yi=(b(d+1)-b(d))/(a(d+e)-a(d))
      ') //function for finding polynomials
10 for i=1:4
11     for j=1:5-i
12         z(j,i)=P(x,y,j,i)
13         y(j)=z(j,i)
14     end
15 end
16 z(6,1)=0;
17 printf('x\t y          f(x0,x1)          f(x0,x1,x3)          f(x0,
      x1,x2,x3)          f(x0,x1,x2,x3,x4)\n')
18 printf('
      n')
19 for j=1:5
20     printf(' %i\t%i \t%i\t\t%i\t\t%i\t\t\t\t\t%i\
      n',x(1,j),y1(1,j),z(j,1),z(j,2),z(j,3),z(
      j,4))
21 end
22 x1=poly(0,'x')
23 fx=y1(1)+(x1-x(1))*z(1,1)+(x1-x(1))*(x1-x(2))*z
      (1,2)+(x1-x(1))*(x1-x(2))*(x1-x(3))*z(1,3)+(
      x1-x(1))*(x1-x(2))*(x1-x(3))*(x1-x(4))*z(1,4)
24 disp(fx,"The Required Equation = ")

```

Scilab code Exa 5.29 Divided Difference Interpolation

```

1 //Example 5.29
2 //Divided Difference Interpolation
3 //Page no. 167
4 clc;close;clear;

```

```

5
6 x=[4,5,7,10,11,13]
7 y=[48,100,294,900,1210,2028];
8 y1=y;
9 def ( 'yi=P(a,b,d,e)', 'yi=(b(d+1)-b(d))/(a(d+e)-a(d))
    ' ) //function for finding polynomials
10 for i=1:6
11     for j=1:6-i
12         z(j,i)=P(x,y,j,i)
13         y(j)=z(j,i)
14     end
15 end
16 z(6,1)=0;
17 printf('x\t y          f(x0,x1)          f(x0,x1,x3)          f(x0,
    x1,x2,x3)          f(x0,x1,x2,x3,x4)\n')
18 printf('
n')
19 for j=1:5
20     printf(' %i\t%i \t%i\t\t%i\t\t%i\t\t %i
    %i\n',x(1,j),y1(1,j),z(j,1),z(j,2),z(
    j,3),z(j,4),z(j,5))
21 end
22 def ('y=f(x1)', 'y=y1(1)+(x1-x(1))*z(1,1)+(x1-x
    (1))*(x1-x(2))*z(1,2)+(x1-x(1))*(x1-x(2))*(x1
    -x(3))*z(1,3)')
23 printf('\n\nf(8) = %g',f(8))
24 printf('\n\nf(15) = %i',f(15))

```

Scilab code Exa 5.30 Maximum Error in Interpolation

```

1 //Example 5.30
2 //Maximum Error in Interpolation
3 //Page no. 169
4 clc;close;clear;

```

```

5 s=1;
6 for i=0:6
7     s=s*((5*%pi)/24-i*%pi/12)
8 end
9 s=s/factorial(7)
10 printf('Maximum Error = %g',s)

```

Scilab code Exa 5.32 Divided Difference Interpolation

```

1 //Example 5.32
2 //Divided Difference Interpolation
3 //Page no. 170
4 clc;close;clear;
5
6 x=[0,1,2,4]
7 y=[1,3,9,81];
8 y1=y;
9 def('yi=P(a,b,d,e)', 'yi=(b(d+1)-b(d))/(a(d+e)-a(d))
    ') //function for finding polynomials
10 for i=1:4
11     for j=1:4-i
12         z(j,i)=P(x,y,j,i)
13         y(j)=z(j,i)
14     end
15 end
16 z(6,1)=0;
17 printf('x\t y          f(x0,x1)          f(x0,x1,x3)          f(x0,
    x1,x2,x3)\n')
18 printf('
    \n')
19     for j=1:3
20         printf(' %i\t%i \t%i\t\t%i\t\t%i\t\t\n',x(1,
            j),y1(1,j),z(j,1),z(j,2),z(j,3))
21     end

```

```

22     deff('y=f(x1)', 'y=y1(1)+(x1-x(1))*z(1,1)+(x1-x
        (1))*(x1-x(2))*z(1,2)+(x1-x(1))*(x1-x(2))*(x1
        -x(3))*z(1,3)')
23     printf('\n\nf(3) = %g', f(3))

```

Scilab code Exa 5.36 Lagranges Interpolation Method

```

1 //Example 5.36
2 //Lagrange's Interpolation Method
3 //Page no. 176
4 clc; close; clear;
5
6 x=[7,8,9,10]
7 y=[3,1,1,9]
8 x0=9.5
9 printf('\tx\t y=f(x)\n-----\n')
10 for i=1:4
11     printf('x%i\t t%i\t   %i\n', i-1, x(i), y(i))
12 end
13 p=1; p1=1; i=1;
14 for k=1:4
15     for j=1:4
16         if k~=j then
17             p=p*(x0-x(j))
18             p1=p1*(x(k)-x(j))
19         end
20     end
21 L(k)=p/p1
22 p=1; p1=1;
23 end
24 p=0;
25 for i=1:4
26     printf('\n L%i (x) = %g\n', i-1, L(i))
27     p=p+L(i)*y(i)
28 end

```

```
29 disp(p,"P(9.5) = ")
```

Scilab code Exa 5.37 Lagranges Interpolation Method

```
1 //Example 5.37
2 //Lagranges Interpolation Method
3 //Page no. 177
4 clc;close;clear;
5
6 x=[0,1,2,5]
7 y=[2,3,12,147]
8 x0=poly(0,'x')
9 printf('\tx\t y=f(x)\n-----\n')
10 for i=1:4
11     printf('x%i\t%i\t %i\n',i-1,x(i),y(i))
12 end
13 p=1;p1=1;i=1;
14 for k=1:4
15     for j=1:4
16         if k~=j then
17             p=p*(x0-x(j))
18             p1=p1*(x(k)-x(j))
19         end
20 end
21 L(k)=p/p1
22 p=1;p1=1;
23 end
24 p=0;
25 for i=1:4
26     disp(L(i),"L(x) = ")
27     p=p+L(i)*y(i)
28 end
29 disp(p,"P(x) = ")
```

Scilab code Exa 5.38 Lagranges Interpolation Method

```
1 //Example 5.38
2 //Lagranges Interpolation Method
3 //Page no. 178
4 clc;close;clear;
5
6 x=[1,2,3,4,7]
7 y=[2,4,8,16,128]
8 x0=5
9 printf( '\tx\ty=f(x)\n-----\n' )
10 for i=1:5
11     printf( 'x%i\t%i\t  %i\n',i-1,x(i),y(i))
12 end
13 p=1;p1=1;i=1;
14 for k=1:5
15     for j=1:5
16         if k~=j then
17             p=p*(x0-x(j))
18             p1=p1*(x(k)-x(j))
19         end
20     end
21 L(k)=p/p1
22 p=1;p1=1;
23 end
24 p=0;
25 for i=1:5
26     printf( '\n L%i (x) = %g\n',i-1,L(i))
27     p=p+L(i)*y(i)
28 end
29 disp(p,"P(5) = ")
```

Scilab code Exa 5.39 Hermite Interpolation Method

```

1 //Example 5.39
2 //Hermite Interpolation Method
3 //Page no. 181
4 clc;close;clear;
5
6 x=[-1,0,1]
7 y=[-10,-4,-2]
8 y1=[10,3,2]
9 x0=poly(0,'x')
10 printf('\tx\t y=f(x)\n-----\n')
11 for i=1:3
12     printf('x%i\t%i\t %i\n',i-1,x(i),y(i))
13 end
14 p=1;p1=1;i=1;
15 for k=1:3
16     for j=1:3
17         if k~=j then
18             p=p*(x0-x(j))
19             p1=p1*(x(k)-x(j))
20         end
21     end
22 L(k)=p/p1
23 p=1;p1=1;
24 end
25 p=0;
26 L1=[-3/2,0,3/2]
27 for i=1:3
28     disp(L(i),"L(x) = ")
29     p=p+(1-2*L1(i)*(x0-x(i)))*L(i)^2*y(i)+(x0-x(i))
        *((L(i))^2)*y1(i)
30 end
31 disp(p,"P(x) = ")

```

Scilab code Exa 5.40 Hermite Interpolation Method

```
1 //Example 5.40
2 //Hermite Interpolation Method
3 //Page no. 182
4 clc;close;clear;
5
6 x=[0,1,2]
7 y=[1,3,21]
8 y1=[0,6,36]
9 x0=poly(0,'x')
10 printf('\tx\t y=f(x)\n-----\n')
11 for i=1:3
12     printf('x%i\t%i\t %i\n',i-1,x(i),y(i))
13 end
14 p=1;p1=1;i=1;
15 for k=1:3
16     for j=1:3
17         if k~=j then
18             p=p*(x0-x(j))
19             p1=p1*(x(k)-x(j))
20         end
21     end
22 L(k)=p/p1
23 p=1;p1=1;
24 end
25 p=0;
26 L1=[-3/2,0,3/2]
27 for i=1:3
28     disp(L(i),"L(x) = ")
29     p=p+(1-2*L1(i)*(x0-x(i)))*L(i)^2*y(i)+(x0-x(i))
        *((L(i))^2)*y1(i)
30 end
31 disp(p,"P(x) = ")
```

Scilab code Exa 5.41 Piecewise Cubic Hermite Interpolation Method

```

1 //Example 5.41
2 //Piecewise Cubic Hermite Interpolation Method
3 //Page no. 182
4 clc;close;clear;
5
6 x=[0,1]
7 y=[1,3]
8 y1=[0,6]
9 x0=poly(0,'x')
10 printf('\tx\tty=f(x)\n-----\n')
11 for i=1:2
12     printf('x%i\t%i\t%i\n',i-1,x(i),y(i))
13 end
14 p=1;p1=1;i=1;
15 for k=1:2
16     for j=1:2
17         if k~=j then
18             p=p*(x0-x(j))
19             p1=p1*(x(k)-x(j))
20         end
21     end
22 L(k)=p/p1
23 p=1;p1=1;
24 end
25 p=0;
26 L1=[-1,1]
27 for i=1:2
28     disp(L(i),"L(x) = ")
29     p=p+(1-2*L1(i)*(x0-x(i)))*L(i)^2*y(i)+(x0-x(i))
        *((L(i))^2)*y1(i)
30 end
31 disp(p,"P2(x) = ")
32 printf('\n\n\n\n\n')
33 x=[1,2]
34 y=[3,21]
35 y1=[6,36]

```

```

36 x0=poly(0, 'x')
37 printf( '\tx\ty=f(x)\n-----\n')
38 for i=1:2
39     printf( 'x%i\t%i\t  %i\n', i-1, x(i), y(i))
40 end
41 p=1;p1=1;i=1;
42 for k=1:2
43     for j=1:2
44         if k~=j then
45             p=p*(x0-x(j))
46             p1=p1*(x(k)-x(j))
47         end
48     end
49 L(k)=p/p1
50 p=1;p1=1;
51 end
52 p=0;
53 L1=[-1,1]
54 for i=1:2
55     disp(L(i), "L(x) = ")
56     p=p+(1-2*L1(i)*(x0-x(i)))*L(i)^2*y(i)+(x0-x(i))
        *((L(i))^2)*y1(i)
57 end
58 disp(p, "P3(x) = ")

```

Scilab code Exa 5.43 Inverse Interpolation using Newtons Forward Difference Formula

```

1 //Example 5.43
2 //Inverse Interpolation using Newton's Forward
  Difference Formula
3 //Page no. 189
4 clc;close;clear;
5 printf( ' \tx\ty\td\td2\td3\n')
6 printf( '\t-----')

```

```

7 h=1;
8 z=[2,8;3,27;4,64;5,125];
9 deff('y=f1(x,s)', 'y=(z(x,3)+(s-1/2)*z(x,4)+z(x,5)
      *(3*s^2-6*s+2)/6)/h')
10 deff('y=f2(x,s)', 'y=(z(x,4)+z(x,5)*(s-1))/h^2')
11 deff('y=f3(x,s)', 'y=z(x,5)/h^3')
12 for i=3:5
13     for j=1:6-i
14         z(j,i)=z(j+1,i-1)-z(j,i-1)
15     end
16 end
17 printf('\n')
18 for i=1:4
19     for j=1:5
20         if z(i,j)==0 then
21             printf(' \t')
22         else
23             printf('\t%g',z(i,j))
24         end
25     end
26     printf('\n')
27 end
28 fp=10;
29 f0=z(1,2);x0=z(1,1);x=fp-f0;p=(z(2,1)-z(1,1))/h;y=0;
      k=1;p=1;
30 for i=1:5
31     if i>3 then
32         l=3;
33     else
34         l=i;
35     end
36     for j=1:l
37         for k=j:-1:2
38             if k==j then
39                 y=1;
40             end
41             y=y*(p-(k-1))
42         end

```

```

43         y=y*z(1,j+2)*p/factorial(j);
44         x=x-y;
45     end
46     p=(x)/z(1,3)
47     x=fp-f0;y=0;
48     printf('\n  p%i = %g\n',i,p)
49 end
50 printf('\n\n  Hence, x = x0+ph = %g ',x+p*h)

```

Scilab code Exa 5.44 Inverse Interpolation using Everett Formula

```

1 //Example 5.44
2 //Inverse Interpolation using Everett Formula
3 //Page no. 191
4 clc;close;clear;
5 printf(' \tx\td(log(x!)/dx)\t\td2\t      d4\n')
6 printf('\t

```

```

7 x
      =[0.46,-0.0015805620,-0.0000888096,-0.000000396;0.47,0.0080664890

8 h=0.001
9 for i=1:2
10     printf('\n')
11     for j=1:4
12         printf('\t%g',x(i,j))
13     end
14 end
15 p(1)=-x(1,2)/(x(2,2)-x(1,2))
16 for i=1:2
17     p(i+1)=(-x(1,2)-(p(i)^3-p(i))*x(1,3)/6-(-p(i)
            ^3+3*p(i)^2-2*p(i))*x(1,3)/6)/(x(2,2)-x(1,2))
18 end
19 for i=1:3

```

```

20     printf('\n\n p(%i) = %g',i,p(i))
21 end
22 x=x(1,1)+p(3)*h
23 printf('\n\n x = x0 + ph = %.8g',x);

```

Scilab code Exa 5.45 Inverse Lagrange Method

```

1 //Example 5.45
2 //Inverse Lagrange Method
3 //Page no. 192
4 clc;close;clear;
5
6 x=[30,34,38,42];
7 y=[-30,-13,3,18];
8 P=0;
9 y1=0;
10 for k=0:3
11     p=1
12     for j=0:3
13         if(j~=k)
14             p=p*((y1-y(j+1))/(y(k+1)-y(j+1)))
15         end
16     end
17     printf('\n L%i(f) = %g\n',k,p)
18     p=p*x(k+1)
19     P=P+p;
20 end
21 disp(P,'Inverse Lagrange interpolation x=')

```

Scilab code Exa 5.46 Newtons Divided Difference Interpolation

```

1 //Example 5.46
2 //Newton's Divided Difference Interpolation

```

```

3 //Page no. 192
4 clc;close;clear;
5
6 x=[3,3.6,3.8]
7 y=[0.13515,0.83059,0.26253];
8 deff('y=f1(x1,x2,y1,y2)', 'y=(y2-y1)/(x2-x1)');
9 deff('y=f2(x1,x2,x3,y1,y2,y3)', 'y=(f1(x2,x3,y2,y3)-
    f1(x1,x2,y1,y2))/(x3-x1)');
10 function [x]=f(x1,x2,x3,y1,y2,y3)
11     x=(x1+2*x2+x3)/4-(f1(x1,x2,y1,y2)+f1(x2,x3,y2,y3)
    )/(4*f2(x1,x2,x3,y1,y2,y3))
12 endfunction
13 disp(f1(x(1),x(2),y(1),y(2)), ' f(x1,x2) = ')
14 disp(f1(x(2),x(3),y(2),y(3)), ' f(x2,x3) = ')
15 disp(f2(x(1),x(2),x(3),y(1),y(2),y(3)), ' f(x1,x2,x3)
    = ')
16 disp(f(x(1),x(2),x(3),y(1),y(2),y(3)), ' x0 = ')

```

Scilab code Exa 5.47 Bessel Interpolation

```

1 //Example 5.47
2 //Bessel Interpolation
3 //Page no. 194
4 clc;close;clear;
5
6 deff('y=f(x)', 'y=x^3-15*x+4');
7 h=0.02;p=1;
8 for i=1:9
9     z(i,1)=0.22+(i-1)*h
10    z(i,2)=f(z(i))
11 end
12 printf('    x\t\t    f(x) \t    d\t\t    d2\t\t    d3\t\t
    \t\t    d4\n')
13 printf('

```

```

    ')
14 for i=3:6
15     for j=1:11-i
16         z(j,i)=z(j+1,i-1)-z(j,i-1)
17     end
18 end
19 printf('\n')
20 for i=1:9
21     for j=1:6
22         if z(i,j)==0 then
23             printf(' \t')
24         else
25             printf('%0.7f\t',z(i,j))
26         end
27     end
28     printf('\n')
29 end
30 for l=1:8
31     if abs(z(l+1,2))/z(l+1,2)~=abs(z(l,2))/z(l,2)
32         then
33             break;
34         else
35             l=9;
36         end
37 end
38 function [y]=f1(x,p1)
39     if x==1 then
40         y=z(1,2)
41     elseif x==2
42         y=z(1,2)+(p1*(p1-1))/factorial(2)*((z(1-1,4)
43             +z(1,4))/2)
44     elseif x==3
45         y=z(1,2)+(p1*(p1-1))/factorial(2)*((z(1-1,4)
46             +z(1,4))/2)+(p1*(p1-1)*(p1-0.5))/
47             factorial(3)*(z(1,5))
48     end
49 endfunction
50 for i=1:3

```

```

47     p=-(f1(i,p))/z(1,3)
48     printf('\n    p%i = %g\n',i,p)
49 end
50 x=z(1,1)+p*h;
51 printf(' \n\n    x = x0 + ph = %g+(%g)(%g) = %g',z(1
    ,1),p,h,x)

```

Scilab code Exa 5.48 Chebyshev Polynomial

```

1 //Example 5.48
2 //Chebyshev Polynomial
3 //Page no. 199
4 clc;close;clear;
5
6 deff('y=f(x)', 'y=4*x^3+2*x^2');
7 n=4;
8 for i=3:-1:0
9     x(i+1)=cosd(((2*i+1)*%pi)/(2*n))
10    printf('\n    x(%i) = %g\n',i,x(i+1))
11 end

```

Scilab code Exa 5.50 Spline Interpolation

```

1 //Example 5.50
2 //Spline Interpolation
3 //Page no. 204
4 clc;close;clear;
5
6 xi=[1,2,3];
7 yi=[-1,4,21];
8 x=poly(0,'x')
9 deff('y=S(x0,x1)', 'y=(x-xi(x1))*yi(x0)/(xi(x0)-xi(x1
    ))+(x-xi(x0))*yi(x1)/(xi(x1)-xi(x0))');

```

```

10 S1=S(1,2);
11 S2=S(2,3);
12 printf('\n The required Spline is : \n')
13 disp(S2,'S2 = ',S1,'S1 = ');

```

Scilab code Exa 5.51 Spline Interpolation

```

1 //Example 5.51
2 //Spline Interpolation
3 //Page no. 204
4 clc;close;clear;
5
6 xi=[1,2,3];
7 yi=[-6,-1,16];
8 h=1;n=2;
9 x=poly(0,'x')
10 m(2)=(6*(yi(3)-2*yi(2)+yi(1)))/4
11 m(1)=0;m(3)=0;
12 function [y]=S(i,x)
13     y=m(i)*(xi(i+1)-x)^3/(6*h)
14     y=y+m(i+1)*(x-xi(i))^3/(6*h)
15     y=y+(yi(i)/h-(m(i)*h)/6)*(xi(i+1)-x)
16     y=y+(yi(i+1)/h-(m(i+1)*h)/6)*(-xi(i)+x)
17 endfunction
18 for i=1:2
19     S1(i)=S(i);
20 end
21 printf('\n The required Spline is : \n')
22 disp(' ','2<x<=3',S1(2),'S2 = ',' ','1<=x<=2',S1(1),
     'S1 = ');
23 x=1.5;
24 if x>=1 & x<=2 then
25     i=1;
26 else x>2 & x<=3
27     i=2;

```

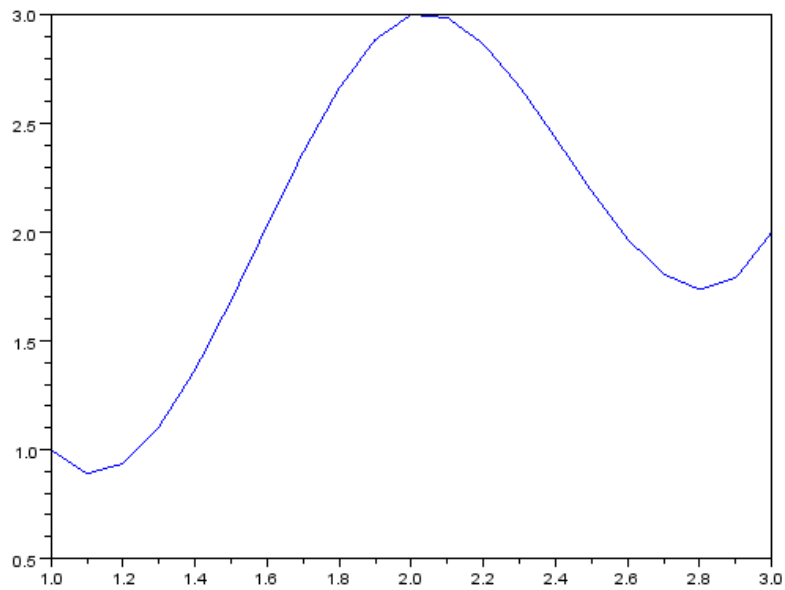


Figure 5.1: Spline Interpolation

```

28 end
29 disp(S(i,x), 'y(1.5) = ')
30 x=2;h1=0.01;
31 for i=1:2
32     Sd(i,x)=(S(i,x+h1)-S(i,x))/h1
33 end
34 disp(Sd(2,2),Sd(1,2), 'y'(2) = ')

```

Scilab code Exa 5.52 Spline Interpolation

```
1 //Example 5.52
```

```

2 //Spline Interpolation
3 //Page no. 205
4 clc;close;clear;
5 deff('y=S1(x)', 'y=18-(75*x)/2+26*x^2-11*x^3/2')
6 deff('y=S2(x)', 'y=-70+(189*x)/2-(40*x^2)+(11*x^3)/2'
    )
7 x=2;h=0.01;
8 S=[S1(x),S2(x)]
9 for i=1:2
10     printf('\n S%i (%i) = %g\n',i-1,x,S(i))
11 end
12 deff('y=S3(x)', 'y=(S1(x+h)-S1(x))/h')
13 deff('y=S4(x)', 'y=(S2(x+h)-S2(x))/h')
14 S=[S3(x),S4(x)]
15 for i=1:2
16     printf('\n S' %i (%i) = %g\n',i-1,x,S(i))
17 end
18 deff('y=S5(x)', 'y=(S3(x+h)-S3(x))/h')
19 deff('y=S6(x)', 'y=(S4(x+h)-S4(x))/h')
20 S=[S5(x),S6(x)]
21 for i=1:2
22     printf('\n S' %i (%i) = %g\n',i-1,x,S(i))
23 end
24 printf('\n\n')
25 for i=1:2
26     for j=1:3
27         if i==1 then
28             printf('\t%i',j)
29         elseif j<3
30             printf('\t%g',S1(j))
31         else
32             printf('\t%g',S2(j))
33         end
34     end
35     printf('\n')
36 end
37 x=[1:0.1:2]
38 plot(x,S1(x))

```

```
39 x=[2:0.1:3]
40 plot(x,S2(x))
```

Scilab code Exa 5.53 Spline Interpolation

```
1 //Example 5.53
2 //Spline Interpolation
3 //Page no. 206
4 clc;close;clear;
5
6 xi=[1,2,3];
7 yi=[-3,4,23];
8 h=1;n=2;
9 x=poly(0,'x')
10 m(2)=(6*(yi(3)-2*yi(2)+yi(1)))/4
11 m(1)=0;m(3)=0;
12 function [y]=S(i,x)
13     y=m(i)*(xi(i+1)-x)^3/(6*h)
14     y=y+m(i+1)*(x-xi(i))^3/(6*h)
15     y=y+(yi(i)/h-(m(i)*h)/6)*(xi(i+1)-x)
16     y=y+(yi(i+1)/h-(m(i+1)*h)/6)*(-xi(i)+x)
17 endfunction
18 for i=1:2
19     S1(i)=S(i);
20 end
21 printf('\n The required Spline is : \n')
22 disp(' ','2<x<=3',S1(2),'S2 = ',' ','1<=x<=2',S1(1),
      'S1 = ');
23 x=1.5;
24 if x>=1 & x<=2 then
25     i=1;
26 else x>2 & x<=3
27     i=2;
28 end
29 disp(S(i,x),'y(1.5) = ')
```

```

30 x=1;h1=0.01;
31 for i=1:1
32     Sd(i,x)=(S(i,x+h1)-S(i,x))/h1
33 end
34 disp(Sd(1,1),'y'(1) = ')

```

Scilab code Exa 5.54 Spline Interpolation

```

1 //Example 5.54
2 //Spline Interpolation
3 //Page no. 207
4 clc;close;clear;
5
6 xi=[0,1,2,3];
7 yi=[1,-1,-1,0];
8 h=1;n=3;
9 x=poly(0,'x')
10 m=[4,1;1,4];
11 mb=[12;6];
12 m=inv(m)*mb
13 m(3)=m(2);
14 m(2)=m(1);
15 m(1)=0;m(4)=0;
16 function [y]=S(i,x)
17     y=m(i)*(xi(i+1)-x)^3/(6*h)
18     y=y+m(i+1)*(x-xi(i))^3/(6*h)
19     y=y+(yi(i)/h-(m(i)*h)/6)*(xi(i+1)-x)
20     y=y+(yi(i+1)/h-(m(i+1)*h)/6)*(-xi(i)+x)
21 endfunction
22 for i=1:3
23     S1(i)=S(i);
24 end
25 printf('\n The required Spline is : \n')
26 disp(' ',S1(3),'S3 = ',' ',S1(2),'S2 = ',' ',S1(1),'
    S1 = ');

```


Chapter 6

Curve Fitting

Scilab code Exa 6.2 Least Line Square Approximation

```
1 //Example 6.2
2 //Least Line Square Approximation
3 //Page no. 216
4 clc;close;clear;
5
6 x=[2;5;6;9;11];
7 y=[2;4;6;9;10];n=1;
8 printf('\t 2\t\t\t 2\nn\tx\ttx\tty\ttxy\tty\n
          _____\n')
9 x1=0;x2=0;x3=0;x4=0;x5=0;x6=0;
10 for i=1:5
11     printf(' %i\t%i\t%i\t%i\t%i\n',n,x(i),x(i)
            ^2,y(i),x(i)*y(i),y(i)^2)
12     x1=x1+n;
13     x2=x2+x(i);
14     x3=x3+x(i)^2;
15     x4=x4+y(i);
16     x5=x5+x(i)*y(i);
17     x6=x6+y(i)^2;
18 end
19 printf('
```

```

20      \t%i\t%i\t%i\t%i\t%i\n',x1,x2,x3,x4,x5,x6)
21  A=[x1,x2;x2,x3]
22  B=[x4;x5]
23  C=inv(A)*B;
24  x7=poly(0,'x')
25  y=C(1)+C(2)*x7
26  disp(y,'y=')
27  x0=x2/x1;
28  y0=x4/x1;
29  A=x3-x1*x0^2;
30  B=x5-x1*x0*y0;
31  C=x6-x1*y0^2;
32  x7=poly(0,'b')
33  y=x7^2+(A-C)*x7/B-1
34  b=roots(y)
35  a=y0-b(2)*x0
36  x7=poly(0,'x')
37  disp('is the required least line',a+b(2)*x7,'y = ')

```

Scilab code Exa 6.3 Least Square Method

```

1  //Example 6.3
2  //Least Square Method
3  //Page no. 217
4  clc;close;clear;
5
6  x=[1,2,3,4,5,6,7,8];
7  y=[3,3,4,5,5,6,6,7];n=1;
8  printf('\t 2\t\t 2\n n\tx\ttx\tty\ttxy\tty\n
          -----\n')
9  x1=0;x2=0;x3=0;x4=0;x5=0;x6=0;
10 for i=1:8
11     printf(' %i\t%i\t%i\t%i\t%i\t%i\n',n,x(i),x(i)

```

```

        ^2,y(i),x(i)*y(i),y(i)^2)
12     x1=x1+n;
13     x2=x2+x(i);
14     x3=x3+x(i)^2;
15     x4=x4+y(i);
16     x5=x5+x(i)*y(i);
17     x6=x6+y(i)^2;
18 end
19 printf('
        _____\n %i
        \t%i\t%i\t%i\t%i\t%i\n',x1,x2,x3,x4,x5,x6)
20 x0=x2/x1;
21 y0=x4/x1;
22 A=x3-x1*x0^2;
23 B=x5-x1*x0*y0;
24 C=x6-x1*y0^2;
25 x7=poly(0,'b')
26 y=x7^2+(A-C)*x7/B-1
27 b=roots(y)
28 a=y0-b(2)*x0
29 x7=poly(0,'x')
30 disp('is the required least line',a+b(2)*x7,'y = ')
        _____

```

Scilab code Exa 6.4 Least Square Method

```

1 //Example 6.4
2 //Least Square Method
3 //Page no. 219
4 clc;close;clear;
5
6 t=[0.2,0.4,0.6,0.8,1]
7 h=[0.196,0.785,1.7665,3.1406,4.9075]
8 m=2;
9 for i=1:5
10     t1(i)=t(i)^(2*m)

```

```

11     h1(i)=h(i)*t(i)^2
12 end
13 g=sum(h1)/sum(t1)
14 disp(g, 'y = ')
15 g=g*2
16 disp(g, 'Gravitational Constatnt :')

```

Scilab code Exa 6.5 Power Fit Method

```

1 //Example 6.5
2 //Power Fit Method
3 //Page no. 220
4 clc;close;clear;
5
6 x=[2,2.3,2.6,2.9,3.2]
7 y=[5.1,7.5,10.6,14.4,19]
8 printf('\t 2\t 3\t 4\t 6\t\t 2\t 3\n x\ttx\ttx\ttx\ttx
\tty\ttyx\ttyx\n
n')
9 x1=0;x2=0;x3=0;x4=0;
10 for i=1:5
11     printf(' %g\t%g\t%g\t%g\t%g\t%g\t%g\t%g\n',x(i),
x(i)^2,x(i)^3,x(i)^4,x(i)^6,y(i),x(i)^2*y(i),
y(i)*x(i)^3)
12     x1=x1+x(i)^4;
13     x2=x2+x(i)^6;
14     x3=x3+x(i)^2*y(i);
15     x4=x4+y(i)*x(i)^3;
16 end
17 printf('
n \t\t\t%g\t%g\t\t%g\t%g\n',x1,x2,x3,x4)
18 a(1)=x3/x1;
19 x5=poly(0, 'x')

```

```

20 disp(a(1)*x5^2, 'The power fit , y =')
21 a(2)=x4/x2;
22 disp(a(2)*x5^3, 'The power fit , y =')
23 e=[0,0]
24 for i=1:2
25     for j=1:5
26         e(i)=e(i)+(a(i)*x(j)^(i+1)-y(j))^2
27     end
28     e(i)=sqrt(e(i)/5)
29     printf('\n\nerror%i = %.2g\n',i,e(i))
30 end
31 if e(1)>e(2) then
32     disp(a(2)*x5^3, 'y = ', 'Hence the best power fir
        curve is ')
33 else
34     disp(a(1)*x5^2, 'y = ', 'Hence the best power fir
        curve is ')
35 end

```

Scilab code Exa 6.6 Least Square Method

```

1 //Example 6.6
2 //Least Square Method
3 //Page no. 221
4 clc;close;clear;
5
6 x=[2,3,4,5];
7 y=[27.8,62.1,110,161];
8 printf('\t 2\t 4\t\t 2\nx\ttx\ttx\ty\tyx\n
        _____\n')
9 x1=0;x2=0;
10 for i=1:4
11     printf(' %g\t%g\t%g\t%g\t%g\n',x(i),x(i)^2,x(i)
        ^4,y(i),y(i)*x(i)^2)
12     x1=x1+x(i)^4;

```

```

13     x2=x2+y(i)*x(i)^2;
14 end
15 printf('-----\n \
      t\t%g\t\t%g\n',x1,x2)
16 a=x2/x1;
17 x1=poly(0,'x')
18 disp(a*x1^2,'y = ')

```

Scilab code Exa 6.7 Parabola Best Fit

```

1 //Example 6.7
2 //Parabola Best Fit
3 //Page no. 222
4 clc;close;clear;
5
6 x=[0,1,2,3,4]
7 y=[-2.1,-0.4,2.1,3.6,9.9]
8 n=1;
9 printf('\t\t 2\t 3\t 4\t\t\t 2\n n\ttx\ttx\ttx\ttx\tty\
      txy\tx y\n
      n')
10 x1=0;x2=0;x3=0;x4=0;x5=0;x6=0;x7=0;x8=0;
11 for i=1:5
12     printf(' %g\t%g\t%g\t%g\t%g\t%g\t%g\n',n,x(i)
      ),x(i)^2,x(i)^3,x(i)^4,y(i),y(i)*x(i),x(i)^2*
      y(i))
13     x1=x1+n;
14     x2=x2+x(i);
15     x3=x3+x(i)^2;
16     x4=x4+x(i)^3;
17     x5=x5+x(i)^4;
18     x6=x6+y(i);
19     x7=x7+y(i)*x(i);
20     x8=x8+x(i)^2*y(i)

```

```

21 end
22 printf('
    n %g\t%g\t%g\t%g\t%g\t%g\t%g\t%g\n', x1, x2, x3, x4,
    x5, x6, x7, x8)
23 A=[x1, x2, x3; x2, x3, x4; x3, x4, x5]
24 B=[x6; x7; x8]
25 C=inv(A)*B;
26 disp(C)
27 x=poly(0, 'x')
28 y=C(1)+C(2)*x+C(3)*x^2
29 disp(y, 'y =')

```

Scilab code Exa 6.8 Parabola Best Fit

```

1 //Example 6.8
2 //Parabola Best Fit
3 //Page no. 223
4 clc; close; clear;
5
6 x=[0.78, 1.56, 2.34, 3.12, 3.81]
7 y=[2.5, 1.2, 1.12, 2.25, 4.28]
8 n=1;
9 for i=1:5
10     x(i)=(x(i)-2.34)/0.78
11 end
12 printf('\t\t 2\t 3\t 4\t\t\t 2\n n\tX\tX\tX\tX\tty\
    tXy\tX y\n
    n')
13 x1=0; x2=0; x3=0; x4=0; x5=0; x6=0; x7=0; x8=0;
14 for i=1:5
15     printf(' %.2g\t%.2g\t%.2g\t%.2g\t%.2g\t%.2g\t%.2g\t%.2g\t%.2g\t%.2g\n', n, x(i), x(i)^2, x(i)^3, x(i)^4, y(i), y
        (i)*x(i), x(i)^2*y(i))

```

```
16     x1=x1+n;
17     x2=x2+x(i);
18     x3=x3+x(i)^2;
19     x4=x4+x(i)^3;
20     x5=x5+x(i)^4;
21     x6=x6+y(i);
22     x7=x7+y(i)*x(i);
23     x8=x8+x(i)^2*y(i)
24 end
25 printf('
      n %.2f\t%.2f\t%.2f\t%.2f\t%.2f\t%.2f\t%.2f\t%.2f\t
      n',x1,x2,x3,x4,x5,x6,x7,x8)
26 A=[x1,x2,x3;x2,x3,x4;x3,x4,x5]
27 B=[x6;x7;x8]
28 C=inv(A)*B;
29 disp(C)
30 x=poly(0,'X')
31 y=C(1)+C(2)*x+C(3)*x^2
32 disp(y,'y =')
```

Scilab code Exa 6.9 Least Square Fit

```
1 //Example 6.9
2 //Least Square Fit
3 //Page no. 224
4 clc;close;clear;
5
6 x=[-3,-1,1,3]
7 y=[15,5,1,5]
8 n=1;
9 printf('\t\t 2\t 3\t 4\t\t\t 2\n n\tx\ttx\ttx\tty\
      txy\tx y\n
      n')
```

```

10 x1=0;x2=0;x3=0;x4=0;x5=0;x6=0;x7=0;x8=0;
11 for i=1:4
12     printf(' %g\t%g\t%g\t%g\t%g\t%g\t%g\t%g\n',n,x(i)
           ),x(i)^2,x(i)^3,x(i)^4,y(i),y(i)*x(i),x(i)^2*
           y(i))
13     x1=x1+n;
14     x2=x2+x(i);
15     x3=x3+x(i)^2;
16     x4=x4+x(i)^3;
17     x5=x5+x(i)^4;
18     x6=x6+y(i);
19     x7=x7+y(i)*x(i);
20     x8=x8+x(i)^2*y(i)
21 end
22 printf('
           n %g\t%g\t%g\t%g\t%g\t%g\t%g\t%g\n',x1,x2,x3,x4,
           x5,x6,x7,x8)
23 A=[x1,x2,x3;x2,x3,x4;x3,x4,x5]
24 B=[x6;x7;x8]
25 C=inv(A)*B;
26 disp(C)
27 x=poly(0,'x')
28 y=C(1)+C(2)*x+C(3)*x^2
29 disp(y,'y =')

```

Scilab code Exa 6.10 Least Square Fit

```

1 //Example 6.10
2 //Least Square Fit
3 //Page no. 224
4 clc;close;clear;
5
6 x=[1,2,3,4]
7 y=[0.3,0.64,1.32,5.4]

```

```

8 n=1;
9 printf('\t\t 2\t 3\t 4\t\t\t 2\n n\tx\ttx\ttx\ttx\tty\t
   txy\tx y\n
   n')
10 x1=0;x2=0;x3=0;x4=0;x5=0;x6=0;x7=0;x8=0;
11 for i=1:4
12     printf(' %g\t%g\t%g\t%g\t%g\t%g\t%g\t%g\n',n,x(i)
   ),x(i)^2,x(i)^3,x(i)^4,y(i),y(i)*x(i),x(i)^2*
   y(i))
13     x1=x1+n;
14     x2=x2+x(i);
15     x3=x3+x(i)^2;
16     x4=x4+x(i)^3;
17     x5=x5+x(i)^4;
18     x6=x6+y(i);
19     x7=x7+y(i)*x(i);
20     x8=x8+x(i)^2*y(i)
21 end
22 printf('
   n %g\t%g\t%g\t%g\t%g\t%g\t%g\t%g\n',x1,x2,x3,x4,
   x5,x6,x7,x8)
23 A=[x1,x2,x3;x2,x3,x4;x3,x4,x5]
24 B=[x6;x7;x8]
25 C=inv(A)*B;
26 disp(C)
27 x=poly(0,'x')
28 y=C(1)+C(2)*x+C(3)*x^2
29 disp(y,'y =')

```

Scilab code Exa 6.11 Least Square Fit

```

1 //Example 6.9
2 //Least Square Fit

```

```

3 //Page no. 224
4 clc;close;clear;
5
6 x=[2,4,6,8,10]
7 y=[3.07,12.85,31.47,57.38,91.29]
8 n=1;
9 printf('\t\t 2\t 3\t 4\t\t\t 2\n n\ttx\ttx\ttx\ttx\ty\
txy\tx y\n
n')
10 x1=0;x2=0;x3=0;x4=0;x5=0;x6=0;x7=0;x8=0;
11 for i=1:5
12     printf(' %g\t%g\t%g\t%g\t%g\t%g\t%g\t%g\n',n,x(i)
, x(i)^2,x(i)^3,x(i)^4,y(i),y(i)*x(i),x(i)^2*
y(i))
13     x1=x1+n;
14     x2=x2+x(i);
15     x3=x3+x(i)^2;
16     x4=x4+x(i)^3;
17     x5=x5+x(i)^4;
18     x6=x6+y(i);
19     x7=x7+y(i)*x(i);
20     x8=x8+x(i)^2*y(i)
21 end
22 printf('
n %g\t%g\t%g\t%g\t%g\t%g\t%g\t%g\n',x1,x2,x3,x4,
x5,x6,x7,x8)
23 A=[x1,x2,x3;x2,x3,x4;x3,x4,x5]
24 B=[x6;x7;x8]
25 C=inv(A)*B;
26 disp(C)
27 x=poly(0,'x')
28 y=C(1)+C(2)*x+C(3)*x^2
29 disp(y,'y =')

```


27 `disp(y, 'y =')`

Chapter 7

Numerical Differentiation

Scilab code Exa 7.1 Differentiation

```
1 //Example 7.1
2 // Differentiation
3 //Page no. 230
4 clc; close; clear;
5 deff('y=f(x)', 'y=sin(x)')
6 deff('y=f1(x,h)', 'y=(f(x+h)-f(x-h))/(2*h)')
7 deff('y=f2(x,h)', 'y=(-f(x+2*h)+8*f(x+h)-8*f(x-h)+f(x
    -2*h))/(12*h)')
8 h=0.01; x=0.5
9 d=f1(x,h)
10 d1=f2(x,h)
11 printf('Centred Formula of Order O(h2) = %g\n',d)
12 printf('\n Centred Formula of Order O(h4) = %g',d1)
```

Scilab code Exa 7.2 Differentiation

```
1 //Example 7.2
2 // Differentiation
```

```

3 //Page no. 232
4 clc;close;clear;
5
6 t=[1,1.1,1.2,1.3,1.4]
7 I=[8.2277,7.2428,5.9908,4.5260,2.9122]
8 L=0.05;R=2;h=0.1;
9 deff('y=f(x)', 'y=L*i1(x)+R*I(x)')
10 deff('y=f1(x,h1)', 'y=(I(x+h1)-I(x-h1))/(2*h)')
11 deff('y=f2(x,h1)', 'y=(-I(x+2*h1)+8*I(x+h1)-8*I(x-h1)
    +I(x-2*h1))/(12*h)')
12 x=3;h1=1;
13 i1(x)=f1(x,h1)
14 E=f(x)
15 printf('Using Centred Tendency of Order O(h2)\n')
16 printf('I'(1.2) = %g\n',i1(x))
17 printf('\n E(1.2) = %g',E)
18 i1(x)=f2(x,h1)
19 E=f(x)
20 printf('\n\n\nUsing Centred Tendency of Order O(h4)\n
    n')
21 printf('I'(1.2) = %g\n',i1(x))
22 printf('\n E(1.2) = %g',E)

```

Scilab code Exa 7.3 Richardson Extrapolation

```

1 //Example 7.3
2 //Richardson Extrapolation
3 //Page no. 233
4 clc;close;clear;
5
6 t=[1,1.1,1.2,1.3,1.4]
7 I=[8.2277,7.2428,5.9908,4.5260,2.9122]
8 h=0.1;
9 deff('y=f1(x,h1)', 'y=(I(x+h1)-I(x-h1))/(2*h)')
10 deff('y=f2(x,h1)', 'y=(I(x+2*h1)-I(x-2*h1))/(4*h)')

```

```

11 deff ('y=f3(x,h1)', 'y=(I(x+h1)-I(x-h1))/(2*h)')
12 x=3;h1=1;
13 D0h=f1(x,h1)
14 printf ('\nD0(h) = %g\n',D0h)
15 D02h=f2(x,h1)
16 printf ('\nD0(2h) = %g\n',D02h)
17 I1=(4*D0h-D02h)/x
18 printf ('\nI'(1.2) = %g',I1)

```

Scilab code Exa 7.4 Differentiation

```

1 //Example 7.4
2 //Differentiation
3 //Page no. 233
4 clc;close;clear;
5
6 t=[1.2,1.3,1.4,1.5,1.6]
7 I=[1.5095,1.6984,1.9043,2.1293,2.3756]
8 h=0.1;
9 deff ('y=f2(x,h1)', 'y=(-I(x+2*h1)+8*I(x+h1)-8*I(x-h1)
    +I(x-2*h1))/(12*h)')
10 x=3;h1=1;
11 i1(x)=f2(x,h1)
12 printf ('\nUsing Centred Tendency of Order O(h4)\n')
13 printf ('f'(1.4) = %g\n',i1(x))

```

Scilab code Exa 7.5 Stirlings Central Difference Derivatives

```

1 //Example 7.5
2 //Stirlings Central Difference Derivatives
3 //Page no. 238
4 clc;close;clear;

```



```

5 printf('  x\t\t  y\t\t  d\t\t  d2\t\t  d3\t
\t  d4\n')
6 printf('
)
7 h=0.1;s=1;
8 e=[1,6,30]
9 deff('y=f1(x,s)', 'y=((z(x,3)+z(x-1,3))/2+s*z(x-1,4)
+(z(x-1,5)+z(x-2,5))*(3*s^2-1)/12)/h')
10 deff('y=f2(x,s)', 'y=(z(x-1,4))/h^2')
11 deff('y=f3(x,s)', 'y=(z(x-1,5)+z(x-2,5))/(2*h^3)')
12 z
    =[0.7,0.644218;0.8,0.717356;0.9,0.783327;1,0.841471;1.1,0.891207;

13 for i=3:6
14     for j=1:9-i
15         z(j,i)=z(j+1,i-1)-z(j,i-1)
16     end
17 end
18 printf('\n')
19 for i=1:7
20     for j=1:6
21         if z(i,j)==0 then
22             printf(' \t')
23         elseif j==1
24             printf(' %.1f\t\t',z(i,j))
25         else
26             printf('%.6f\t',z(i,j))
27         end
28     end
29     printf('\n')
30 end
31 fp=0;i=5;
32 for j=2:2:6
33     fp=fp+((-1)^(j/2+1))*(z(i,j)-z(i-2,j))/(2*h*e(j
/2))
34     i=i-1;
35 end

```

```
36 printf('\n\nf'p (sin '(x))= %g',fp)
```

Scilab code Exa 7.6 Stirlings Central Difference Derivatives

```
1 //Example 7.6
2 //Stirlings Central Difference Derivatives
3 //Page no. 239
4 clc;close;clear;
5 printf('  x\t\t  y\t\t  d\t\t  d2\t\t  d3\t\t
  \t  d4\t\t  d5\n')
6 printf('

  ')
7 h=0.2;s=1;
8 deff('y=f1()', 'y=(z(4,3)+(3*p^2-1)*z(4,4)/factorial
  (3)-(3*p^2-6*p+2)*z(3,4)/factorial(3))/h')
9 z
  =[0.2,2.10022;0.4,1.98730;0.6,1.90940;0.8,1.86672;1,1.85937;1.2,1

10 x0=0.8;p=poly(0,'p');
11 for i=3:7
12     for j=1:9-i
13         z(j,i)=z(j+1,i-1)-z(j,i-1)
14     end
15 end
16 printf('\n')
17 for i=1:7
18     for j=1:7
19         if z(i,j)==0 then
20             printf(' \t')
21         elseif j==1
22             printf(' %.1f\t\t',z(i,j))
23         else
24             printf('%.6f\t',z(i,j))
25         end
```

```

26     end
27     printf('\n')
28 end
29 f1p=f1()
30 disp(f1p)
31 r=roots(f1p);
32 for i=1:2
33     if abs(r(i))==r(i) then
34         r1=r(i)
35         disp(r(i),"p = ")
36     end
37 end
38 x=x0+r1*h;
39 disp(x,"x = ")

```

Scilab code Exa 7.7 Stirlings Central Difference Derivatives

```

1 //Example 7.7
2 //Stirlings Central Difference Derivatives
3 //Page no. 240
4 clc;close;clear;
5 printf('    x\t\t y\t\t d\t\t d2\t\t d3\t\t
\t\t d4\n')
6 printf('
)
7 h=0.2;s=1;
8 a=poly(0,'a');
9 b=poly(0,'b');
10 deff('y=f3(x)','y=z(x,1)*y2(x)+(z(x,1)-b)*z(x,2)')
11 deff('y=f4(x)','y=y1(x)*a')
12 deff('y=f1(x)','y=(z(x+1,2)-z(x-1,2)-(z(x,4)-z(x
-2,4))/factorial(3)+4*(z(x-1,6)-z(x-3,6)))/
factorial(5))/(2*h)')
13 deff('y=f2(x)','y=(z(x-1,4)-2*(z(x-2,6)))/factorial

```

```

        (4))/h^2')
14 z
    =[0.8,1.73036;1,1.95532;1.2,2.19756;1.4,2.45693;1.6,2.73309;1.8,3

15 x0=0.8;
16 for i=3:6
17     for j=1:10-i
18         z(j,i)=z(j+1,i-1)-z(j,i-1)
19     end
20 end
21 printf('\n')
22 for i=1:8
23     for j=1:6
24         if z(i,j)==0 then
25             printf(' \t')
26         elseif j==1
27             printf(' %.1f\t\t',z(i,j))
28         else
29             printf('%.6f\t',z(i,j))
30         end
31     end
32     printf('\n')
33 end
34 y1(4)=f1(4);
35 y2(4)=f2(4);
36 y1(5)=f1(5);
37 y2(5)=f2(5);
38 g=f3(4)
39 printf('\n\nny'(1.4) = %g\n\nny'(1.4) = %g\n\nny'(1.6)
    = %g\n\nny'(1.6) = %g\n\n',y1(4),y2(4),y1(5),y2
    (5))
40 disp(f3(4),f4(4))
41 printf('\n\n')
42 A=[y1(4),z(4,2);y1(5),z(5,2)];
43 B=[z(4,1)*(y2(4)+z(4,2));z(5,1)*(y2(5)+z(5,2))];
44 disp(f3(5),f4(5))
45
46 C=inv(A)*B;

```

```
47 printf('\n\n a = %g\n\n b = %g',C(1),C(2))
```

Scilab code Exa 7.8 Stirlings Central Difference Derivatives

```
1 //Example 7.8
2 //Stirlings Central Difference Derivatives
3 //Page no. 242
4 clc;close;clear;
5 printf('      x\t\t\t y\t\t\t d\t\t\t d2\t\t\t d3\t\t\t
      \t\t\t d4\n')
6 printf('
      ')
7 h=0.01;
8 a=poly(0,'n');
9 deff('y=f3(x)', 'y=z(x,1)^2*y2(x)+z(x,1)*y1(x)+(z(x
      ,1)^2-a^2)*z(x,2)')
10 deff('y=f1(x)', 'y=(z(x+1,2)-z(x-1,2)-(z(x,4)-z(x
      -2,4))/factorial(3))/(2*h)')
11 deff('y=f2(x)', 'y=(z(x-1,4)-2*(z(x-2,6))/factorial
      (4))/h^2')
12 z
      =[85,0.0353878892;85.01,0.0346198696;85.02,0.0338490002;85.03,0.0
13 for i=3:6
14     for j=1:7-i
15         z(j,i)=z(j+1,i-1)-z(j,i-1)
16     end
17 end
18 printf('\n')
19 for i=1:5
20     for j=1:6
21         if z(i,j)==0 then
22             printf(' \t')
23         elseif j==1
```

```

24         printf(' %.2f\t',z(i,j))
25     else
26         printf('%.10f\t',z(i,j))
27     end
28 end
29 printf('\n')
30 end
31 y1(3)=f1(3);
32 y2(3)=f2(3);
33
34 printf('\n\nty'(85.02) = %g\n\nty''(85.02) = %.7g\n\n',
        ,y1(3),y2(3))
35 n=f3(3)
36 disp(n,"0 =")
37 n=roots(n)
38 for i=1:2
39     if abs(n(i))==n(i) then
40         n1=n(i)
41     end
42 end
43 printf('\n\nn = %.2g',n1)

```

Scilab code Exa 7.9 Newtons Backward Formula

```

1 //Example 7.9
2 //Newtons Backward Formula
3 //Page no. 243
4 clc;close;clear;
5 printf(' x\t y\t d\t d2\t d3\t d4\t d5\t d6
        \n')
6 printf('

```

```

        ')
7 h=0.1;
8 defff('y=f2(x)', 'y=(z(x-2,4)+z(x-3,5)+z(x-4,6))/h^2')

```

```

9 z
  =[1,7.989;1.1,8.403;1.2,8.781;1.3,9.129;1.4,9.451;1.5,9.750;1.6,1

10 for i=3:8
11     for j=1:9-i
12         z(j,i)=z(j+1,i-1)-z(j,i-1)
13     end
14 end
15 printf('\n')
16 for i=1:7
17     for j=1:8
18         if z(i,j)==0 then
19             printf(' \t')
20         elseif j==1
21             printf(' %.1f\t',z(i,j))
22         else
23             printf('%.3f\t',z(i,j))
24         end
25     end
26     printf('\n')
27 end
28
29 j=6;y1=0;
30 for i=3:6
31     y1=y1+z(j,i)/(i-2)
32     j=j-1
33 end
34 y1=y1/h;
35 y2(7)=f2(7);
36 printf('\n\n dy\n --- = %.10g\n dx',y1)
37 printf('\n\n\n d2y\n ---- = %.5g\n dx2',y2(7))

```

Scilab code Exa 7.10 Lagranges Differentiation

```
1 //Example 7.10
```

```

2 //Lagrange's Differentiation
3 //Page no. 246
4 clc;close;clear;
5
6 z
   =[0,0.989992;0.1,0.999135;0.2,0.998295;0.3,0.987480];

7 h=0.1;
8 deff('y=f(x)', 'y=(-3*z(x,2)+4*z(x+1,2)-z(x+2,2))/(2*
   h)')
9 printf('\n f'(0) = %g\n\n f'(0.1) = %g',f(1),f(2))

```

Scilab code Exa 7.11 Newtons Divided Difference Interpolation

```

1 //Example 7.11
2 //Newton's Divided Difference Interpolation
3 //Page no. 247
4 clc;close;clear;
5
6 x=[-1,1,2,3]
7 y=[-21,15,12,3];
8 y1=y;h=0.0000001
9 deff('yi=P(a,b,d,e)', 'yi=(b(d+1)-b(d))/(a(d+e)-a(d))
   ') //function for finding polynomials
10 for i=1:3
11     for j=1:4-i
12         z(j,i)=P(x,y,j,i)
13         y(j)=z(j,i)
14     end
15 end
16 z(6,1)=0;
17 printf('x      y      f(x0,x1)      f(x0,x1,x3)      f
   (x0,x1,x2,x3)\n')
18 printf('

```

```

n')
19     for j=1:4
20         printf(' %i      %i \t%i\t\t%i\t\t%i\n',x(1,j)
                ,y1(1,j),z(j,1),z(j,2),z(j,3))
21     end
22     x1=poly(0,'x');
23     p=1;f=y1(1);
24     for i=1:3
25         for j=1:i
26             p=p*(x1-x(j))
27         end
28         p=p*z(1,i)
29         f=f+p
30         p=1;
31     end
32     disp(f,"f(x) = ")
33     f1=y1(1)
34     x2=poly(h,'x');
35     for i=1:3
36         for j=1:i
37             p=p*(x2-x(j))
38         end
39         p=p*z(1,i)
40         f1=f1+p
41         p=1;
42     end
43     f1=(f1-f)/h
44     disp(f1,"f'(x) = ")
45     r=roots(f1)
46     disp(r,"Roots = ")
47     x1=r(2)
48     p=1;f=y1(1);
49     for i=1:3
50         for j=1:i
51             p=p*(x1-x(j))
52         end
53         p=p*z(1,i)
54         f=f+p

```

```

55     p=1;
56 end
57 disp(f,"Maximum Value = ")

```

Scilab code Exa 7.12 Stirlings Central Difference Derivatives

```

1 //Example 7.12
2 //Stirlings Central Difference Derivatives
3 //Page no. 248
4 clc;close;clear;
5 printf('  x\t  y\t  d\t  d2\t  d3\t  d4\n
        ')
6 printf('
        _____
        ')
7 function [x]=f(x1)
8     x=0;
9     for i=3:6
10        x=x+(-1)^(i-1)*(z(x1,i))/((i-2)*h)
11    end
12 endfunction
13 h=1;
14 z=[-3,-33;-2,-12;-1,-3;0,0;1,3;2,12;3,33];
15 for i=3:6
16     for j=1:9-i
17         z(j,i)=z(j+1,i-1)-z(j,i-1)
18     end
19 end
20 printf('\n')
21 for i=1:7
22     for j=1:6
23         if j==1
24             printf('  %g\t  ',z(i,j))
25         else
26             printf('%i\t  ',z(i,j))

```

```

27         end
28     end
29     printf('\n')
30 end
31 printf("\n\nf'(-3) = %g\n\nf'(0) = %g", f(1), f(4))

```

Scilab code Exa 7.13 Newtons Backward Formula

```

1 //Example 7.13
2 //Newtons Backward Formula
3 //Page no. 248
4 clc;close;clear;
5 printf('  x\t  y\t  d\t  d2\t  d3\t  d4\t  d5\n')
6 printf('
    ')
7 h=0.5;
8 deff('y=f2(x)', 'y=(z(x,4)-z(x,5)+z(x,6))/h^2')
9 z=[1.5,3.375;2,7;2.5,13.625;3,24;3.5,38.875;4,59];
10 for i=1:6
11     for j=3:7
12         z(i,j)=-1
13     end
14 end
15 for i=3:7
16     for j=1:8-i
17         z(j,i)=z(j+1,i-1)-z(j,i-1)
18     end
19 end
20 printf('\n')
21 for i=1:6
22     for j=1:7
23         if z(i,j)==-1 then
24             printf(' \t')
25         elseif j==1

```

```

26         printf(' %.1f\t',z(i,j))
27     else
28         printf('%.3f\t',z(i,j))
29     end
30 end
31 printf('\n')
32 end
33
34 j=1;y1=0;
35 for i=3:6
36     y1=y1+(-1)^(i-1)*z(j,i)/(i-2)
37 end
38 y1=y1/h;
39 y2(7)=f2(1);
40 printf('\n\n f '(1.5)= %g',y1)
41 printf('\n\n f ''(1.5) = %g',y2(7))

```

Scilab code Exa 7.14 Newtons Divided Difference

```

1 //Example 7.14
2 //Newtons Divided Difference
3 //Page no. 249
4 clc;close;clear;
5
6 x=[3,5,11,27,34]
7 y=[-13,23,899,17315,35606]
8 deff('y=f(x1)', 'y=a1+a2*((x1-x(2))+(x1-x(3)))')
9 a1=(y(3)-y(2))/(x(3)-x(2))
10 a2=((y(4)-y(3))/(x(4)-x(3))-(y(3)-y(2))/(x(3)-x(2)))
    /(x(4)-x(2))
11 disp(y,"y:",x,"x:")
12 printf('\n\n a1 = %g\t\t a2 = %g\n\n',a1,a2)
13 disp(f(10),"f '(10) = ")

```

Chapter 8

Numerical Quadrature

Scilab code Exa 8.1 Simpsons 1 3rd Rule

```
1 //Example 8.1
2 //Simpsons 1/3rd Rule
3 //Page no 264
4 clc;clear;close;
5 a=0;b=5;n=10;h=(-a+b)/n
6
7 for i=1:n
8     if i==1 then
9         x(1,i)=a
10    else
11        x(1,i)=x(i-1)+h
12    end
13    y(1,i)=1/(4*x(i)+5)
14 end
15 disp(y,"f(x) = ",x,"x = ")
16 S=0;
17 for i=1:n
18     if(i==1 | i==n)
19         S=S+y(1,i)
20     elseif(((i)/2)-fix((i)/2)==0)
21         S=S+4*y(1,i)
```

```

22     else
23         S=S+2*y(1,i)
24     end
25 end
26 S=S*h/3;
27 printf('\n\nSimpsons 1/3rd Rule Sum = %g\n\nlog(5) =
    %.3g',S,log(5))

```

Scilab code Exa 8.2 Simpsons 1 3rd Rule and Richardson Extrapolation

```

1 //Example 8.2
2 //Simpsons 1/3rd Rule and Richardson Extrapolation
3 //Page no 264
4 clc;clear;close;
5 a=1;b=2;
6 // simpsons rule when h=0.5
7 h=0.5
8 n=(b-a)/h+1;
9 for i=1:n
10     if i==1 then
11         x(1,i)=a
12     else
13         x(1,i)=x(i-1)+h
14     end
15     y(1,i)=1/x(i)
16 end
17 disp(y,"f(x) = ",x,"x = ")
18 S=0;
19 for i=1:n
20     if(i==1 | i==n)
21         S=S+y(1,i)
22     elseif(((i)/2)-fix((i)/2)==0)
23         S=S+4*y(1,i)
24     else
25         S=S+2*y(1,i)

```

```

26     end
27 end
28 S=S*h/3;
29 printf('\n\nSimpsons 1/3rd Rule Sum when h is 0.5 =
      %g\n\n\n',S)
30
31
32 //simpsons rule when h=0.25
33 h=0.25
34 n=(b-a)/h+1;
35 for i=1:n
36     if i==1 then
37         x(1,i)=a
38     else
39         x(1,i)=x(i-1)+h
40     end
41     y(1,i)=1/x(i)
42 end
43 disp(y,"f(x) = ",x,"x = ")
44 S2=0;
45 for i=1:n
46     if(i==1 | i==n)
47         S2=S2+y(1,i)
48     elseif(((i)/2)-fix((i)/2)==0)
49         S2=S2+4*y(1,i)
50     else
51         S2=S2+2*y(1,i)
52     end
53 end
54 S2=S2*h/3;
55 printf('\n\nSimpsons 1/3rd Rule Sum when h is 0.25
      = %g\n\n\n',S2)
56
57
58 //Richardson Extrapolation
59 Q12=16*S2/15-S/15;
60 disp(Q12,"Q12 = ")
61 disp(log(2)-log(1),"Exact Value = ")

```

Scilab code Exa 8.6 Simpsons 1 3rd Rule and Bessels Quadrature

```
1 //Example 8.6
2 //Simpsons 1/3rd Rule and Bessels Quadrature
3 //Page no 271
4 clc;clear;close;
5
6 z
   =[0,0.5;0.25,0.4794;0.5,0.4594;0.75,0.4398;1,0.4207]

7 h=0.25;
8 for i=1:3
9     printf('\nWhen x = %g',z(i,1))
10    if i==1 then
11        printf(' clearly we have \n\n')
12        for j=1:5
13            y(i,j)=1
14        end
15    elseif i==2
16        printf(', using Bessels formula \n\n')
17        for j=1:5
18            if j==1 then
19                y(i,j)=1
20            else
21                y(i,j)=1+h*(z(i-1,2)*y(i-1,j)+z(i,2)
22                    *y(i,j-1))/2
23            end
24        end
25    else
26        printf(', using Simpsons formula \n\n')
27        for j=1:5
28            if j==1 then
29                y(i,j)=1+h*(z(i-2,2)+4*z(i-1,2)+z(i
30                    ,2))/3
```



```

29         else
30             y(i,j)=1+h*(z(i-2,2)*y(i-2,j)+4*z(i
                -1,2)*y(i-1,j)+z(i,2)*y(i,j-1))/3
31         end
32     end
33 end
34 for j=1:5
35     printf('y%i(%g) = %g\n\n',j,z(i,1),y(i,j))
36 end
37 end

```

Scilab code Exa 8.7 Simpsons 1 3rd Rule

```

1 //Example 8.7
2 //Simpsons 1/3rd Rule
3 //Page no 273
4 clc;clear;close;
5
6 a=100;b=200;
7 h=50;
8 n=(b-a)/h+1
9 for i=1:n
10     x(1,i)=a+(i-1)*h
11     f(1,i)=1/log(x(1,i))
12 end
13 disp(f,"f = ",x,"x = ", "If h = 50")
14 l=h*(f(1,1)+4*f(1,2)+f(1,3))/3
15 disp(l,"I = ")
16 printf('\n\n')
17 h=25;
18 n=(b-a)/h+1
19 for i=1:n
20     x(1,i)=a+(i-1)*h
21     f(1,i)=1/log(x(1,i))
22 end

```

```

23 disp(f,"f = ",x,"x = ", "If h = 25")
24 l=h*(f(1,1)+f(1,5)+4*(f(1,2)+f(1,4))+2*f(1,3))/3
25 disp(1,"I = ")
26 f1=0;
27 for i=100:200
28     l=0;
29     for j=2:i/2+1
30         if fix(i/j)~=i/j then
31             l=l+1;
32         end
33     end
34     if l==fix(i/2) then
35         f1=f1+1
36     end
37 end
38 disp(f1,"Exact no. of Prime Numbers = ")

```

Scilab code Exa 8.8 Rombers Method

```

1 //Example 8.8
2 //Rombers Method
3 //Page no 274
4 clc;clear;close;
5
6 a=0;b=1;
7 h=[0.5,0.25,0.125];
8 for j=1:3
9     n=(b-a)/h(j)+1
10 for i=1:n
11     x(1,i)=a+(i-1)*h(j)
12     y(1,i)=1/(1+x(1,i))
13 end
14 Q(j)=0;
15 for i=1:n
16     if i==1 | i==n then

```

```

17         Q(j)=Q(j)+h(j)*(y(1,i))/2
18     else
19         Q(j)=Q(j)+h(j)*(y(1,i))
20     end
21 end
22 printf('\nx : ')
23 for k=1:n
24     printf('\t %g',x(1,k))
25 end
26 printf('\nf(x) : ')
27 for k=1:n
28     printf('\t%.4f',y(1,k))
29 end
30 printf('\n\nQ(%i) = %g\n\n\n',j,Q(j))
31 end
32 R1=4*Q(2)/3-Q(1)/3
33 S=16*Q(3)/15-R1/15;
34 printf('S = %g',S)

```

Scilab code Exa 8.9 Rombers Method

```

1 //Example 8.9
2 //Rombers Method
3 //Page no 275
4 clc;clear;close;
5
6 a=4;b=5.2;
7 h=[0.4,0.2];
8 for j=1:2
9     n=(b-a)/h(j)+1
10 for i=1:n
11     x(1,i)=a+(i-1)*h(j)
12     y(1,i)=log(x(1,i))
13 end
14 Q(j)=0;

```

```

15 for i=1:n
16     if i==1 | i==n then
17         Q(j)=Q(j)+h(j)*(y(1,i))/2
18     else
19         Q(j)=Q(j)+h(j)*(y(1,i))
20     end
21 end
22 printf('\nx : ')
23 for k=1:n
24     printf('\t %g',x(1,k))
25 end
26 printf('\nf(x) : ')
27 for k=1:n
28     printf('\t%.4f',y(1,k))
29 end
30 printf('\n\nQ(%i) = %g\n\n\n',j,Q(j))
31 end
32 R1=4*Q(2)/3-Q(1)/3
33 printf('R1 = %g',R1)

```

Scilab code Exa 8.10 Rombers Method

```

1 //Example 8.10
2 //Rombers Method
3 //Page no 275
4 clc;clear;close;
5
6 a=0;b=1;
7 h=[0.5,0.25,0.125];
8 for j=1:3
9     n=(b-a)/h(j)+1
10 for i=1:n
11     x(1,i)=a+(i-1)*h(j)
12     y(1,i)=1/(1+x(1,i)^2)
13 end

```

```

14 Q(j)=0;
15 if j~=3 then
16     for i=1:n
17         if i==1 | i==n then
18             Q(j)=Q(j)+h(j)*(y(1,i))/2
19         else
20             Q(j)=Q(j)+h(j)*(y(1,i))
21         end
22     end
23 else
24     R2=0;
25     for i=1:n
26         if(i==1 | i==n)
27             R2=R2+y(1,i)
28         elseif(((i)/2)-fix((i)/2)==0)
29             R2=R2+4*y(1,i)
30         else
31             R2=R2+2*y(1,i)
32         end
33     end
34 R2=R2*h(j)/3
35 end
36 printf('\nx : ')
37 for k=1:n
38     printf('\t %g',x(1,k))
39 end
40 printf('\nf(x) : ')
41 for k=1:n
42     printf('\t%.4 f',y(1,k))
43 end
44 if j~=3 then
45     printf('\n\nQ(%i) = %g\n\n\n',j,Q(j))
46 else
47     printf('\n\nR2 = %.4 g\n\n\n',R2)
48 end
49 end
50
51 R1=4*Q(2)/3-Q(1)/3

```

```
52 S=16*R2/15-R1/15;
53 printf('\nTherefore by Rombergs Method, S = %.4g',S)
```

Scilab code Exa 8.11 Integration by Various Methods

```
1 //Example 8.11
2 //Integration by Various Methods
3 //Page no 276
4 clc;clear;close;
5 deff('y=f(x)', 'y=1/(1+x^2)')
6 a=0;b=1;
7 S=0;h=1/4;
8 n=(b-a)/h+1
9 for i=1:n
10     x(i)=(i-1)*h
11     y(i)=f(x(i))
12 end
13 c=['x', 'f(x)']
14 for i=1:2
15     printf('\n%s :\t ',c(i))
16     for j=1:n
17         if i==1 then
18             printf('%g\t',x(j))
19         else
20             printf('%.4g\t',y(j))
21         end
22     end
23 end
24
25 //trapezoidal rule
26 for i=1:n
27     if(i==1 | i==n)
28         S=S+y(i)
29     else
30         S=S+2*y(i)
```

```

31         end
32     end
33     S=S*h/2
34     printf('\n\n By Trapezoidal Method, I = %.4f ',S)
35 //Simpsons 1/3rd Rule
36 S=0;
37 for i=1:n
38     if(i==1 | i==n)
39         S=S+y(i)
40     elseif(((i)/2)-fix((i)/2)==0)
41         S=S+4*y(i)
42     else
43         S=S+2*y(i)
44     end
45 end
46 S=S*h/3;
47 printf('\n\n By Simpsons 1/3rd Rule , I = %.4g \n\n\n
',S)
48
49 S=0;h=1/6;
50 n=(b-a)/h+1
51 for i=1:n
52     x(i)=(i-1)*h
53     y(i)=f(x(i))
54 end
55 for i=1:2
56     printf('\n%s :\t ',c(i))
57     for j=1:n
58         if i==1 then
59             printf('%.4g\t ',x(j))
60         else
61             printf('%.4g\t ',y(j))
62         end
63     end
64 end
65 //Simpsons 3/8 rule
66 for i=1:n
67     if(i==1 | i==n)

```

```

68         S=S+y(i)
69         elseif i~=(n+1)/2
70             S=S+3*y(i)
71         else
72             S=S+2*y(i)
73         end
74     end
75     S=S*3*h/8
76     printf('\n\n By Simpsons 3/8 rule , I = %.5f ',S)
77
78 //Weddle's Rule
79 S=0;
80 for i=1:n
81     if i==(n+1)/2
82         S=S+6*y(i)
83     elseif ((i)/2)-fix((i)/2)~=0
84         S=S+y(i)
85     else
86         S=S+5*y(i)
87     end
88 end
89 S=S*3*h/10;
90 printf('\n\n By Weddles Rule , I = %.5f ',S)

```

Scilab code Exa 8.12 Euler Maclaurin Methods

```

1 //Example 8.12
2 //Euler Maclaurin Methods
3 //Page no 278
4 clc;clear;close;
5
6 a=0;b=1;h=[0.5,0.25]
7 h1=[6,360,15120]
8 for j=1:2
9     n=(b-a)/h(j)+1

```



```

10     for i=1:n
11         x(i)=(i-1)*h(j)
12         y(i)=sin(%pi*x(i))
13     end
14     printf('\n x = \t')
15     for i=1:n
16         printf('\t%g',x(i))
17     end
18     printf('\n f(x) = \t')
19     for i=1:n
20         printf('%0.4f\t',y(i))
21     end
22     s=0;
23     for i=0:2
24         s=s+((-1)^i)*(%pi^(2*i+1))*(h(j)^(2*(i+1)))/
                h1(i+1)
25     end
26     for i=1:n
27         if i==1 | i==n then
28             s=s+y(i)*(h(j))/2
29         else
30             s=s+2*y(i)*(h(j))/2
31         end
32     end
33     printf('\n\nI = %g\n\n\n',s)
34 end

```

Scilab code Exa 8.13 Trapezoidal and Simpsons Rule

```

1 //Example 8.13
2 //Trapezoidal and Simpsons Rule
3 //Page no. 283
4 clc;close;clear;
5
6 ax=4;bx=4.4;ay=2;by=2.4;h=0.1

```

```

7 n=(bx-ax)/h+1
8 n=5;
9 for i=1:n
10     x(i)=ax+(i-1)*h
11     y(i)=ay+(i-1)*h
12 end
13 printf(' y/x\t| ')
14 for i=1:n
15     printf('\t%g',x(i))
16 end
17 printf('\n
-----|-----
    ')
18 for i=1:n
19     printf('\n%g\t\t',y(i))
20     for j=1:n
21         z(i,j)=x(j)*y(i)
22         printf('%g\t',z(i,j))
23     end
24 end
25
26 //trapezoidal rule
27 s=0;
28 for i=1:n
29     for j=1:n
30         if (i==1 | i==n) & (j==1 | j==n) then
31             s=s+z(i,j)
32         elseif i==1 | i==n | j==1 | j==n
33             s=s+2*z(i,j)
34         else
35             s=s+4*z(i,j)
36         end
37     end
38 end
39 s=(s*(h^2))/4
40 printf('\n\n')
41 disp(s,'Trapezoidal Rule Sum = ')
42 printf('\n\n')

```

```

43 //simpsons rule
44 s=0;
45 for i=1:n
46     for j=1:n
47         if (i==1 | i==n) & (j==1 | j==n) then
48             s=s+z(i,j)
49         elseif (i/2-fix(i/2)~=0) & (j/2-fix(j/2)~=0)
50             & (i==1 | j==1 | i==n | j==n)
51             s=s+2*z(i,j)
52         elseif (i/2-fix(i/2)==0) & (j/2-fix(j/2)==0)
53             & (i==1 | j==1 | i==n | j==n)
54             s=s+4*z(i,j)
55         elseif (i/2-fix(i/2)==0) & (j/2-fix(j/2)==0)
56             & (i==ceil(n/2) | j==ceil(n/2))
57             s=s+8*z(i,j)
58         elseif (i/2-fix(i/2)==0) & (j/2-fix(j/2)==0)
59             s=s+16*z(i,j)
60         else
61             s=s+4*z(i,j)
62         end
63     end
64 end
65 s=(s*(h^2))/9
66 disp(s, 'Simpsons Rule Sum = ')

```

Scilab code Exa 8.14 Trapezoidal Rule

```

1 //Example 8.14
2 //Trapezoidal Rule
3 //Page no. 284
4 clc;close;clear;
5
6 ax=1;bx=2;ay=1;by=2;h=0.25
7 n=(bx-ax)/h+1
8 n=5;

```

```

9  for i=1:n
10     x(i)=ax+(i-1)*h
11     y(i)=ay+(i-1)*h
12 end
13 printf(' y/x\t| ')
14 for i=1:n
15     printf('\t%g\t',x(i))
16 end
17 printf('\n
    |-----|
    ')
18 for i=1:n
19     printf('\n%g\t\t',y(i))
20     for j=1:n
21         z(i,j)=1/(x(j)+y(i))
22         printf('%.5g\t\t',z(i,j))
23     end
24 end
25
26 //trapezoidal rule
27 s=0;
28 for i=1:n
29     for j=1:n
30         if (i==1 | i==n) & (j==1 | j==n) then
31             s=s+z(i,j)
32         elseif i==1 | i==n | j==1 | j==n
33             s=s+2*z(i,j)
34         else
35             s=s+4*z(i,j)
36         end
37     end
38 end
39 s=(s*(h^2))/4
40 printf('\n\n')
41 disp(s,'Trapezoidal Rule Sum = ')

```

Chapter 9

Difference Equations

Scilab code Exa 9.1 Recurrence formula

```
1 //Example 9.1
2 //Recurrence formula
3 //Page no. 288
4 clc;clear;close;
5
6 y(1)=5;
7 for i=2:7
8     y(i)=2*y(i-1)
9     printf('\ny(%i) = %g\n',i-1,y(i-1))
10 end
```

Scilab code Exa 9.2 Recurrence formula

```
1 //Example 9.3
2 //Recurrence formula
3 //Page no. 291
4 clc;clear;close;
5
```

```

6 x=poly(0, 'x')
7 f=16*x^2-8*x+1;
8 z=roots(f)
9 disp(z,f)
10 printf('\t\t n\n(c1+n*c2) (%g)', z(1))

```

Scilab code Exa 9.3 Recurrence formula

```

1 //Example 9.3
2 //Recurrence formula
3 //Page no. 291
4 clc;clear;close;
5
6 x=poly(0, 'x')
7 f=16*x^2-8*x+1;
8 z=roots(f)
9 disp(z,f)
10 printf('\t\t n\n(c1+n*c2) (%g)', z(1))

```

Scilab code Exa 9.4 Recurrence formula

```

1 //Example 9.4
2 //Recurrence formula
3 //Page no. 291
4 clc;clear;close;
5
6 x=poly(0, 'x')
7 n=3;
8 f=x^(n)-2*x^(n-1)-x^(n-2)+2;
9 z=roots(f)
10 disp(z,f)
11 printf('\n\n')
12 printf('      n      n      n\n')

```

```

13 for i=1:n
14     printf('c%i(%g)',i,z(i))
15     if i~=n then
16         printf(' + ')
17     end
18 end

```

Scilab code Exa 9.5 Difference Equation

```

1 //Example 9.5
2 //Difference Equation
3 //Page no. 291
4 clc;clear;close;
5
6 y(1)=1.5;y(2)=3;
7 n=poly(0,'n')
8 x=poly(0,'x')
9 f=x^2-2*x+1;
10 disp(f)
11 x=roots(f)
12 disp(x,"x = ")
13 A=[1,1;1,2];
14 B=[y(1);y(2)]
15 C=inv(A)*B
16 for i=1:2
17     printf('\nc(%i) = %g\n',i,C(i))
18 end
19 yn=C(1)+C(2)*n
20 disp(yn,"yn = ")

```

Scilab code Exa 9.6 Difference Equation

```

1 //Example 9.6

```

```

2 //Recurrence formula
3 //Page no. 292
4 clc;clear;close;
5
6 x=poly(0,'x')
7 f=x^2-2*x+2;
8 z=roots(f)
9 disp(z,f)
10 r=z(1)*z(2);
11 r=sqrt(r);
12 theta=atan(real(z(1)));
13 printf('\n\n\t n\n(%g)      (c1*cos(n*%g)+c2*sin(n*%g))
      ',r,theta,theta)

```

Scilab code Exa 9.8 Recurrence formula

```

1 //Example 9.8
2 //Recurrence formula
3 //Page no. 292
4 clc;clear;close;
5
6 x=poly(0,'x')
7 n=2;
8 f=x^(n)-5*x^(n-1)+6*x^(n-2);
9 z=roots(f)
10 disp(z,f)
11 printf('\n\n')
12 printf('          n          n\nf(n) = ')
13 for i=1:n
14     printf('c%i(%g)',i,z(i))
15     if i~=n then
16         printf(' + ')
17     end
18 end

```

Scilab code Exa 9.9 Particular Solution

```
1 //Example 9.9
2 //Particular Solution
3 //Page no. 293
4 clc;clear;close;
5
6 x=poly(0,'x')
7 n=2;
8 f=x^(n)-x^(n-1)-2*x^(n-2);
9 z=roots(f)
10 disp(z,f)
11 printf('\n\n')
12 printf('          n          n\nC.F. = ')
13 for i=1:n
14     printf('c%i(%g)',i,z(i))
15     if i~=n then
16         printf(' + ')
17     end
18 end
19 A=[-2,0,0;10,-2,0;9,-5,2];
20 B=[2;0;0];
21 C=inv(A)*B;
22 printf('\n\n\t 2\nP.I = (%g)n+(%g)n+(%g)',C(1),C
      (2),C(3))
```

Scilab code Exa 9.10 Particular Solution

```
1 //Example 9.10
2 //Particular Solution
3 //Page no. 294
4 clc;clear;close;
```

```

5
6 x=poly(0, 'x')
7 n=2;
8 f=x^(n)-0*x^(n-1)-4*x^(n-2);
9 z=roots(f)
10 disp(z,f)
11 printf('\n\n')
12 printf('          n          n\nC.F. = ')
13 for i=1:n
14     printf('c%i(%g)',i,z(i))
15     if i~=n then
16         printf(' + ')
17     end
18 end
19 A=[-3,0,0;4,-3,0;4,2,-3];
20 B=[9;0;0];
21 C=inv(A)*B;
22 printf('\n\n\t 2\nP.I = (%g)n+(%g)n+(%g)',C(1),C
(2),C(3))

```

Scilab code Exa 9.11 Particular Solution

```

1 //Example 9.11
2 //Particular Solution
3 //Page no. 294
4 clc;clear;close;
5
6 x=poly(0, 'x')
7 n=2;
8 f=x^(n)-0*x^(n-1)-4*x^(n-2);
9 z=roots(f)
10 disp(z,f)
11 printf('\n\n')
12 printf('          n          n\nC.F. = ')
13 for i=1:n

```

```

14     printf('c%i(%g)',i,z(i))
15     if i~=n then
16         printf(' + ')
17     end
18 end
19 A=[-3,0,0;4,-3,0;4,2,-3];
20 B=[1;1;-1];
21 C=inv(A)*B;
22 printf('\n\n\t \t 2\nP.I = (%g)n+(%g)n+(%g)',C(1),
        C(2),C(3))

```

Scilab code Exa 9.12 Particular Solution

```

1 //Example 9.12
2 //Particular Solution
3 //Page no. 295
4 clc;clear;close;
5
6 x=poly(0,'x')
7 n=2;s=['+', '-'];
8 f=x^(n)-4*x^(n-1)+5*x^(n-2);
9 z=roots(f)
10 disp(z,f)
11 printf('\n\n')
12 printf('          n          n\nC.F. = ')
13 for i=1:n
14     printf('c%i(%g %s i)',i,z(i),s(i))
15     if i~=n then
16         printf(' + ')
17     end
18 end
19 C=1;
20 printf('\n\n\t \t \nP.I = %g',C)

```

Scilab code Exa 9.13 Particular Solution

```
1 //Example 9.13
2 //Particular Solution
3 //Page no. 295
4 clc;clear;close;
5
6 x=poly(0,'x')
7 n=2;
8 f=x^(n)-7*x^(n-1)+10*x^(n-2);
9 z=roots(f)
10 disp(z,f)
11 printf('\n\n')
12 printf('          n          n\nC.F. = ')
13 for i=1:n
14     printf('c%i(%g)',i,z(i))
15     if i~=n then
16         printf(' + ')
17     end
18 end
19 C=-6;
20 printf('\n\n\t n\t \nP.I = %g(4)',C)
```

Scilab code Exa 9.14 Particular Solution

```
1 //Example 9.14
2 //Particular Solution
3 //Page no. 296
4 clc;clear;close;
5
6 x=poly(0,'x')
7 n=2;
```

```

8 f=x^(n)+5*x^(n-1)+4*x^(n-2);
9 z=roots(f)
10 disp(z,f)
11 printf('\n\n')
12 printf('          n          n\nC.F. = ')
13 for i=1:n
14     printf('c%i(%g)',i,z(i))
15     if i~=n then
16         printf(' + ')
17     end
18 end
19 C=18;
20 printf('\n\n\t      \t      \nP.I = %g(3)',C)

```

Scilab code Exa 9.15 Particular Solution

```

1 //Example 9.15
2 //Particular Solution
3 //Page no. 296
4 clc;clear;close;
5
6 x=poly(0,'x')
7 n=2;
8 f=x^(n)-1*x^(n-1)-2*x^(n-2);
9 z=roots(f)
10 disp(z,f)
11 printf('\n\n')
12 printf('          n          n\nC.F. = ')
13 for i=1:n
14     printf('c%i(%g)',i,z(i))
15     if i~=n then
16         printf(' + ')
17     end
18 end
19 A=[5/8,0;1/2,5/8];

```

```

20 B=[3;0]
21 C=inv(A)*B;
22 printf('\n\n\t \t      \t      \n P.I = (%gn%g)(4)',C(1),
        C(2))

```

Scilab code Exa 9.16 Particular Solution

```

1 //Example 9.16
2 //Particular Solution
3 //Page no. 297
4 clc;clear;close;
5
6 x=poly(0,'x')
7 n=2;
8 f=x^(n)-2*x^(n-1)+x^(n-2);
9 z=roots(f)
10 disp(z,f)
11 printf('\t\t\t\n C.F. = (c1+n*c2) (%g)',z(1))
12 A=[1,0,0;8,1,0;12,4,1];
13 B=[1;0;0];
14 C=inv(A)*B;
15 printf('\n\n\t \t      2\n P.I = (%g)n+(%g)n+(%g)',C(1),C(2)
        ,C(3))

```

Scilab code Exa 9.17 Particular Solution

```

1 //Example 9.17
2 //Particular Solution
3 //Page no. 298
4 clc;clear;close;
5
6 x=poly(0,'x')
7 n=2;

```

```

8 f=x^(n)-3*x^(n-1)+2*x^(n-2);
9 z=roots(f)
10 disp(z,f)
11 printf('\t\t\t\t\t n\nC.F. = (c1+n*c2) (%g)',z(1))
12 A=[-4,0;2,-2];
13 B=[2;0];
14 C=inv(A)*B;
15 printf('\n\n\t\t\t\t\t 2\nP.I = (%g)n+(%g)n',C(1),C(2))

```

Scilab code Exa 9.18 Particular Solution

```

1 //Example 9.16
2 //Particular Solution
3 //Page no. 297
4 clc;clear;close;
5
6 x=poly(0,'x')
7 n=1;
8 f=x^(n)-4*x^(n-1);
9 z=roots(f)
10 disp(z,f)
11 printf('\t\t\t\t\t n\nC.F. = (c1) (%g)',z(1))
12 A=[1,1;0,1];
13 B=[6;0];
14 C=inv(A)*B;
15 printf('\n\n\t\t\t\t\t \t\t\t\t\t n\nP.I = ((%g)n+(%g)n)*(4)',C
\t\t\t\t\t (1),C(2))

```

Chapter 10

Ordinary Differential Equations

Scilab code Exa 10.1 Taylor Method

```
1 //Example 10.1
2 //Taylor Method
3 //Page no. 302
4 clc;clear;close;
5
6 deff('y=f1(x,y)', 'y=y-2*x/y')
7 deff('y=f2(x,y)', 'y=(2*y*f1(x,y)-2-f1(x,y)^2)/y')
8 deff('y=f3(x,y)', 'y=(2*y*f2(x,y)-3*f1(x,y)*f2(x,y)
    +2*f1(x,y)^2)/y')
9 h=0.1;y=1;
10 x=[0.1;-0.1]
11 for i=1:2
12     k=y;
13     for j=1:3
14         if j==1 then
15             k=k+(-1)^((i-1)*j)*(h^j)*f1(0,y)/factorial(j)
16         elseif j==2
17             k=k+(-1)^((i-1)*j)*(h^j)*f2(0,y)/factorial(j)
18         elseif j==3
```



```

19         k=k+(-1)^((i-1)*j)*(h^j)*f3(0,y)/factorial(j
           )
20     end
21 end
22 printf('\ny(%g) = %g\n\n',x(i),k)
23 end

```

Scilab code Exa 10.2 Taylor Method

```

1 //Example 10.2
2 //Taylor Method
3 //Page no. 303
4 clc;clear;close;
5
6 deff('y=f1(x,y)', 'y=x-y^2')
7 deff('y=f2(x,y)', 'y=1-2*x*y+2*y^3')
8 deff('y=f3(x,y)', 'y=-2*(y-4*x*y^2+3*y^4+x^2)')
9 deff('y=f4(x,y)', 'y=-2*y*f3(x,y)-6*f1(x,y)*f2(x,y)')
10 h=0.2;y=1;
11 x=[0.2,0.4]
12 for i=1:2
13     if i==1 then
14         k=y;
15     end
16     for j=1:4
17         if j==1 then
18             k=k+(h^j)*f1((i-1)*h,y)/factorial(j)
19         elseif j==2
20             k=k+(h^j)*f2((i-1)*h,y)/factorial(j)
21         elseif j==3
22             k=k+(h^j)*f3((i-1)*h,y)/factorial(j)
23         elseif j==4
24             k=k+(h^j)*f4((i-1)*h,y)/factorial(j)
25         end
26     end

```

```

27 printf( '\ny(%g) = %g\n\n', x(i), k)
28 y=k
29 end

```

Scilab code Exa 10.3 Taylor Method

```

1 //Example 10.3
2 //Taylor Method
3 //Page no. 304
4 clc;clear;close;
5
6 deff( 'y=f1(x,y)', 'y=1' )
7 deff( 'y=f2(x,y)', 'y=x*y' )
8 deff( 'y=f3(x,y)', 'y=x*f1(x,y)+y' )
9 deff( 'y=f4(x,y)', 'y=x*f2(x,y)+2*f1(x,y)' )
10 deff( 'y=f5(x,y)', 'y=x*f3(x,y)+3*f2(x,y)' )
11 h=0.5;y=0;
12 x=[0.5,1]
13 for i=1:2
14     if i==1 then
15         k=y;
16     end
17     for j=1:5
18         if j==1 then
19             k=k+(h^j)*f1((i-1)*h,y)/factorial(j)
20         elseif j==2
21             k=k+(h^j)*f2((i-1)*h,y)/factorial(j)
22         elseif j==3
23             k=k+(h^j)*f3((i-1)*h,y)/factorial(j)
24         elseif j==4
25             k=k+(h^j)*f4((i-1)*h,y)/factorial(j)
26         elseif j==5
27             k=k+(h^j)*f5((i-1)*h,y)/factorial(j)
28         end
29     end

```

```
30 printf( '\ny(%g) = %g\n\n', x(i), k)
31 y=k
32 end
```

Scilab code Exa 10.4 Euler Method

```
1 //Example 10.4
2 //Euler Method
3 //Page no. 309
4 clc;clear;close;
5 deff( 'y=f(x,y)', 'y=(x-y)/2' )
6 y(1)=1;
7 h=0.5;
8 for i=1:7
9     printf( '\ny(%g) = %g\n', (i-1)*h, y(i) )
10    y(i+1)=y(i)+h*f((i-1)*h, y(i))
11
12 end
```

Scilab code Exa 10.5 Euler Method

```
1 //Example 10.5
2 //Euler Method
3 //Page no. 309
4 clc;clear;close;
5 deff( 'y=f(x,y)', 'y=(y-x)/(x+y)' )
6 y(1)=1;
7 h=0.02;
8 for i=1:6
9     printf( '\ny(%g) = %g\n', (i-1)*h, y(i) )
10    y(i+1)=y(i)+h*f((i-1)*h, y(i))
11
12 end
```

Scilab code Exa 10.6 Euler and Modified Euler Method

```
1 //Example 10.6
2 //Euler and Modified Euler Method
3 //Page no. 311
4 clc;clear;close;
5 deff('y=f(x,y)', 'y=y-x^2')
6 y(1)=1;
7 h=0.2;
8 for i=1:4
9     printf('\ny(%g) = %g\n', (i-1)*h, y(i))
10    y(i+1)=y(i)+h*f((i-1)*h, y(i))
11 end
12 printf('\n\n\n By Modified Euler Method\n')
13 for i=1:4
14    printf('\ny(%g) = %g\n', (i-1)*h, y(i))
15    y(i+1)=y(i)+h*f((i-1)*h+h/2, y(i)+h*f((i-1)*h, y(i)
16    ))/2)
16 end
```

Scilab code Exa 10.7 Modified Euler Method

```
1 //Example 10.7
2 //Modified Euler Method
3 //Page no. 312
4 clc;clear;close;
5 deff('y=f(x,y)', 'y=x+abs(sqrt(y))')
6 y(1)=1;
7 h=0.2;
8 for i=1:4
9     printf('\ny(%g) = %g\n', (i-1)*h, y(i))
```

```

10     y(i+1)=y(i)+h*f((i-1)*h+h/2,y(i)+h*f((i-1)*h,y(i)
        ))/2)
11 end
12 disp(" Computation Error in book solved example 10.7"
        )

```

Scilab code Exa 10.8 Picard Method

```

1 //Example 10.8
2 //Picard Method
3 //Page no. 313
4 clc;clear;close;
5 deff('y=f(x,y)', 'y=x^2-y')
6 y(1)=1;
7 for i=1:5
8     y(i+1)=y(1)+integrate('f(x,y(i))', 'x', 0, 0.2)
9     printf('\n y%g = %g\n', i, y(i+1))
10 end

```

Scilab code Exa 10.9 Picard Method

```

1 //Example 10.9
2 //Picard Method
3 //Page no. 313
4 clc;clear;close;
5 x=poly(0, 'x')
6 deff('y=f1(x,y)', 'y=x^2')
7 deff('y=f2(x,y)', 'y=2*x*y')
8 y(1)=0;
9 h=poly(0, 'x')
10 for i=1:4
11     for j=1:i
12         if j==1 then

```

```

13         y1(j)=y(1)+integrate('f1(x,y(j))', 'x'
14             ,0,1)
15     else
16         y1(j)=integrate('f2(x,y1(j))*(x^(2*j-1))'
17             , 'x',0,1)
18     end
19 end
20 printf("\n\n y%i = ",i)
21 for j=1:i
22     if j==i then
23         printf("x^%i * %g",2*j+1,y1(j))
24     else
25         printf("x^%i * %g + ",2*j+1,y1(j))
26     end
27 end
28 for j=i:-1:1
29     y1(j+1)=y1(j)
30 end

```

Scilab code Exa 10.10 Picard Method

```

1 //Example 10.10
2 //Picard Method
3 //Page no. 314
4 clc;clear;close;
5 deff('y=f(x,y)', 'y=1+x*y')
6 y(1)=1;
7 for i=1:5
8     y(i+1)=y(1)+integrate('f(x,y(i))', 'x',0,0.1)
9     printf('\n y%g = %.10g\n',i,y(i+1))
10 end

```

Scilab code Exa 10.11 Heun Method

```
1 //Example 10.11
2 //Heun Method
3 //Page no. 316
4 clc;clear;close;
5 deff('y=f(x,y)', 'y=(x-y)/2')
6 y=1;
7 h=0.5;
8 for i=1:4
9     x=(i-1)*h
10    x1=x+h
11    p=y+h*f(x,y)
12    y=y+h*(f(x,y)+f(x1,p))/2
13    printf('\n p(%g) = %g\n y(%g) = %g\n\n',i,p,i,
14           y)
15 end
```

Scilab code Exa 10.12 Third Order Runge Kutta Method

```
1 //Example 10.12
2 //Third Order Runge Kutta Method
3 //Page no. 322
4 clc;clear;close;
5 deff('y=f(x,y)', 'y=x^2-y')
6 y=1;h=0.1;
7 for i=1:2
8     x=(i-1)*h
9     K1=h*f(x,y);
10    K2=h*f(x+h/2,y+K1/2);
11    K3=h*f(x+h,y+K2);
```

```

12 y=y+(K1+4*K2+K3)/6
13 printf('\ny(%g) = %.9f\n\n',x+h,y)
14 end

```

Scilab code Exa 10.13 Fourth Order Runge Kutta Method

```

1 //Example 10.13
2 //Fourth Order Runge Kutta Method
3 //Page no. 323
4 clc;clear;close;
5 deff('y=f(x,y)', 'y=x+y')
6 y=1;x=0;h=0.1;
7 K1=h*f(x,y);
8 K2=h*f(x+h/2,y+K1/2);
9 K3=h*f(x+h/2,y+K2/2);
10 K4=h*f(x+h,y+K3);
11 disp(K4,'K4 =',K3,'K3 =',K2,'K2 =',K1,'K1 =')
12 y1=y+(K1+2*K2+2*K3+K4)/6
13 printf('\ny(1.1) = %.8f\n\n',y1)

```

Scilab code Exa 10.14 Fourth Order Runge Kutta Method for higher order equations

```

1 //Example 10.18
2 //Fourth Order Runge Kutta Method for higher order
  equations
3 //Page no. 328
4 clc;clear;close;
5 deff('y=f(x,y,z)', 'y=z')
6 deff('y=g(x,y,z)', 'y=(x^2-y^2)/(1+z^2)')
7 y=1;h=0.5;z=0;
8 for i=1:2
9     x=(i-1)*h

```



```

10     K(1)=h*f(x,y,z);
11     L(1)=h*g(x,y,z);
12     K(2)=h*f(x+h/2,y+K(1)/2,z+L(1)/2);
13     L(2)=h*g(x+h/2,y+K(1)/2,z+L(1)/2);
14     K(3)=h*f(x+h/2,y+K(2)/2,z+L(2)/2);
15     L(3)=h*g(x+h/2,y+K(2)/2,z+L(2)/2);
16     K(4)=h*f(x+h,y+K(3),z+L(3));
17     L(4)=h*g(x+h,y+K(3),z+L(3));
18     y=y+(K(1)+2*K(2)+2*K(3)+K(4))/6
19     z=z+(L(1)+2*L(2)+2*L(3)+L(4))/6
20     for j=1:4
21         printf('\n K%i = %g\t\tL%i = %g\n',j,K(j),j,
                L(j))
22     end
23     printf('\ny(%g) = %.8f\t\tz(%g) = %.8f\n\n\n\n',
            x+h,y,x+h,z)
24 end

```

Scilab code Exa 10.15 Fourth Order Runge Kutta Method

```

1 //Example 10.15
2 //Fourth Order Runge Kutta Method
3 //Page no. 324
4 clc;clear;close;
5 deff('y=f(x,y)', 'y=x^2+y^2')
6 y=1;h=0.1;
7 for i=1:2
8     x=(i-1)*h
9     K1=h*f(x,y);
10    K2=h*f(x+h/2,y+K1/2);
11    K3=h*f(x+h/2,y+K2/2);
12    K4=h*f(x+h,y+K3);
13    disp(K4,'K4 =',K3,'K3 =',K2,'K2 =',K1,'K1 =')
14    y=y+(K1+2*K2+2*K3+K4)/6
15    printf('\ny(%g) = %.13f\n\n\n\n',x+h,y)

```

16 end

Scilab code Exa 10.16 Fourth Order Runge Kutta Method

```
1 //Example 10.16
2 //Fourth Order Runge Kutta Method
3 //Page no. 326
4 clc;clear;close;
5 deff('y=f(x,y)', 'y=(2*x*y+exp(x))/(x^2+x*exp(x))')
6 y=0;h=0.2;
7 for i=1:2
8     x=1+(i-1)*h
9     K1=h*f(x,y);
10    K2=h*f(x+h/2,y+K1/2);
11    K3=h*f(x+h/2,y+K2/2);
12    K4=h*f(x+h,y+K3);
13    disp(K4, 'K4 =', K3, 'K3 =', K2, 'K2 =', K1, 'K1 =')
14    y=y+(K1+2*K2+2*K3+K4)/6
15    printf('\ny(%g) = %.13f\n\n\n\n', x+h, y)
16 end
```

Scilab code Exa 10.17 Fourth Order Runge Kutta Method for system of 1st order equations

```
1 //Example 10.17
2 //Fourth Order Runge Kutta Method for system of 1st
   order equations
3 //Page no. 327
4 clc;clear;close;
5 deff('y=f(x,y,z)', 'y=x+z')
6 deff('y=g(x,y,z)', 'y=x-y')
7 y=0;h=0.1;z=1;
8 for i=1:2
```

```

9      x=(i-1)*h
10     K(1)=h*f(x,y,z);
11     L(1)=h*g(x,y,z);
12     K(2)=h*f(x+h/2,y+K(1)/2,z+L(1)/2);
13     L(2)=h*g(x+h/2,y+K(1)/2,z+L(1)/2);
14     K(3)=h*f(x+h/2,y+K(2)/2,z+L(2)/2);
15     L(3)=h*g(x+h/2,y+K(2)/2,z+L(2)/2);
16     K(4)=h*f(x+h,y+K(3),z+L(3));
17     L(4)=h*g(x+h,y+K(3),z+L(3));
18     y=y+(K(1)+2*K(2)+2*K(3)+K(4))/6
19     z=z+(L(1)+2*L(2)+2*L(3)+L(4))/6
20     for j=1:4
21         printf('\n K%i = %g\t\tL%i = %g\n',j,K(j),j,
                L(j))
22     end
23     printf('\ny(%g) = %.8f\t\tz(%g) = %.8f\n\n\n\n',
            x+h,y,x+h,z)
24 end

```

Scilab code Exa 10.18 Fourth Order Runge Kutta Method for higher order equations

```

1 //Example 10.18
2 //Fourth Order Runge Kutta Method for higher order
  equations
3 //Page no. 328
4 clc;clear;close;
5 deff('y=f(x,y,z)', 'y=z')
6 deff('y=g(x,y,z)', 'y=(x^2-y^2)/(1+z^2)')
7 y=1;h=0.5;z=0;
8 for i=1:2
9     x=(i-1)*h
10    K(1)=h*f(x,y,z);
11    L(1)=h*g(x,y,z);
12    K(2)=h*f(x+h/2,y+K(1)/2,z+L(1)/2);

```

```

13     L(2)=h*g(x+h/2,y+K(1)/2,z+L(1)/2);
14     K(3)=h*f(x+h/2,y+K(2)/2,z+L(2)/2);
15     L(3)=h*g(x+h/2,y+K(2)/2,z+L(2)/2);
16     K(4)=h*f(x+h,y+K(3),z+L(3));
17     L(4)=h*g(x+h,y+K(3),z+L(3));
18     y=y+(K(1)+2*K(2)+2*K(3)+K(4))/6
19     z=z+(L(1)+2*L(2)+2*L(3)+L(4))/6
20     for j=1:4
21         printf('\n K%i = %g\t\tL%i = %g\n',j,K(j),j,
                L(j))
22     end
23     printf('\ny(%g) = %.8f\t\tz(%g) = %.8f\n\n\n',
            x+h,y,x+h,z)
24 end

```

Scilab code Exa 10.19 Adams Basforth formula

```

1 //Example 10.19
2 //Adams Basforth formula
3 //Page no. 333
4 clc;clear;close;
5 x=[0,0.1,0.2,0.3,0.4];i=5;
6 y=[1,1.0025,1.0101,1.0228];
7 h=0.1;
8 deff('y=f(x,y)', 'y=x*y/2')
9 //adams basforth formula
10 y(i)=y(i-1)+h*(55*f(x(i-1),y(i-1))-59*(f(x(i-2),y(i-2))
    -2))+37*f(x(i-3),y(i-3))-9*f(x(i-4),y(i-4)))/24
11 disp(y(i),"By Adams Basforth Formula : ")
12 //adams moulton formula
13 y(i)=y(i-1)+h*(9*f(x(i),y(i))+19*f(x(i-1),y(i-1))
    -5*(f(x(i-2),y(i-2))+f(x(i-3),y(i-3))))/24
14 disp(y(i),"By Adams Moulton Formula : ")

```

Scilab code Exa 10.20 Adams Moulton formula

```
1 //Example 10.20
2 //Adams Moulton formula
3 //Page no. 334
4 clc;clear;close;
5 x=[1,1.1,1.2,1.3,1.4];i=5;
6 y=[1,1.233,1.548488,1.978921];
7 h=0.1;
8 deff('y=f(x,y)', 'y=x^2*y+x^2')
9 //adams basforth formula
10 y(i)=y(i-1)+h*(55*f(x(i-1),y(i-1))-59*(f(x(i-2),y(i-2)))
    +37*f(x(i-3),y(i-3))-9*f(x(i-4),y(i-4)))/24
11 disp(y(i),"By Adams Basforth Formula : ")
12 //adams moulton formula
13 y(i)=y(i-1)+h*(9*f(x(i),y(i))+19*f(x(i-1),y(i-1))
    -5*(f(x(i-2),y(i-2)))+f(x(i-3),y(i-3)))/24
14 disp(y(i),"By Adams Moulton Formula : ")
```

Scilab code Exa 10.21 Adams formula

```
1 //Example 10.21
2 //Adams formula
3 //Page no. 335
4 clc;clear;close;
5 h=0.1;
6 deff('y=f(x,y)', 'y=x-y^2')
7 y(1)=1;
8 for i=1:5
9     x(i)=(i-1)*h
10    K(1)=h*f(x(i),y(i));
11 K(2)=h*f(x(i)+h/2,y(i)+K(1)/2);
```

```

12 K(3)=h*f(x(i)+h/2,y(i)+K(2)/2);
13 K(4)=h*f(x(i)+h,y(i)+K(3));
14 y(i+1)=y(i)+(K(1)+2*K(2)+2*K(3)+K(4))/6
15 printf('\ny(%g) = %.13f\n\n',x(i)+h,y(i+1))
16 end
17 i=5;
18 //adams basforth formula
19 y(i)=y(i-1)+h*(55*f(x(i-1),y(i-1))-59*(f(x(i-2),y(i
    -2))))+37*f(x(i-3),y(i-3))-9*f(x(i-4),y(i-4)))/24
20 disp(y(i),"By Adams Basforth Formula : ")
21 //adams moulton formula
22 y(i)=y(i-1)+h*(9*f(x(i),y(i))+19*f(x(i-1),y(i-1))
    -5*(f(x(i-2),y(i-2)))+f(x(i-3),y(i-3)))/24
23 disp(y(i),"By Adams Moulton Formula : ")

```

Scilab code Exa 10.22 Milne Simpson Predictor Corrector Method

```

1 //Example 10.22
2 //Milne Simpson Predictor Corrector Method
3 //Page no. 336
4 clc;clear;close;
5 deff('y=f11(x,y)', 'y=x^2+y^2-2')
6 deff('y=f22(x,y)', 'y=2*x+2*y*f11(x,y)')
7 deff('y=f33(x,y)', 'y=2+2*y*f22(x,y)+2*f11(x,y)^2')
8 deff('y=f44(x,y)', 'y=2*y*f33(x,y)+6*f11(x,y)*f22(x,y)
    ')')
9 h=0.1;
10 y=1;y1=y;
11 x(1)=0;k=y;
12 for i=2:3
13     x(i)=x(i-1)+h
14     for j=1:4
15         if j==1 then
16             k=k+(h^j)*f11(x(i-1),y)/factorial(j)
17         elseif j==2

```

```

18         k=k+(h^j)*f22(x(i-1),y)/factorial(j)
19     elseif j==3
20         k=k+(h^j)*f33(x(i-1),y)/factorial(j)
21     elseif j==4
22         k=k+(h^j)*f44(x(i-1),y)/factorial(j)
23     end
24 end
25 printf('\ny%i = %g\n\n',i-1,k)
26 if i==2 then
27     y=k;
28 else
29     y2=k;
30 end
31 end
32 k=y1;
33 for j=1:4
34     if j==1 then
35         k=k+(-h^j)*f11(x(1),y1)/factorial(j)
36     elseif j==2
37         k=k+(-h^j)*f22(x(1),y1)/factorial(j)
38     elseif j==3
39         k=k+(-h^j)*f33(x(1),y1)/factorial(j)
40     elseif j==4
41         k=k+(-h^j)*f44(x(1),y1)/factorial(j)
42     end
43 end
44 printf('\ny%i = %g\n\n',-1,k)
45 y3=k+4*h*(2*f11(x(1),y1)-f11(x(2),y)+2*f11(x(3),y2))
46     /3
47 printf('\nPredictor y(0.3) = %.9f\n\n',y3)
48 y4=y+h*(f11(x(3),y)+4*f11(x(3)+h,y2)+f11(x(3)+2*h,y3)
49     ))/3
50 printf('Corrector y(0.3) = %.9f',y4)
51 y3=y1+4*h*(2*f11(x(2),y)-f11(x(3),y2)+2*f11(x(3)+h,
52     y3))/3
53 printf('\n\nPredictor y(0.4) = %.9f\n\n',y3)
54 y4=y2+h*(f11(x(3)+h,y2)+4*f11(x(3)+2*h,y3)+f11(x(3)
55     +3*h,y3))/3

```

```
52 printf('Corrector y(0.4) = %.9f',y4)
```

Scilab code Exa 10.23 Milne Simpson Predictor Corrector Method

```
1 //Example 10.23
2 //Milne Simpson Predictor Corrector Method
3 //Page no. 338
4 clc;clear;close;
5 deff('y=f11(y)', 'y=2*y-y^2')
6 h=0.05;
7 y=[1,1.0499584,1.0996680,1.1488850]
8 for i=1:6
9     x(i)=(i-1)*h
10 end
11 for i=5:6
12     y(i)=y(i-4)+4*h*(2*f11(y(i-1))-f11(y(i-2))+2*f11
13         (y(i-3)))/3
14     printf('\nPredictor y(%g) = %.9f\n\n',x(i),y(i))
15     y(i)=y(i-2)+h*(f11(y(i-2))+4*f11(y(i-1))+f11(y(i
16         ))) /3
17     printf('Corrector y(%g) = %.9f\n\n',x(i),y(i))
18 end
```

Scilab code Exa 10.24 Milne Simpson Predictor Corrector Method

```
1 //Example 10.24
2 //Milne Simpson Predictor Corrector Method
3 //Page no. 339
4 clc;clear;close;
5 deff('y=f11(x,y)', 'y=1+x*y^2')
6 h=0.1;
7 y=[1,1.105,1.223,1.355]
8 for i=1:5
```



```

9      x(i)=(i-1)*h
10 end
11 i=5;
12     y(i)=y(i-4)+4*h*(2*f11(x(i-1),y(i-1))-f11(x(i-2)
      ,y(i-2))+2*f11(x(i-3),y(i-3)))/3
13     printf('\nPredictor y(%g) = %.9f\n\n',x(i),y(i))
14     y(i)=y(i-2)+h*(f11(x(i-2),y(i-2))+4*f11(x(i-1),y
      (i-1))+f11(x(i),y(i)))/3
15     printf('Corrector y(%g) = %.9f\n\n',x(i),y(i))

```

Scilab code Exa 10.25 Milne Simpsons formula

```

1 //Example 10.25
2 //Milne Simpsons formula
3 //Page no. 340
4 clc;clear;close;
5 h=0.1;
6 deff('y=f(x,y)', 'y=x*y+y^2')
7 y(1)=1;
8 for i=1:5
9     x(i)=(i-1)*h
10 end
11 for i=1:3
12     K(1)=h*f(x(i),y(i));
13     K(2)=h*f(x(i)+h/2,y(i)+K(1)/2);
14     K(3)=h*f(x(i)+h/2,y(i)+K(2)/2);
15     K(4)=h*f(x(i)+h,y(i)+K(3));
16     y(i+1)=y(i)+(K(1)+2*K(2)+2*K(3)+K(4))/6
17     for j=1:4
18         printf('\n K%i = %.4g\n',j,K(j))
19     end
20     printf('\ny(%g) = %.4f\n\n',x(i)+h,y(i+1))
21 end
22 i=5;
23     y(i)=y(i-4)+4*h*(2*f(x(i-1),y(i-1))-f(x(i-2),y(i

```

```

-2))+2*f(x(i-3),y(i-3)))/3
24 printf('\nPredictor y(%g) = %.4f\n\n',x(i),y(i))
25 y(i)=y(i-2)+h*(f(x(i-2),y(i-2))+4*f(x(i-1),y(i
-1))+f(x(i),y(i)))/3
26 printf('Corrector y(%g) = %.4f\n\n',x(i),y(i))

```

Scilab code Exa 10.26 Mullers Method

```

1 //Example 10.26
2 //Milne Simpson and Picard Method
3 //Page no. 341
4 clc;clear;close;
5 deff('y=f(x,y)', 'y=x-y^2')
6 y(1)=0;h=0.2;
7 for i=1:4
8     x(i)=(i-1)*h
9     y(i+1)=y(1)+integrate('f(x,y(i))', 'x',0,x(i))
10    printf('\n y%g = %.4g\n\n y' %g = %.4g\n\n',i-1,y
(i+1),i-1,f(x(i),y(i+1)))
11 end
12 for i=5:6
13     x(i)=(i-1)*h
14     if i==5 then
15         y1=y(i)
16     else
17         y1=y(i-1)
18     end
19     y(i)=y(i-3)+4*h*(2*f(x(i-1),y(i-(i-5)))-f(x(i-2)
,y(i-1))+2*f(x(i-3),y(i-2)))/3
20    printf('\nPredictor y(%g) = %.4f\n\n',x(i),y(i))
21    y(i)=y(i-1)+h*(f(x(i-2),y(i-2))+4*f(x(i-1),y1)+f
(x(i),y(i)))/3
22
23    printf('Corrector y(%g) = %.4f\n\n',x(i),y(i))
24 end

```

```
25 printf('\n\n\n\nNote : Computation error in book
    while calculation of predictor and corrector')
```

Scilab code Exa 10.33 Numerov Method

```
1 //Example 10.33
2 //Numerov Method
3 //Page no. 350
4 clc;clear;close;
5 k=0.5;h=%pi/6
6 y(1)=0;y(2)=k;
7 deff('y=f2(x,y)', 'y=-y')
8 deff('y=g()', 'y=-1')
9 fi=acos(((2+5*h^2*g()/6)-(1-h^2*g()/12)*y(1))/(2*(1-
    h^2*g()/12)))
10 y6=k*(sin(6*fi)/sin(fi))
11 disp(y6,"y6 = ")
```

Scilab code Exa 10.34 Numerov Method

```
1 //Example 10.34
2 //Numerov Method
3 //Page no. 351
4 clc;clear;close;
5 k=0.42;h=0.5
6 y(1)=0.5;y(2)=k;
7 deff('y=f2(x,y)', 'y=-y')
8 deff('y=g(x)', 'y=(x-1)*(x-2)')
9 for i=1:4
10     x(i)=(i-1)*h
11 end
12
13 for i=3:4
```

```

14     y(i) = ((2+5*h^2*g(x(i-1)))/6)*y(i-1) - (1-h^2*g(x(i
        -2))/12)*y(i-2))/(1-h^2*g(x(i))/12)
15     printf('\ny(%g) = %.6g\n',x(i),y(i))
16 end

```

Scilab code Exa 10.36 Finite Difference Method

```

1 //Example 10.36
2 //Finite Difference Method
3 //Page no. 353
4 clc;close;clear;
5
6 h=0.2;
7 y(1)=0;
8 deff('y=f2(x,y)', 'y=x+y')
9 for i=1:4
10     x(i)=i*h
11 end
12 for i=1:4
13     B(i,1)=h^2*x(i)
14     if i==4 then
15         B(4,1)=1-B(4,1)
16     end
17     printf('\ny%i - 2.04y%i + y%i = %g\n',i-1,i,i+1,
        B(i,1))
18 end
19 A=[-2-h^2,1,0,0;1,-2-h^2,1,0;0,1,-2-h^2,1;0,0,1,-2-h
    ^2]
20 C=inv(A)*B;
21 printf('\n\n')
22 for i=1:4
23     printf('\ny%i = %g\n',i,C(i))
24 end

```

Scilab code Exa 10.37 Finite Difference Method

```
1 //Example 10.37
2 //Finite Difference Method
3 //Page no. 354
4 clc;close;clear;
5
6 h=0.2;
7 y(1)=1;y(2)=1;
8 deff('y=f2(x,y)','y=x+y')
9 for i=1:4
10     x(i)=i*h
11 end
12 A=[0,1,0,0;1,0,1,0;0,1,0,1;0,0,1,0]
13 j=1;
14 for i=1:4
15     A(i,i)=-(1.96+2*x(i)^2)/(1+x(i)^2)
16 end
17 for i=1:4
18     B(i,1)=7*h^2*x(i)
19     if i==4 then
20         B(4,1)=2-B(4,1)
21     end
22     printf('\ny%i %gy%i + y%i = %g\n',i-1,A(i,i),i,i
23         +1,B(i,1))
24 end
25 C=inv(A)*B;
26 printf('\n\n')
27 for i=1:4
28     printf('\ny%i = %g\n',i,C(i))
29 end
```

Scilab code Exa 10.38 Finite Difference Method

```
1 //Example 10.38
2 //Finite Difference Method
3 //Page no. 354
4 clc;close;clear;
5
6 h=0.25;
7 y(1)=0;
8 deff('y=f2(x,y)', 'y=x+y')
9 for i=1:3
10     x(i)=i*h
11 end
12 A=[0,1,0;1,0,1;0,1,0]
13 j=1;
14 for i=1:3
15     A(i,i)=-(2-h^2*x(i)^2)
16 end
17 for i=1:3
18     B(i,1)=0
19     if i==3 then
20         B(3,1)=-1
21     end
22     printf('\ny%i %gy%i + y%i = %g\n',i-1,A(i,i),i,i
23         +1,B(i,1))
24 end
25 C=inv(A)*B;
26 printf('\n\n')
27 for i=1:3
28     printf('\ny%i = %g\n',i,C(i))
29 end
```

Scilab code Exa 10.39 Finite Difference Method

```
1 //Example 10.39
```

```

2 //Finite Difference Method
3 //Page no. 355
4 clc;close;clear;
5
6 h=0.25;
7 y(1)=0;
8 deff('y=f2(x,y)', 'y=x+y')
9 for i=1:3
10     x(i)=i*h
11 end
12 A=[0,1,0;1,0,1;0,1,0]
13 j=1;
14 for i=1:3
15     A(i,i)=-(2+64*h^2)
16 end
17 for i=1:3
18     B(i,1)=-10*h^2
19     printf('\ny%i %gy%i + y%i = %g\n',i-1,A(i,i),i,i
20           +1,B(i,1))
21 end
22 C=inv(A)*B;
23 printf('\n\n')
24 for i=1:3
25     printf('\ny%i = %g\n',i,C(i))
26 end

```

Scilab code Exa 10.40 Formula Method

```

1 //Example 10.40
2 //Formula Method
3 //Page no. 355
4 clc;clear;close;
5
6 deff('y=f(x,y)', 'y=x*y')
7 y(1)=0;y(6)=1;h=0.2;

```

```

8 for i=1:6
9     x(i)=(i-1)*h
10 end
11 A=eye(4,5)-eye(4,5)
12 B=eye(4,1)-eye(4,1)
13 B(4,1)=-y(6)
14 for i=1:4
15     A(i,i)=1;
16     A(i,i+1)=-2-h^2*x(i+1)
17     A(i,i+2)=1;
18 end
19 for i=1:4
20     for j=1:4
21         C(i,j)=A(i,j+1)
22     end
23 end
24 printf('\n\n')
25 A=C;
26 D=inv(A)*B
27 for i=1:4
28     y(i+1)=D(i);
29     printf('\ty%i = %.5f\t',i,y(i+1))
30 end
31 printf('\n\n-----\n')
32 k=0;
33 for i=1:6
34     for j=1:3
35         if j==1 then
36             D(i,j)=x(i)*y(i)
37             printf(' f%i\t%.4f\t',i-1,D(i,j))
38         elseif (i~=1 & i~=2) | k==1
39             D(i,j)=D(i,j-1)-D(i-1,j-1)
40             printf('%.4f\t',D(i,j))
41             if i==2 then
42                 k=2;
43             end
44         end
45     end

```



```

46     if i==1 then
47         k=1;
48     end
49     printf('\n')
50 end
51 printf('-----\n')
52 for i=1:4
53     B(i)=D(i+2,3)*(h^2)/12
54 end
55
56 B(4,1)=-(B(4,1)-y(6))
57 printf('\n\n')
58 for i=1:4
59     A(i,i)=-2
60 end
61 z=inv(A)*B
62 for i=1:4
63     printf('\tz%i = %.5f\t',i,z(i))
64 end
65 printf('\n\n')
66 for i=1:4
67     y(i+1)=y(i+1)+z(i);
68     printf('\ty%i = %.5f\t',i,y(i+1))
69 end
70 printf('\n\n\n\n\n Note : Computation errors in book
    ')

```

Scilab code Exa 10.41 Eigenvalue Problem

```

1 //Example 10.41
2 //Eigenvalue Problem
3 //Page no. 359
4 clc;close;clear;
5
6 h=0.25;

```

```

7 y(1)=0;
8 l=poly(0, 'lbd')
9 deff('y=f2(x,y)', 'y=x+y')
10 for i=1:3
11     x(i)=i*h
12 end
13 A=[0,1,0;1,0,1;0,1,0]
14 j=1;
15 for i=1:3
16     A(i,i)=-(2-l*h^2)
17 end
18 for i=1:3
19     B(i,1)=0
20     printf('\ny%i -(2-0.0625*lbd)y%i + y%i = %g\n',i
        -1,i,i+1,B(i,1))
21 end
22 disp(A)
23 disp(det(A),"Determinant of A =")
24 disp(roots(det(A)),"Roots = ")
25 a=roots(det(A))
26 disp(a(3),"Minimum Value =")

```

Chapter 11

Partial Differential Equations

Scilab code Exa 11.1 Gauss Seidel Method

```
1 //Example 11.1
2 //Gauss-Seidel Method
3 //Page no. 366
4 clc;clear;close;
5
6 U=[50,100,100,50;0,0,0,0;0,0,0,0;0,0,0,0]
7 A=[4,0,0,-1;0,4,-1,0;0,-1,4,0;-1,0,0,4] //
   equation matrix
8 B=[150;150;0;0] //solution matrix
9 X=inv(A)*B
10 for i=1:4
11     printf('\n U%i = %g\n',i,X(i))
12 end
13
14 //Jacobi method
15
16 for k=1:2
17     printf('\n')
18     p=0;
19     for i=1:2
20         for j=1:2
```

```

21         U(i+1,j+1)=X(i+p)
22     end
23     p=2;
24 end
25 p=3;
26 for i=2:3
27     for j=2:3
28         X(i+j-p)=(U(i,j-1)+U(i,j+1)+U(i-1,j)+U(i+1,j
                ))/4
29     end
30     p=2;
31 end
32 for i=1:4
33     printf('\n U%i(%i) = %g\n',i,k,X(i))
34 end
35 printf('\n')
36 end
37 printf('\nHence the solution is : \n\n')
38 for i=1:4
39     printf(' U%i = %g, ',i,X(i))
40 end

```

Scilab code Exa 11.2 Gauss Seidel Method

```

1 //Example 11.2
2 //Gauss-Seidel Method
3 //Page no. 368
4 clc;clear;close;
5
6 U=[0,1,2,0;1,0,0,4;2,0,0,5;0,4,5,0]
7 k=1;
8 for i=2:3
9     for j=2:3
10         if (i==2 & j==3) | (i==3 & j==2) then
11             U(i,j)=0

```

```

12         else
13             U(i,j)=(U(i-1,j)+U(i+1,j)+U(i,j-1)+U(i,j
                +1))/4
14         end
15         printf(" u%i=%g, ",k,U(i,j))
16         k=k+1
17     end
18 end
19 for l=1:7
20     printf('\n\n')
21     k=1;
22     for i=2:3
23         for j=2:3
24             U(i,j)=(U(i-1,j)+U(i+1,j)+U(i,j-1)+U(i,j+1))
                /4
25             printf("\n u%i(%i)=%g\n",k,l,U(i,j))
26             k=k+1
27         end
28     end
29 end

```

Scilab code Exa 11.3 Gauss Seidel Method

```

1 //Example 11.3
2 //Gauss-Seidel Method
3 //Page no. 370
4 clc;clear;close;
5
6 U=[60,60,60,60;40,0,0,50;20,0,0,40;0,10,20,30]
7 def('y=d(i,j)', 'y=(U(i-1,j-1)+U(i+1,j+1)+U(i-1,j+1)
    +U(i+1,j-1))/4') //diagonal 5 point
    formula
8 def('y=s(i,j)', 'y=(U(i-1,j)+U(i+1,j)+U(i,j-1)+U(i,j
    +1))/4') //std 5 point formula
9 U(2,2)=d(2,2);

```

```

10 for k=0:5
11     for i=2:3
12         p=3;
13         for j=2:3
14             if k==0 & i==2 & j==2 then
15                 U(i,j)=d(i,j)
16             else
17                 U(i,j)=s(i,j)
18             end
19             if k==0 then
20                 printf('\n U%i = %g\n',i+j-p,U(i,j))
21             else
22                 printf('\n U%i(%i) = %g\n',i+j-p,k,U
23                     (i,j))
24             end
25         end
26     end
27     printf('\n\n')
28 end
29 printf('\nHence the solution is : \n\n')
30 for i=2:3
31     for j=2:3
32         printf(' U%i = %g, ',i,U(i,j))
33     end
34 end

```

Scilab code Exa 11.4 Gauss Seidel Method

```

1 //Example 11.4
2 //Gauss-Seidel Method
3 //Page no. 372
4 clc;clear;close;
5
6 U=[1,2,2,2;0,0,0,2;0,0,0,2;0,0,0,1]

```

```

7  defff('y=d(i,j)', 'y=(U(i-1,j-1)+U(i+1,j+1)+U(i-1,j+1)
    +U(i+1,j-1))/4') //diagonal 5 point
    formula
8  defff('y=s(i,j)', 'y=(U(i-1,j)+U(i+1,j)+U(i,j-1)+U(i,j
    +1))/4') //std 5 point formula
9  U(2,2)=d(2,2);
10 for k=0:4
11     for i=2:3
12         p=3;
13         for j=2:3
14             if k==0 & i==2 & j==2 then
15                 U(i,j)=d(i,j)
16             else
17                 U(i,j)=s(i,j)
18             end
19             if k==0 then
20                 printf('\n U%i = %g\n', i+j-p, U(i,j))
21             else
22                 printf('\n U%i(%i) = %g\n', i+j-p, k, U
                    (i,j))
23             end
24         end
25     end
26     p=2;
27     printf('\n\n')
28 end
29 printf('\nHence the solution is : \n\n')
30 for i=2:3
31     for j=2:3
32         printf(' U%i = %.3f, ', i, U(i,j))
33     end
34 end

```

Scilab code Exa 11.5 Gauss Seidel Method

```

1 //Example 11.5
2 //Gauss-Seidel Method
3 //Page no. 373
4 clc;clear;close;
5
6 U
   =[0,500,1000,500,0;1000,0,0,0,1000;2000,0,0,0,2000;1000,0,0,0,1000;0,1000,1000,1000,0];

7 deff('y=d(i,j)', 'y=(U(i-1,j-1)+U(i+1,j+1)+U(i-1,j+1)
   +U(i+1,j-1))/4') //diagonal 5 point
   formula
8 deff('y=s(i,j,1)', 'y=(U(i-1,j)+U(i+1,j)+U(i,j-1)+U(i
   ,j+1))/4') //std 5 point formula
9 U(3,3)=s(3,3,2);
10 for k=0:10
11     p=3;
12     for i=2:4
13         for j=2:4
14             if k==0 & (i==3 & j==3) | (i==2 & j==4)
   | (i==4 & j==2) | (i==4 & j==4) then
15                 printf('\n U%i(%i) = %g\n',i+j-p,k,U
   (i,j))
16                 continue
17             end
18             if k==0 & i==2 & j==2 then
19                 U(i,j)=d(i,j)
20             else
21                 U(i,j)=s(i,j,1)
22             end
23             if i==2 & j==2 then
24                 U(2,4)=U(2,2);
25                 U(4,2)=U(2,2);
26                 U(4,4)=U(2,2);
27             end
28             if k==0 then
29                 printf('\n U%i = %g\n',i+j-p,U(i,j))
30             else
31                 printf('\n U%i(%i) = %g\n',i+j-p,k,U

```



```

                                (i,j))
32         end
33     end
34     p=p-2;
35 end
36 printf('\n\n')
37 end
38 printf('\nHence the solution is : \n\n')
39 p=3;
40 for i=2:4
41     for j=2:4
42         printf(' U%i = %.3f, ',i+j-p,U(i,j))
43     end
44     p=p-2
45 end

```

Scilab code Exa 11.6 Gaussian Elimination Method

```

1 //Example 11.6
2 //Gaussian Elimination Method
3 //Page no. 374
4 clc;clear;close;
5 A
   =[-4,1,1,0,-80;1,-4,0,1,-10;1,0,-4,1,-160;0,1,1,-4,-90]
   //augmented matrix
6 disp(A,'Augmented Matrix=')
7 C=A;
8 //triangularization
9 for i=1:4
10     for j=1:5
11         if i==1 then
12             B(i,j)=A(i,j)
13         elseif i==2
14             B(i,j)=A(i,j)-A(i,1)*A(i-1,j)/A(1,1)
15             B(i+1,j)=A(i+1,j)-A(i+1,1)*A(i-1,j)/A

```

```

        (1,1)
16         B(i+2,j)=A(i+2,j)-A(i+2,1)*A(i-1,j)/A
           (1,1)
17     elseif i==3
18         if j==1 then
19             C=B
20         else
21             B(i,j)=B(i,j)-C(i,2)*B(i-1,j)/B(2,2)
22             B(i+1,j)=C(i+1,j)-C(i+1,2)*C(i-1,j)/
                C(2,2)
23         end
24     else
25         if j==1 then
26             C=B
27         end
28         B(i,j)=B(i,j)-C(i,3)*B(i-1,j)/B(3,3)
29     end
30 end
31 end
32
33 disp(B, 'Triangulated Matrix=')
34 //back substitution
35 x(4)=B(4,5)/B(4,4);
36 printf('\n p(4) = %.2f\n',x(4))
37 for i=3:-1:1
38     k=0
39     for j=i+1:4
40         k=k+B(i,j)*x(j)
41     end
42     x(i)=(1/B(i,i))*(B(i,5)-k)
43     printf('\n p(%i) = %.2f\n',i,x(i))
44 end

```

Scilab code Exa 11.7 Relaxation Method

```

1 //Example 11.7
2 //Relaxation Method
3 //Page no. 376
4 clc;clear;close;
5
6 for i=0:4
7     for j=0:4
8         if i==0 | j==0 then
9             U(5-i,j+1)=0
10        elseif i==4 | j==4
11            U(5-i,j+1)=(i*j)^2
12        else
13            U(5-i,j+1)=0;
14        end
15    end
16 end
17 S=['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I']
18 disp(U)
19 deff('y=d(i,j)', 'y=(U(i-1,j-1)+U(i+1,j+1)+U(i-1,j+1)
    +U(i+1,j-1))/4') //diagonal 5 point
    formula
20 deff('y=s(i,j,l)', 'y=(U(i-1,j)+U(i+1,j)+U(i,j-1)+U(i
    ,j+1))/4') //std 5 point formula
21 U(3,3)=s(3,3,2);
22 for k=0:0
23     p=3;
24     for i=2:4
25         for j=2:4
26             if k==0 & (i==3 & j==3) then
27                 printf('\n U %s(%i) = %g\n',S(i+j-p)
                    ,k,U(i,j))
28                 continue
29             end
30             if k==0 & i==4 & j==2 then
31                 U(i,j)=d(i,j)
32             else
33                 U(i,j)=s(i,j,1)
34             end

```

```

35         if k==0 then
36             printf('\n U %s = %g\n',S(i+j-p),U(i
                ,j))
37         else
38             printf('\n U %s(%i) = %g\n',S(i+j-p)
                ,k,U(i,j))
39         end
40     end
41     p=p-2;
42 end
43 printf('\n\n')
44 end
45 printf('\nHence the solution is : \n\n')
46 p=3;
47 for i=2:4
48     for j=2:4
49         printf(' U%s = %.3f, ',S(i+j-p),U(i,j))
50     end
51     p=p-2
52 end

```

Scilab code Exa 11.8 Relaxation Method

```

1 //Example 11.7
2 //Relaxation Method
3 //Page no. 378
4 clc;clear;close;
5 h=1/3;k=1/3;
6 for i=0:3
7     for j=0:3
8         if i==0 | j==0 then
9             U(4-i,j+1)=i*h+j*k
10        elseif i==3 | j==3
11            U(4-i,j+1)=i*h+j*k
12        end

```

```

13     end
14 end
15 //disp(U, 'U = ')
16 for i=1:4
17     for j=1:4
18         if U(i,j)==0 then
19             U(i,j)=1;
20         end
21     end
22 end
23 U(3,2)=U(3,2)-1/3;
24 U(2,2)=U(2,2)-1/3;
25 U(3,3)=U(3,3)-1/3;
26 U(3,2)=U(3,2)-1/3;
27 U(2,3)=U(2,3)+1/3;
28 //disp(U, 'U = ')
29 for i=2:3
30     for j=2:3
31         U1(i,j)=U(i+1,j)+U(i-1,j)+U(i,j+1)+U(i,j-1)
32             -4*U(i,j)
33     end
34 for i=2:3
35     for j=2:3
36         U(i,j)=U1(i,j)
37     end
38 end
39 //disp(U, 'U = ')
40 disp(U, 'U = ')
41 disp(' ')
42 k=1;
43 for i=2:3
44     for j=2:3
45         printf('\t u%i = %g ',k,U(i,j))
46         k=k+1
47     end
48 end

```

Scilab code Exa 11.9 Gauss Seidel Method

```

1 //Example 11.9
2 //Gauss-Seidel Method
3 //Page no. 380
4 clc;clear;close;
5
6 U=eye(4,4)-eye(4,4)
7 U(2,1)=150;U(3,1)=120;
8 U(2,4)=180;U(3,4)=150
9 deff('y=d(i,j)', 'y=(U(i-1,j-1)+U(i+1,j+1)+U(i-1,j+1)
    +U(i+1,j-1))/4') //diagonal 5 point
    formula
10 deff('y=s(i,j)', 'y=(U(i-1,j)+U(i+1,j)+U(i,j-1)+U(i,j
    +1))/4') //std 5 point formula
11 for k=1:6
12     for i=2:3
13         p=3;
14         for j=2:3
15             U(i,j)=s(i,j)
16             if i==2 & j==2 then
17                 U(i+1,j+1)=U(i,j)
18             elseif i==3 & j==3
19                 continue
20             end
21             printf('\n U%i(%i) = %g\n',i+j-p,k,U
                (i,j))
22         end
23         p=2;
24     end
25     printf('\n\n')
26 end
27 printf('\nHence the solution is : \n\n')
28 p=3;

```

```

29 for i=2:3
30
31     for j=2:3
32         printf(' U%i = %.3f, ',i+j-p,U(i,j))
33     end
34     p=2
35 end

```

Scilab code Exa 11.10 Gauss Seidel Method

```

1 //Example 11.10
2 //Gauss Seidel Method
3 //Page no. 382
4 clc;clear;close;
5
6 O=0.5;
7 A=[-8,8,0;2,-8,4;0,4,-8]; //equation matrix
8 B=[-1;-1;-1]; //solution matrix
9 Ov=inv(A)*B;
10 disp(Ov,'Values = ')
11 Ox=Ov(1)+(Ov(1)-O)/3
12 disp(Ox,'O* = ')

```

Scilab code Exa 11.11 Eigenvalue Problem

```

1 //Example 11.11
2 //Eigenvalue Problem
3 //Page no. 383
4 clc;clear;close;
5 h1=1;h2=3/4;
6 lbd1=2;
7 lbd=poly(0,'lbd')
8 mu=9*lbd/16;

```

```

9 A=[4-mu,-2,0;-2,4-mu,-1;0,-4,4-mu];
10 disp(determ(A),'Characteristic Equation = ');
11 r=roots(determ(A))
12 disp(r,'Roots = ')
13 r1=r(3)
14 Q=((h1/h2)^2*r1-lbd1)/((h1/h2)^2-1)
15 disp(Q,'Q12 = ')

```

Scilab code Exa 11.12 Eigenvalue Problem

```

1 //Example 11.12
2 //Eigenvalue Problem
3 //Page no. 385
4 clc;clear;close;
5
6 h1=1/4;h2=1/5;
7 lbd=poly(0,'lbd')
8 mu=9*lbd/16;
9 A=[lbd-64,16;32,lbd-64];
10 disp(determ(A),'Characteristic Equation = ');
11 r=roots(determ(A))
12 disp(r,'Roots = ')
13 r1(1)=r(2)
14 A=[lbd-100,0,25;0,lbd-100,50;25,50,lbd-100];
15 disp(determ(A),'Characteristic Equation = ');
16 r=roots(determ(A))
17 disp(r,'Roots = ')
18 r1(2)=r(3)
19 Q=((h1/h2)^2*r1(2)-r1(1))/((h1/h2)^2-1)
20 disp(Q,'Q12 = ')

```

Scilab code Exa 11.13 Eigenvalue Problem


```

1 //Example 11.13
2 //Eigenvalue Problem
3 //Page no. 387
4 clc;clear;close;
5
6 h1=1/2;h2=1/3;
7 lbd=poly(0,'lbd')
8 mu=9*lbd/16;
9 r1(1)=64
10 A=[2*lbd-324,81;243,lbd-324];
11 disp(determ(A),'Characteristic Equation = ');
12 r=roots(determ(A))
13 disp(r,'Roots = ')
14 r1(2)=r(2)
15 Q=((h1/h2)^2*r1(2)-r1(1))/((h1/h2)^2-1)
16 disp(Q,'Q12 = ')

```

Scilab code Exa 11.14 Crank Nicolson Method

```

1 //Example 11.14
2 //Crank Nicolson Method
3 //Page no. 390
4 clc;clear;close;
5 h=1/2;k=1/8;
6 r=k/h^2;
7 for i=1:3
8     for j=1:2
9         if i==1 | j==1 then
10             u(i,j)=0;
11         end
12         if i==3 then
13             u(i,j)=(j-1)*k
14         end
15     end
16 end

```

```

17 for j=2:2
18     u(2,j)=(u(1,j-1)+2*u(2,j-1)+u(3,j-1)+u(1,j)+
           u(3,j))/6
19 end
20 u=u'
21 printf('\nfor h = 1/2 and k=1/8\n\n')
22 printf('i\\j --> ')
23 for i=1:1
24     printf('\tu%i\t',i)
25 end
26 printf('\n
           -----\n'
           )
27 for i=2:2
28     for j=2:2
29         printf('\t  %.9f',u(i,j))
30     end
31 end
32
33
34
35
36 h=1/4;k=1/8;
37 r=k/h^2;
38 for i=1:5
39     for j=1:5
40         if i==1 | j==1 then
41             u(i,j)=0;
42         end
43         if i==5 then
44             u(i,j)=(j-1)*k
45         end
46     end
47 end
48 a=[3,-1,0;-1,3,-1;0,-1,3];
49 a=inv(a);
50 for j=2:5
51     b=[u(1,j-1)-u(2,j-1)+u(3,j-1)+u(1,j);u(2,j

```

```

-1)-u(3,j-1)+u(4,j-1);u(3,j-1)-u(4,j-1)+u
(5,j-1)+u(5,j)]
52     x=a*b
53     u(2,j)=x(1);u(3,j)=x(2);u(4,j)=x(3);
54 end
55 u=u'
56 printf('\n\n\n\n\n\nfor h = 1/4 and k=1/8\n\n')
57 printf('i\\j --> ')
58 for i=1:3
59     printf('\tu%i\t',i)
60 end
61 printf('\n
-----\
n')
62 for i=2:2
63     for j=2:4
64         printf('\t    %.9f',u(i,j))
65     end
66 end
67
68
69
70
71
72 h=1/4;k=1/16;
73 r=k/h^2;
74 for i=1:5
75     for j=1:3
76         if i==1 | j==1 then
77             u(i,j)=0;
78         end
79         if i==5 then
80             if j==3 then
81                 k=1/8;
82             end
83             u(i,j)=(j-1)*k
84         end
85     end

```

```

86 end
87 a=[4, -1, 0; -1, 4, -1; 0, -1, 4];
88 a=inv(a);
89 for j=2:3
90     b=[u(1, j-1)-u(2, j-1)+u(3, j-1)+u(1, j); u(2, j
          -1)-u(3, j-1)+u(4, j-1); u(3, j-1)-u(4, j-1)+u
          (5, j-1)+u(5, j)]
91     x=a*b
92     u(2, j)=x(1); u(3, j)=x(2); u(4, j)=x(3);
93 end
94 u=u'
95 printf('\n\n\n\n\n\nfor h = 1/4 and k=1/16\n\n')
96 printf('i\\j --> ')
97 for i=1:3
98     printf('\tu%i\t', i)
99 end
100 printf('\n
-----\
n')
101 for i=2:3
102     printf('\n')
103     for j=2:4
104         printf('\t %.9f', u(i, j))
105     end
106 end
-----

```

Scilab code Exa 11.15 Bender Schmidt Method

```

1 //Example 11.15
2 //Bender Schmidt Method
3 //Page no. 393
4 clc;clear;close;
5
6 h=1;k=1;c=1/sqrt(2);
7 r=k*c^2/h^2;

```

```

8 for i=1:5
9     u(4,i)=i-1;
10 end
11 k=0;
12 for i=4:-1:1
13     u(i,1)=0
14     k=k+1;
15 end
16 k=1;
17 for i=2:5
18     u(4,i)=k*(4-k)
19     k=k+1;
20 end
21 disp(u, 'u = ')
22 k=1;
23 printf('\n\n')
24 for i=3:-1:1
25     for j=2:4
26         u(i,j)=(u(i+1,j-1)+u(i+1,j+1))/2
27         printf('\n\tu%i = %g\n',k,u(i,j))
28         k=k+1;
29     end
30 end

```

Scilab code Exa 11.16 Crank Nicolson Method

```

1 //Example 11.16
2 //Crank Nicolson Method
3 //Page no. 394
4 clc;clear;close;
5 //case 1
6 h=1/4;k=1/8;
7 r=k/h^2;
8 n=1/h+1;
9 for i=1:2

```

```

10     for j=1:n
11         if i==2 then
12             u(i,j)=sin(%pi*(j-1)*h)
13         end
14         if j==1 | j==n then
15             u(i,j)=0;
16         end
17     end
18 end
19
20 a=[3,-1,0;-1,3,-1;0,-1,3];
21 a=inv(a);
22 for j=2:4
23
24 end
25 for i=2:-1:2
26     for j=2:4
27         b(j-1)=u(i,j-1)+(1-r)*u(i,j)+u(i,j+1)
28     end
29     x=a*b
30     u(i-1,2)=x(1);u(i-1,3)=x(2);u(i-1,4)=x(3);
31 end
32 printf('\nfor h = 1/4 and k=1/8\n\n')
33 for i=1:1
34     for j=2:4
35         printf('\t u%i = %.9f',j-1,u(i,j))
36     end
37 end
38 printf('\n\n\n')
39
40
41
42 //case 2
43 h=1/4;k=1/16;
44 r=k/h^2;
45 n=1/h+1;
46 for i=1:3
47     for j=1:n

```

```

48         if i==3 then
49             u(i,j)=sin(%pi*(j-1)*h)
50         end
51         if j==1 | j==n then
52             u(i,j)=0;
53         end
54     end
55 end
56
57 a=[4,-1,0;-1,4,-1;0,-1,4];
58 a=inv(a);
59 for j=2:4
60
61 end
62 for i=3:-1:2
63     for j=2:4
64         b(j-1)=u(i,j-1)+(1-r)*u(i,j)+u(i,j+1)
65     end
66     x=a*b
67     u(i-1,2)=x(1);u(i-1,3)=x(2);u(i-1,4)=x(3);
68 end
69 printf('\nfor h = 1/4 and k=1/16\n\n')
70 l=1;
71 for i=2:-1:1
72     for j=2:4
73         printf('\t u%i = %.9f',j+1-i,u(i,j))
74     end
75     printf('\n')
76     l=3;
77 end
78 printf('\n\n\n')
79 printf('The Analytical Solution u1 = %g',exp(-%pi^2*k
80         )*sin(%pi*h))
81 printf('\n\n\n\nNote : Computation Errors in the
82         book')

```

Scilab code Exa 11.17 Bender Schmidt Method

```
1 //Example 11.17
2 //Bender Schmidt Method
3 //Page no. 396
4 clc;clear;close;
5
6 h=1;k=1/10;c=sqrt(5);
7 r=k*c^2/h^2;
8 for i=1:6
9     if i<4 then
10         u(6,i)=20*(i-1)
11     else
12         u(6,i)=60
13     end
14 end
15 disp(u,'u = ')
16 k=1;
17 printf('\n\n')
18 for i=5:-1:1
19     for j=2:6
20         if j~=6 then
21             u(i,j)=(u(i+1,j-1)+u(i+1,j+1))/2
22         else
23             u(i,j)=60
24         end
25         printf('\t u%i = %g \t',k,u(i,j))
26         k=k+1;
27     end
28 end
29 printf('\n')
30 printf('\n\n j\\i | \t')
31 for i=1:6
32     printf('%i\t',i-1)
```



```

33 end
34 printf('\n')
35 for i=1:51
36     printf('_ ')
37 end
38
39 k=0;
40 for i=6:-1:1
41     printf('\n %i   |\t',k)
42     for j=1:6
43         printf('%g\t',u(i,j))
44     end
45     k=k+1;
46 end

```

Scilab code Exa 11.18 Bender Schmidt Method

```

1 //Example 11.18
2 //Bender Schmidt Method
3 //Page no. 398
4 clc;clear;close;
5
6 h=1;k=1/8;c=sqrt(4);
7 r=k*c^2/h^2;
8 deff('y=f(x)', 'y=4*x-x^2/2')
9 for i=1:9
10     if i~=1 & i~=9 then
11         u(6,i)=f(i-1)
12     else
13         u(6,i)=0
14     end
15 end
16 k=1;
17 printf('\n\n')
18 for i=5:-1:1

```

```

19     for j=2:8
20         u(i,j)=(u(i+1,j-1)+u(i+1,j+1))/2
21         printf('\t u%i = %.4f \t',k,u(i,j))
22         k=k+1;
23     end
24 end
25 printf('\n')
26 printf('\n\n j\\i | \t')
27 for i=1:9
28     printf(' %i\t',i-1)
29 end
30 printf('\n')
31 for i=1:80
32     printf('_ ')
33 end
34
35 k=0;
36 for i=6:-1:1
37     printf('\n %i | \t',k)
38     for j=1:9
39         printf('%.4f\t',u(i,j))
40     end
41     k=k+1;
42 end

```

Scilab code Exa 11.19 Gauss Seidel Method

```

1 //Example 11.19
2 //Gauss Seidel Method
3 //Page no. 399
4 clc;clear;close;
5
6 h=0.2;k=0.02;r=k/h^2;
7 deff('y=f(x)', 'y=sin(%pi*x)')
8 n=1/h+1;

```

```

9  for i=1:n
10     u(n,i)=f((i-1)*h)
11  end
12  disp(u)
13  m=1;l=1;
14  printf('\n\n')
15  for i=5:-1:1
16     for j=2:5
17         u(i,j)=(u(i,j-1)+u(i+1,j+1))/6+2*(u(i+1+l-1,
                j)+r*(u(i+1+l-1,j-1)-2*(u(i+1+l-1,j))+u(i
                +1+l-1,j+1))/2)/3
18         printf(' u%i(%i) = %g \t ',m,l,u(i,j))
19         m=m+1;
20     end
21     printf('\n')
22     l=l+1
23 end
24 printf('\n\n\n')
25 printf('The Analytical Solution u1 = %g',exp(-%pi^2*k
        )*sin(%pi*h))

```

Scilab code Exa 11.20 Finite Difference Method

```

1  //Example 11.20
2  //Finite Difference Method
3  //Page no. 403
4  clc;clear;close;
5
6  h=1;k=0.5;c=sqrt(4);
7  r=k^2*c^2/h^2;
8  for i=2:5
9     if i<5 then
10        u(4,i)=(i-1)*(4-(i-1))
11        u(5,i)=(i-1)*(4-(i-1))
12    else

```

```

13         u(5,i)=0
14     end
15 end
16 disp(u, 'u = ')
17 k=2;
18 printf('\n\n')
19 for i=3:-1:1
20     for j=2:4
21         u(i,j)=u(i+1,j-1)+u(i+1,j+1)-u(i+2,j)
22         printf('\tu%i,%i = %g',j-1,k,u(i,j))
23     end
24     k=k+1;
25     printf('\n')
26 end

```

Scilab code Exa 11.21 Finite Difference Method

```

1 //Example 11.21
2 //Finite Difference Method
3 //Page no. 404
4 clc;clear;close;
5
6 h=1;k=0.25;c=sqrt(16);
7 r=k^2*c^2/h^2;
8 for i=2:6
9     if i<6 then
10         u(6,i)=(i-1)^2*(5-(i-1))
11         u(5,i)=(i-1)^2*(5-(i-1))
12     else
13         u(5,i)=0
14     end
15 end
16 disp(u, 'u = ')
17 k=2;
18 printf('\n\n')

```

```
19 for i=4:-1:1
20     for j=2:5
21         u(i,j)=u(i+1,j-1)+u(i+1,j+1)-u(i+2,j)
22         printf('\tu%i,%i = %g',j-1,k,u(i,j))
23     end
24     k=k+1;
25     printf('\n')
26 end
```
