

Scilab Textbook Companion for  
Theory Of Machines  
by R. S. Khurmi And J. K. Gupta<sup>1</sup>

Created by  
Shreerang Deshpande  
Bachelor of Engineering  
Mechanical Engineering  
B.V.Bhoomaraddi College of Engg. & Tech.  
College Teacher  
P.g.tewari  
Cross-Checked by

May 19, 2016

<sup>1</sup>Funded 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:** Theory Of Machines

**Author:** R. S. Khurmi And J. K. Gupta

**Publisher:** Eurasia Publishing House (Pvt.) Ltd., New Delhi

**Edition:** 14

**Year:** 2009

**ISBN:** 81-219-2524-X

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
2 Kinematics of Motion	5
3 Kinetics of Motion	10
4 Simple Harmonic Motion	32
5 Simple Mechanisms	42
6 Velocity in Mechanisms Instantaneous Centre Method	46
7 Velocity in Mechanisms Relative Velocity Method	52
8 Acceleration in Mechanisms	66
9 Mechanisms with Lower Pairs	90
10 Friction	94
11 Belt Rope and Chain Drives	129
12 Toothed Gearing	156
13 Gear Trains	178
14 Gyroscopic Couple and Precessional Motion	202
15 Inertia Forces in Reciprocating Parts	220



# List of Scilab Codes

Exa 2.1	To Find the Acceleration and Distance . . . . .	5
Exa 2.3	To Find the Velocity . . . . .	6
Exa 2.5	To Find the Maximum Cutting Speed . . . . .	7
Exa 2.6	To Find the Angular Acceleration . . . . .	7
Exa 2.7	To Find Velocity and Acceleration . . . . .	8
Exa 3.1	To find the angular acceleration and KE . . . . .	10
Exa 3.2	To find the time torque and power . . . . .	11
Exa 3.3	To find the reduction of speed . . . . .	13
Exa 3.4	To find the torque and power . . . . .	14
Exa 3.5	To find the KE and braking force . . . . .	15
Exa 3.7	To Find the Speed and Energy Lost . . . . .	17
Exa 3.8	To find the velocity KE and compression . . . . .	17
Exa 3.9	To find energy lost and resistance . . . . .	19
Exa 3.10	To find the angular velocities . . . . .	20
Exa 3.11	To find the velocity impulse angle of swing and average force . . . . .	21
Exa 3.12	To find the speed time and KE lost . . . . .	22
Exa 3.13	To find the acceleration . . . . .	24
Exa 3.18	To find the torque . . . . .	25
Exa 3.19	To find velocities and loss of KE . . . . .	27
Exa 3.20	To find the speed and energy dissipated . . . . .	28
Exa 3.21	To find strain energy twist and speed . . . . .	30
Exa 4.1	To Find the Velocity and Acceleration . . . . .	32
Exa 4.2	To Find the Angular Velocity Time and Acceleration . . . . .	32
Exa 4.3	To Find the Frequency and Velocity . . . . .	33
Exa 4.4	To Find the Frequency of Oscillation . . . . .	34
Exa 4.5	To Find the Moment of Inertia . . . . .	35
Exa 4.6	To Find the Moment of Inertia . . . . .	36

Exa 4.7	To Find the Time of Swing . . . . .	37
Exa 4.8	To Find Centre of Percussion and Impulse . . . . .	38
Exa 4.9	To Find the Radius of Gyration . . . . .	39
Exa 4.10	To Find the Radius of Gyration . . . . .	40
Exa 4.11	To Find the Mass Moment of Inertia . . . . .	40
Exa 5.1	To find the time ratio . . . . .	42
Exa 5.2	To find the time ratio . . . . .	43
Exa 5.3	To find the dimensions of AC and AP . . . . .	43
Exa 5.4	To find the time ratio . . . . .	44
Exa 6.1	To find the angular velocity . . . . .	46
Exa 6.2	To find velocity and angular velocity . . . . .	47
Exa 6.3	To find velocity and angular velocity . . . . .	47
Exa 6.4	To find the velocity . . . . .	49
Exa 6.5	To find the velocity . . . . .	49
Exa 6.6	To find the velocity of ram . . . . .	50
Exa 7.1	To find the angular velocity . . . . .	52
Exa 7.2	To find velocities angular velocities and position . . . . .	53
Exa 7.3	To find the velocity . . . . .	54
Exa 7.4	To find velocity angular velocity and rubbing speed . . . . .	55
Exa 7.5	To find velocity and angular velocity . . . . .	56
Exa 7.6	To find the absolute velocity . . . . .	57
Exa 7.7	To find linear and angular velocity . . . . .	57
Exa 7.8	To find speed and time ratio . . . . .	58
Exa 7.9	To find velocity and angular velocity . . . . .	59
Exa 7.10	To find velocity and resisting torque . . . . .	60
Exa 7.11	To find velocity ratio . . . . .	61
Exa 7.12	To find velocity angular velocity and torque . . . . .	62
Exa 7.13	To find the velocities . . . . .	64
Exa 8.1	To find linear and angular velocity and acceleration . . . . .	66
Exa 8.2	To find linear and angular velocity and acceleration . . . . .	67
Exa 8.3	To find linear and angular acceleration . . . . .	68
Exa 8.4	To find the angular velocity and acceleration . . . . .	70
Exa 8.5	To find angular velocities and accelerations . . . . .	71
Exa 8.6	To find velocities and accelerations . . . . .	72
Exa 8.7	To find linear and angular acceleration . . . . .	74
Exa 8.8	To find linear and angular velocities and accelerations . . . . .	75
Exa 8.9	To find velocity and accelerations . . . . .	76
Exa 8.10	To find velocity torque and acceleration . . . . .	78

Exa 8.11	To find velocity and acceleration . . . . .	80
Exa 8.12	To find the velocity and acceleration . . . . .	81
Exa 8.13	To find velocity and acceleration . . . . .	82
Exa 8.14	To find the acceleration . . . . .	84
Exa 8.15	To find linear and angular acceleration . . . . .	85
Exa 8.16	To find linear and angular acceleration . . . . .	86
Exa 8.17	To find velocities and acceleration . . . . .	87
Exa 9.1	To find inclination of track arm . . . . .	90
Exa 9.3	To find the angle turned . . . . .	90
Exa 9.4	To find the greatest permissible angle . . . . .	91
Exa 9.5	To find speeds and permissible angle . . . . .	92
Exa 9.7	To find speeds of shafts . . . . .	92
Exa 10.1	To find weight and coefficient of friction . . . . .	94
Exa 10.2	To find weight and coefficient of friction . . . . .	95
Exa 10.3	To estimate the power . . . . .	95
Exa 10.4	To find work done . . . . .	96
Exa 10.5	To find the torque required . . . . .	97
Exa 10.6	To find force required . . . . .	98
Exa 10.7	To find diameter of hand wheel . . . . .	99
Exa 10.8	To find the power required . . . . .	100
Exa 10.9	To find the force applied . . . . .	101
Exa 10.10	To find ratio of torques and efficiency . . . . .	102
Exa 10.11	To find work done and efficiency . . . . .	103
Exa 10.12	To find length of lever . . . . .	104
Exa 10.13	To find the torque required . . . . .	105
Exa 10.14	To find the force . . . . .	106
Exa 10.15	To find power transmitted . . . . .	107
Exa 10.16	To estimate power lost in friction . . . . .	107
Exa 10.17	To find power absorbed in friction . . . . .	108
Exa 10.18	To find power lost in friction . . . . .	109
Exa 10.19	To find power absorbed in friction . . . . .	110
Exa 10.20	To find power absorbed . . . . .	111
Exa 10.21	To find diameter and number of collars . . . . .	111
Exa 10.22	To find the pressure . . . . .	112
Exa 10.23	To find power transmitted . . . . .	113
Exa 10.24	To find radii and axial thrust . . . . .	114
Exa 10.25	To find dimensions of clutch plate . . . . .	115
Exa 10.26	To find dimensions of clutch plate . . . . .	116



Exa 10.27	To find speed time and KE lost . . . . .	117
Exa 10.28	To find the power transmitted . . . . .	119
Exa 10.29	To find maximum intensity of pressure . . . . .	120
Exa 10.30	To find maximum power transmitted . . . . .	121
Exa 10.31	To find dimensions and axial load . . . . .	122
Exa 10.32	To find axial force and face width . . . . .	123
Exa 10.33	To find dimensions of contact surfaces . . . . .	124
Exa 10.34	To find time required and energy lost . . . . .	125
Exa 10.35	To find mass and size of shoes . . . . .	126
Exa 10.36	To find power transmitted . . . . .	127
Exa 11.1	To find speed of shaft . . . . .	129
Exa 11.2	To find speed lost . . . . .	130
Exa 11.3	To find radii of stepped pulleys . . . . .	130
Exa 11.4	To find the power transmitted . . . . .	132
Exa 11.5	To find force and power . . . . .	133
Exa 11.6	To find length and power transmitted . . . . .	134
Exa 11.7	To find stress in the belt . . . . .	135
Exa 11.8	To find width of the belt . . . . .	136
Exa 11.9	To find width of the belt . . . . .	137
Exa 11.10	To find greatest power transmitted . . . . .	138
Exa 11.11	To find torque power and efficiency . . . . .	139
Exa 11.12	To find power transmitted . . . . .	141
Exa 11.13	To find power transmitted . . . . .	141
Exa 11.14	To find diameter power and tension . . . . .	142
Exa 11.15	To find width tension and length . . . . .	143
Exa 11.16	To find power transmitted . . . . .	145
Exa 11.17	To find power and shaft speed . . . . .	147
Exa 11.18	To find maximum power transmitted . . . . .	148
Exa 11.19	To find number of V belts . . . . .	149
Exa 11.20	To find number of ropes required . . . . .	151
Exa 11.21	To find speed of pulley . . . . .	152
Exa 11.22	To find initial tension and diameter . . . . .	153
Exa 11.23	To find pitch and length of chain . . . . .	154
Exa 12.1	To find total load . . . . .	156
Exa 12.2	To find addendum . . . . .	157
Exa 12.3	To find length of path of contact . . . . .	157
Exa 12.4	To find angle and maximum velocity . . . . .	159
Exa 12.5	To find velocity and angle turned . . . . .	160

Exa 12.6	To find teeth angle and ratio . . . . .	162
Exa 12.7	To find length of path of contact . . . . .	164
Exa 12.8	To find path and arc of contact . . . . .	165
Exa 12.9	To find number of teeth . . . . .	166
Exa 12.10	To find number of teeth . . . . .	167
Exa 12.11	To find addenda and velocity . . . . .	167
Exa 12.12	To find sliding velocities and contact ratio . . . . .	169
Exa 12.13	To find teeth and velocity . . . . .	171
Exa 12.14	To find pressure angle and teeth . . . . .	172
Exa 12.15	To find axial thrust . . . . .	173
Exa 12.16	To find teeth distance and efficiency . . . . .	175
Exa 12.17	To find angle teeth and efficiency . . . . .	176
Exa 13.1	To find speed of gear F . . . . .	178
Exa 13.2	To design the gears . . . . .	178
Exa 13.3	To find the number of teeth . . . . .	179
Exa 13.4	To find speed of gear B . . . . .	181
Exa 13.5	To find speed of gear C . . . . .	182
Exa 13.6	To find speed of gears B and C . . . . .	182
Exa 13.7	To find revolutions of arm . . . . .	183
Exa 13.8	To find teeth and speed . . . . .	184
Exa 13.9	To find number of teeth . . . . .	186
Exa 13.10	To find angular velocities . . . . .	186
Exa 13.11	To find angular velocities . . . . .	187
Exa 13.12	To find speed of shaft . . . . .	188
Exa 13.14	To find number of teeth and speed . . . . .	189
Exa 13.15	To find velocity ratio . . . . .	190
Exa 13.16	To find speed of shaft . . . . .	191
Exa 13.17	To find speed of output shaft . . . . .	192
Exa 13.18	To find speed of road wheel . . . . .	193
Exa 13.19	To find torque exerted . . . . .	194
Exa 13.20	To find teeth and torque . . . . .	195
Exa 13.21	To find speed direction and torque . . . . .	195
Exa 13.22	To find holding torque . . . . .	197
Exa 13.23	To find speed direction and torque . . . . .	198
Exa 13.24	To find torque and forces . . . . .	199
Exa 14.1	To find speed of precession . . . . .	202
Exa 14.2	To find the resultant reaction . . . . .	203
Exa 14.3	To find gyroscopic couple . . . . .	204

Exa 14.4	To find gyroscopic couple . . . . .	204
Exa 14.5	To find gyroscopic couple and direction . . . . .	205
Exa 14.6	To find gyroscopic couple and effect . . . . .	206
Exa 14.7	To find gyroscopic couple and acceleration . . . . .	207
Exa 14.8	To find the gyroscopic effects . . . . .	208
Exa 14.9	To find maximum acceleration during pitching . . . . .	210
Exa 14.10	To find centrifugal and gyroscopic effects . . . . .	211
Exa 14.12	To find load on each wheel . . . . .	212
Exa 14.13	To find pressure on each rail . . . . .	214
Exa 14.14	To find gyroscopic couple and reaction . . . . .	215
Exa 14.15	To find angle of inclination . . . . .	216
Exa 14.16	To find inclination of gyrowheel . . . . .	217
Exa 14.17	To find the gyroscopic couple . . . . .	218
Exa 15.1	To find linear and angular velocity and acceleration . . . . .	220
Exa 15.2	To find linear and angular velocity and acceleration . . . . .	221
Exa 15.3	To find crank angle and velocity . . . . .	223
Exa 15.4	To find velocity and acceleration . . . . .	224
Exa 15.5	To find linear and angular velocity and acceleration . . . . .	225
Exa 15.6	To find inertia force . . . . .	226
Exa 15.7	To find pressure thrust force and moment . . . . .	227
Exa 15.8	To find turning moment . . . . .	228
Exa 15.9	To find load thrust reaction and speed . . . . .	229
Exa 15.10	To find reaction thrust and turning moment . . . . .	231
Exa 15.11	To find force load thrust and speed . . . . .	232
Exa 15.12	To find turning moment thrust and acceleration . . . . .	233
Exa 15.13	To find effort thrust and turning moment . . . . .	235
Exa 15.14	To find turning moment . . . . .	236
Exa 15.15	To find equivalent system . . . . .	238
Exa 15.16	To find equivalent system . . . . .	239
Exa 15.17	To find radius and MI . . . . .	239
Exa 15.18	To find correcting couple . . . . .	241
Exa 15.19	To find acceleration and inertia force . . . . .	242
Exa 15.20	To find torque exerted . . . . .	243
Exa 15.21	To find acceleration and inertia torque . . . . .	244
Exa 15.22	To find resultant force . . . . .	247
Exa 16.1	To find maximum and minimum speeds . . . . .	249
Exa 16.2	To find angular acceleration and KE . . . . .	250
Exa 16.3	To find weight of flywheel . . . . .	251

Exa 16.4	To find coefficient of fluctuation of speed . . . . .	251
Exa 16.5	To find mass of the flywheel . . . . .	252
Exa 16.6	To find power and speed fluctuation . . . . .	253
Exa 16.7	To find coefficient of fluctuation . . . . .	254
Exa 16.8	To find power and coefficients . . . . .	255
Exa 16.9	To find moment of inertia . . . . .	257
Exa 16.10	To find mass of the rim . . . . .	258
Exa 16.12	To find fluctuation of energy and speed . . . . .	259
Exa 16.13	To find power fluctuation and torque . . . . .	260
Exa 16.14	To find diameter and cross section . . . . .	262
Exa 16.15	To find mass and cross section . . . . .	263
Exa 16.16	To find MI and dimensions . . . . .	264
Exa 16.17	To find MI and size . . . . .	266
Exa 16.18	To find diameter and cross section . . . . .	267
Exa 16.19	To find power and mass . . . . .	269
Exa 16.20	To find power and mass . . . . .	270
Exa 16.21	To find speed and number of rivets . . . . .	271
Exa 16.22	To find mass of the flywheel . . . . .	272
Exa 16.23	To find power and cross section . . . . .	273

# Chapter 2

## Kinematics of Motion

Scilab code Exa 2.1 To Find the Acceleration and Distance

```
1 //To Find the Acceleration and Distance
2 clc
3 //Given:
4 u1=0, v1=72*1000/3600 //m/s
5 s1=500 //m
6 //Solution:
7 //Calculating the initial acceleration of the car
8 a1=(v1^2-u1^2)/(2*s1) //m/s^2
9 //Calculating time taken by the car to attain the
   speed
10 t1=(v1-u1)/a1 //seconds
11 //Parameters for the second case
12 u2=v1, v2=90*1000/3600 //m/s
13 t2=10 //seconds
14 //Calculating the acceleration for the second case
15 a2=(v2-u2)/t2 //m/s^2
16 //Calculating the distance moved by the car in the
   second case
17 s2=(u2*t2)+(a2/2*t2^2)
18 //Parameters for the third case
19 u3=v2, v3=0 //m/s
```

```

20 t3=5 //seconds
21 //Calculating the distance moved by the car
22 s3=(u3+v3)*t3/2 //m
23 //Results:
24 printf("\n\n The acceleration of the car , a = %.1f m
    /s ^2. \n",a1)
25 printf(" The car takes t = %d s to attain the speed
    .\n",t1)
26 printf(" The acceleration of the car in the second
    case , a = %.1f m/s ^2.",a2)
27 printf("\n The distance moved by the car , s = %d m.\n",s2)
28 printf(" The distance travelled by the car during
    braking , s = %.1f m.\n\n",s3)

```

---

### Scilab code Exa 2.3 To Find the Velocity

```

1 //To Find the Velocity
2 clc
3 //Given:
4 //Initial parameters
5 v0=100 //kmph
6 t0=0
7 //Parameters at the end of 40 seconds
8 v1=90/100*v0 //kmph
9 t1=40 //seconds
10 //Solution:
11 //The acceleration is given by, a=(-dv/dt)=k*v
12 //Integrating , we get ln(v)=-k*t+C
13 //Calculating the constant of integration
14 C=integrate('1/v','v',1,100)
15 //Calculating the constant of proportionality
16 k=(C-2.3*log10(90))/40
17 //Time after 120 seconds
18 t2=120 //seconds

```

```

19 //Calculating the velocity after 120 seconds
20 v120=10^((-k*t2+C)/2.29)
21 //Results:
22 printf("\n\n The velocity at the end of 120 seconds ,
        v120 = %.1 f kmph.\n\n",v120)

```

---

**Scilab code Exa 2.5** To Find the Maximum Cutting Speed

```

1 //To Find the Maximum Cutting Speed
2 clc
3 //Given:
4 s=500 , s1=125 , s2=250 , s3=125 //mm
5 t=1 //second
6 //Solution:
7 //Matrices for the velocity vs. time graph
8 V=[0,750,750,0] //The velocity matrix
9 T=[0,1/3,2/3,1] //The time matrix
10 plot2d(T,V)
11 //Calculating the time of uniform acceleration
12 t1=rdivf('s1','v/2')
13 //Calculating the time of constant speed
14 t2=rdivf('s2','v')
15 //Calculating the time of uniform retardation
16 t3=rdivf('s3','v/2')
17 //Equating the time taken to complete the stroke to
    1 second
18 v=(125/(1/2)+250/1+125/(1/2))/1 //mm/s
19 //Results:
20 printf("\n\n The maximum cutting speed , v = %d mm/s
        .\n\n",v)

```

---

**Scilab code Exa 2.6** To Find the Angular Acceleration

```

1 //To Find the Angular Acceleration
2 clc
3 //Given:
4 N0=0,N=2000 //rpm
5 t=20 //seconds
6 //Solution:
7 //Calculating the angular velocities
8 omega0=0, omega=2*pi*N/60 //rad/s
9 //Calculating the angular acceleration
10 alpha=(omega-omega0)/t //rad/s^2
11 //Calculating the angular distance moved by the
    wheel during 2000 rpm
12 theta=(omega0+omega)*t/2 //rad
13 //Calculating the number of revolutions made by the
    wheel
14 n=theta/(2*pi)
15 //Results:
16 printf(" \n\n The angular acceleration of the wheel,
    alpha = %.3f rad/s^2.\n",alpha)
17 printf(" The wheel makes n = %.1f revolutions.\n\n",
    n)

```

---

**Scilab code Exa 2.7** To Find Velocity and Acceleration

```

1 //To Find Velocity and Acceleration
2 clc
3 //Given:
4 r=1.5 //m
5 N0=1200,N=1500 //rpm
6 t=5 //seconds
7 //Solution:
8 //Calculating the angular velocities
9 omega0=2*pi*N0/60, omega=2*pi*N/60 //rad/s
10 //Calculating the linear velocity at the beginning
11 v0=r*omega0 //m/s

```



```

12 //Calculating the linear velocity at the end of 5
    seconds
13 v5=r*omega //m/s
14 //Calculating the angular acceleration
15 alpha=(omega-omega0)/t //ad/s^2
16 //Calculating the tangential acceleration after 5
    seconds
17 TangentialAcceleration=alpha*(r/2) //m/s^2
18 //Calculating the radial acceleration after 5
    seconds
19 RadialAcceleration=(omega^2)*(r/2) //m/s^2
20 //Results:
21 printf("\n\n The linear velocity at the beginning ,
    v0 = %.1f m/s.\n",v0)
22 printf(" The linear velocity after 5 seconds , v5 = %
    .1f m/s.\n",v5)
23 printf(" The tangential acceleration after 5 seconds
    is %.1f m/s^2.\n",TangentialAcceleration)
24 printf(" The radial acceleration after 5 seconds is
    %d m/s^2.",RadialAcceleration)

```

---

# Chapter 3

## Kinetics of Motion

Scilab code Exa 3.1 To find the angular acceleration and KE

```
1 //To find the angular acceleration and KE
2 clc
3 //Given:
4 k=1 //m
5 m=2500 //kg
6 T=1500 //N-m
7 //Solution:
8 //Calculating the mass moment of inertia of the
   flywheel
9 I=m*k^2 //kg-m^2
10 //Calculating the angular acceleration of the
   flywheel
11 alpha=T/I //rad/s^2
12 //The angular speed at start
13 omega1=0
14 t=10 //seconds
15 //Calculating the angular speed of the flywheel
   after t=10 seconds from start
16 omega2=omega1+alpha*t //rad/s
17 //Calculating the kinetic energy of the flywheel
   after 10 seconds from start
```

```

18 E=1/2*I*omega2^2/1000 //kJ
19 //Results:
20 printf("\n\n The angular acceleration of the
    flywheel, alpha = %.1f rad/s^2.\n",alpha)
21 printf(" The kinetic energy of the flywheel, E = %d
    kJ.\n\n",E)

```

---

**Scilab code Exa 3.2** To find the time torque and power

```

1 //To find the time, torque and power
2 clc
3 //Given:
4 mC=500, mD=250 //kg
5 s=100, r=0.5, k=0.35 //m
6 m=3 //kg/m
7 //Solution:
8 //Velocities of the cage
9 u1=0, v1=10, v2=10, u3=10, v3=0 //m/s
10 //Accelerations of the cage
11 a1=1.5, a3=-6 //m/s^2
12 s=100 //m
13 //Calculating the time taken by the cage to reach
    the top
14 t1=(v1-u1)/a1 //seconds
15 //Calculating the distance moved by the cage during
    time t1
16 s1=(v1+u1)/2*t1 //m
17 //Calculating the time taken for the cage from
    initial velocity u3=10 m/s to final velocity of
    v3=0
18 t3=(v3-u3)/a3 //seconds
19 //Calculating the distance moved by the cage during
    time t3
20 s3=(v3+u3)/2*t3 //m
21 //Calculating the distance travelled during constant

```

```

    velocity of v2=10 m/s
22 s2=s-s1-s3 //m
23 //Calculating the time taken for the cage during
    constant velocity
24 t2=s2/v2 //seconds
25 //Calculating the time taken for the cage to reach
    the top
26 t=t1+t2+t3 //seconds
27 //Calculating the total mass of the rope for 100
    metres
28 mR=m*s //kg
29 //Calculating the force to raise the cage and rope
    at uniform speed
30 F1=(mC+mR)*9.81 //N
31 //Calculating the torque to raise the cage and rope
    at uniform speed
32 T1=F1*r //N-m
33 //Calculating the force to accelerate the cage and
    rope
34 F2=(mC+mR)*a1 //N
35 //Calculating the torque to accelerate the cage and
    rope
36 T2=F2*r //N-m
37 //Calculating the mass moment of inertia of the drum
38 I=mD*k^2 //kg-m^2
39 //Calculating the angular acceleration of the drum
40 alpha=a1/r //rad/s^2
41 //Calculating the torque to accelerate the drum
42 T3=I*alpha //N-m
43 //Calculating the total torque which must be applied
    to the drum at starting
44 T=T1+T2+T3 //N-m
45 //Calculating the mass of 33.35 m rope
46 m1=m*33.35 //kg
47 //Calculating the reduction of torque
48 T4=(m1*9.81+m1*a1)*r //N-m
49 //Calculating the angular velocity of drum
50 omega=v2/(2*pi*r) //rad/s

```

```

51 //Calculating the power
52 P=T4*omega/1000 //Power, kW
53 //Results:
54 printf("\n\n The time taken for the cage to reach
    the top, t = %.2f s.\n",t)
55 printf(" The total torque which must be applied to
    the drum during starting, T = %.1f N-m.\n",T)
56 printf(" The power required is %.3f kW.\n\n",P)

```

---

**Scilab code Exa 3.3** To find the reduction of speed

```

1 //To find the reduction of speed
2 clc
3 //Given:
4 P=4*1000 //W
5 I=140 //kg-m^2
6 N1=240 //rpm
7 //Solution:
8 //Calculating the angular acceleration at the
    commencement of operation
9 omega1=2*pi*N1/60 //rad/s
10 //Calculating the energy supplied by the motor (E1)
    and the energy consumed in closing a revet in 1
    second
11 E1=4000,E2=10000 //N-m
12 //Calculating the loss of kinetic energy of the
    flywheel during the operation
13 E=E2-E1 //N-m
14 //Calculating the kinetic energy of the flywheel at
    the commencement of operation
15 KEc=1/2*I*omega1^2 //Kinetic energy at the
    commencement, N-m
16 //Calculating the kinetic energy of the flywheel at
    the end of operation
17 KEe=KEc-E //Kinetic energy at the end, N-m

```

```

18 //Calculating the angular speed of the flywheel
    immediately after closing a revet
19 omega2=sqrt(KEe*2/I) //rad/s
20 //Calculating the reduction of speed
21 ReductionofSpeed=(omega1-omega2)*60/(2*pi) //rpm
22 //Calculating the maximum rate at which the revets
    can be closed per minute
23 Rate=P*60/E2 //Maximum rate at which the revets can
    be closed per minute
24 //Results:
25 printf("\n\n The reduction of speed is %.1f rpm.\n",
    ReductionofSpeed)
26 printf(" The maximum rate at which rivets can be
    closed per minute is %d.\n\n",Rate)

```

---

**Scilab code Exa 3.4** To find the torque and power

```

1 //To find the torque and power
2 clc
3 //Given:
4 m=14*1000 ,m1=1.25*1000 ,m2=110 //kg
5 d=1 ,r=d/2 ,k1=450/1000 ,k2=125/1000 //m
6 F=1.2*1000 //N
7 eta=0.85
8 v=1.8 //m/s
9 a=0.1 //m/s^2
10 //Solution:
11 //Calculating the forces opposing the motion
12 P1=m*9.81*1/20+m*a+F //N
13 //Calculating the torque on the drum shaft to
    accelerate the load
14 T1=P1*r //N-m
15 //Calculating the mass moment of inertia of the drum
16 I1=m1*k1^2 //kg-m^2
17 //Calculating the angular acceleration of the drum

```

```

18 alpha1=a/r //rad/s
19 //Calculating the torque on the drum to accelerate
    the drum shaft
20 T2=I1*alpha1 //N-m
21 //Calculating the torque on the armature to
    accelerate drum and load
22 T3=(T1+T2)/(40*eta) //N-m
23 //Calculating the mass moment of inertia of the
    armature
24 I2=m2*k2^2 //kg-m^2
25 //Calculating the angular acceleration of the
    armature
26 alpha2=a/r*40 //rad/s^2
27 //Calculating the torque on the armature to
    accelerate armature shaft
28 T4=I2*alpha2 //N-m
29 //Calculating the torque on the motor shaft
30 T=T3+T4 //N-m
31 //Calculating the angular speed of the motor
32 omega=v/r*40 //rad/s
33 //Calculating the power developed by the motor
34 P=T*omega/1000 //Power developed by the motor, kW
35 //Results:
36 printf("\n\n The torque on the motor shaft , T = %.2 f
    N-m.\n",T)
37 printf(" The power developed by the motor is %.2 f kW
    .\n\n",P)

```

---

**Scilab code Exa 3.5** To find the KE and braking force

```

1 //To find the KE and braking force
2 clc
3 //Given:
4 m=12*1000 , m1=2*1000 , m2=2.5*1000 //kg
5 k1=0.4 , d1=1.2 , r1=d1/2 , k2=0.6 , d2=1.5 , r2=d2/2 , s=6 //m

```

```

6 v=9*1000/3600 //m/s
7 //Solution:
8 //Calculating the mass moment of inertia of the
  front roller
9 I1=m1*k1^2 //kg-m^2
10 //Calculating the mass moment of inertia of the rear
  axle together with its wheels
11 I2=m2*k2^2 //kg-m^2
12 //Calculating the angular speed of the front roller
13 omega1=v/r1 //rad/s
14 //Calculating the angular speed of rear wheels
15 omega2=v/r2 //rad/s
16 //Calculating the kinetic energy of rotation of the
  front roller
17 E1=1/2*I1*omega1^2 //N-m
18 //Calculating the kinetic energy of rotation of the
  rear axle with its wheels
19 E2=1/2*I2*omega2^2 //N-m
20 //Calculating the total kinetic energy of rotation
  of the wheels
21 E=E1+E2 //N-m
22 //Calculating the kinetic energy of translation of
  the road roller
23 E3=1/2*m*v^2 //N-m
24 //Calculating the total kinetic energy of the road
  roller
25 E4=E3+E //N-m
26 //Calculating the braking force to bring the roller
  to rest
27 F=E4/s //N
28 //Results:
29 printf("\n\n The total kinetic energy of rotation of
  the wheels , E = %d N-m.\n",E)
30 printf(" The total kinetic energy of the road roller
  , E4 = %d N-m.\n",E4)
31 printf(" The braking force required to bring the
  roller to rest , F = %.1f N.\n\n",F)

```

---



**Scilab code Exa 3.7** To Find the Speed and Energy Lost

```
1 //To Find the Speed and Energy Lost
2 clc
3 //Given:
4 r=500/1000 ,k=450/1000 //m
5 m1=500 ,m2=1250 //kg
6 u=0.75 //m/s
7 //Solution:
8 //Calculating the mass moment of inertia of drum
9 I2=m2*k^2 //kg-m^2
10 //Calculating the speed of truck
11 //Impulse , F=m1*v or , F-m1*v=0
    ..... ( i )
12 //Moment of impulse , F*r=I2*(omega2-omega2) or , F*r=
    I2*(u-v)/r or , F*r+I2*v/r=I2*u/r ..... ( ii )
13 //Solving (i) and (ii)
14 A=[1 -m1; r I2/r]
15 B=[0; I2*u/r]
16 V=A \ B
17 v=V(2)
18 //Calculating the energy lost to the system
19 E=1/2*I2*(u^2-v^2)/r^2-1/2*m1*v^2 //Energy lost to
    the system , N-m
20 //Results:
21 printf(" \n\n The speed of the truck when the motion
    becomes steady , v = %.3f m/s.\n",v)
22 printf(" The energy lost to the system is %d N-m.\n\
    n" ,E)
```

---

**Scilab code Exa 3.8** To find the velocity KE and compression

```

1 //To find the velocity , KE and compression
2 clc
3 //Given:
4 s=0.7*10^6 //N/m
5 m1=10*10^3 ,m2=15*10^3 //kg
6 v1=1.8 ,v2=0.6 //m/s
7 //Solution:
8 //Calculating the common velocity when moving
   together during impact
9 v=(m1*v1+m2*v2)/(m1+m2)
10 //Calculating the kinetic energy lost to the system
11 E=(1/2*m1*v1^2+1/2*m2*v2^2)-1/2*(m1+m2)*v^2
12 //Calculating the compression of each buffer spring
13 x=sqrt(E/(2*s))
14 //Calculating the velocity of each truck on
   separation
15 //Final KE after separation=KE at common velocity+
   Half of energy stored in springs.
16 //And initial and final momentum must be equal.
17 //Simplifying the two equations , we get ,
18 //1/2*m1*v3^2+1/2*m2*v4^2=1/2*(m1+m2)*v^2+1/2*E
   ..... ( i )
19 //m1*v3+m2*v4=(m1+m2)*v
20 function y=f(x)
21     v3=x(1)
22     v4=x(2)
23     y(1)=1/2*m1*v3^2+1/2*m2*v4^2-1/2*(m1+m2)*v
       ^2-1/2*E
24     y(2)=m1*v3+m2*v4-(m1+m2)*v
25 endfunction
26 z=fsolve([1,1],f)
27 v3=z(1)
28 v4=z(2)
29 //Results:
30 printf(" \n\n The common velocity when moving
   together during impact , v = %.2f m/s.\n",v)
31 printf(" The kinetic energy lost to the system is %
   .2f kN-m.\n",E/1000)

```

```

32 printf(" The compression of each buffer spring , x =
    %d mm.\n",x*1000)
33 printf(" The velocity of separation for 10 tonnes
    truck , v3 = %.1f m/s.\n",v3)
34 printf(" The velocity of separation for 15 tonnes
    truck , v4 = %.1f m/s.\n",v4)

```

---

**Scilab code Exa 3.9** To find energy lost and resistance

```

1 //To find energy lost and resistance
2 clc
3 //Given:
4 m1=300,m2=500 //kg
5 s=1,x=150/1000 //m
6 //Solution:
7 //Calculating the velocity with which mass m1 hits
    the pile
8 u=0
9 v1=sqrt(2*9.81*s+u^2) //m/s
10 //Calculating the common velocity after impact
11 v2=0
12 v=(m1*v1+m2*v2)/(m1+m2) //m/s
13 //Calculating the kinetic energy before impact
14 KEb=m1*9.81*s //Kinetic energy before impact, N-m
15 //Calculating the kinetic energy after impact
16 KEa=1/2*(m1+m2)*v^2 //Kinetic energy after impact, N
    -m
17 //Calculating the energy lost in the blow
18 E=KEb-KEa //Energy lost in the blow, N-m
19 //Calculating the average resistance against the
    pile
20 R=KEa/x+m1*9.81+m2*9.81
21 //Results:
22 printf("\n\n The energy lost in the blow is %d N-m.\n
    n",E)

```

```

23 printf(" The average resistance against the pile , R
    = %.3 f kN.\n\n",R/1000)

```

---

**Scilab code Exa 3.10** To find the angular velocities

```

1 //To find the angular velocities
2 clc
3 //Given:
4 m1=0.7 ,m2=2.4 //kg
5 k1=270/1000 ,k2=185/1000 ,h1=0.25 ,DL=0.2 ,CM=0.275 //m
6 //Solution:
7 //Calculating the angular velocity of hammer just
  before impact
8 h=h1*(1-cos(20*%pi/180))
9 omega=sqrt(m1*9.81*h*2/(m1*k1^2)) //rad/s
10 //Calculating the relative linear velocity
11 RLV=0.8*omega*CM
12 //Calculating the values of angular velocities
13 //The two equations we get in terms of omegaA and
  omegaB are
14 //DL*omegaA-CM*omegaB=RLV
  .....(
  i)
15 //m1*k1^2*(omega-omegaB)=.275/.2*m2*k2^2*omegaA , or
16 //2.21*omegaA+omegaB=2.01
  .....(
  ii)
17 A=[DL -CM; 2.21 1]
18 B=[RLV; 2.01]
19 V=A \ B
20 //Results:
21 printf(" \n\n The angular velocity of the anvil A,
  omegaA = %.2 f rad/s.\n",V(1))
22 printf(" The angular velocity of the hammer B,
  omegaB = %.2 f rad/s, i.e. %.2 f rad/s, in the

```

reverse direction.\n\n",V(2),V(2)\*-1)

---

**Scilab code Exa 3.11** To find the velocity impulse angle of swing and average force

```
1 //To find the velocity , impulse , angle of swing and
  average force
2 clc
3 //Given:
4 m=30 //kg
5 AG=1,GB=150/1000,k1=1.1,k2=350/1000 //m
6 theta=60*%pi/180 //rad
7 t=0.005 //s
8 a=AG,b=GB
9 //Solution:
10 //Calculating the mass moment of inertia of the
  pendulum about the point of suspension A
11 IA=m*k1^2 //kg-m^2
12 //Calculating the mass moment of inertia of the
  pendulum about centre of gravity G
13 IG=m*k2^2 //kg-m^2
14 //Calculating the angular velocity of the pendulum
15 h1=a-a*cos(theta)
16 omega=sqrt(2*m*9.81*h1/IA) //rad/s
17 //Calculating the striking velocity of the pendulum
18 v=omega*(a+b) //m/s
19 //Calculating the angular velocity of the pendulum
  just after the breakage of the specimen
20 omega1=sqrt(omega^2-2*54/IA)
21 //Calculating the linear velocity of G just before
  the breakage of specimen
22 vG=omega*AG //m/s
23 //Calculating the linear velocity of G just after
  the breakage of specimen
24 vGdash=omega1*AG //m/s
```

```

25 //Calculating the impulses at pivot A and knife edge
    B
26 //F1+F2=m*(vG-vGdash)
    .....( i)
27 //b*F2-a*F1=IG*(omega-omega1)
    .....( ii)
28 A=[1 1; -a b]
29 B=[m*(vG-vGdash); IG*(omega-omega1)]
30 V=A \ B
31 F1=V(1),F2=V(2)
32 //Calculating the angle of swing of the pendulum
    after impact
33 theta1=acos(a-1/2*IA*omega1^2/(m*9.81))/a //radians
34 //Calculating the average force exerted at the pivot
35 Fp=F1/t //N
36 //Calculating the average force exerted at the knife
    edge
37 Fk=F2/t //N
38 //Results:
39 printf("\n\n The striking velocity of the pendulum ,
    v = %.2f m/s.\n",v)
40 printf(" Impulse at the pivot A, F1 = %.1f N.\n",F1)
41 printf(" Impulse at the knife edge B, F2 = %.1f N.\n
    ",F2)
42 printf(" Angle of swing of the pendulum after impact
    , theta = %.2f degree.\n",theta1*180/%pi)
43 printf(" Average force exerted at the pivot is %d N
    .\n",Fp)
44 printf(" Average force exerted at the knife edge is
    %d N.\n\n",Fk)

```

---

**Scilab code Exa 3.12** To find the speed time and KE lost

```

1 //To find the speed , time and KE lost
2 clc

```

```

3 //Given :
4 T=150 //N-m
5 m1=60 ,m2=20 //kg
6 k1=140/1000 ,k2=80/1000 //m
7 N1=750 ,N2=0 //rpm
8 //Sloution :
9 //Calculating the angular speeds
10 omega1=2*%pi*N1/60 ,omega2=0 //rad/s
11 //Calculating the mass moment of inertia of the
    rotor on motor
12 I1=m1*k1^2 //kg-m^2
13 //Calculating the mass moment of inertia of the
    parts attached to machine
14 I2=m2*k2^2 //kg-m^2
15 //Calculating the speed after engagement of the
    clutch and the time taken
16 //We know that impulsive torque = change in angular
    momentum
17 //T*t = I1*(omega1-omega) , or I1*omega+T*t = I1*
    omega1 ..... ( i)
18 //T*t = I2*(omega-omega2) , or I2*omega-T*t = I2*
    omega2 ..... ( ii)
19 A=[I1 T; I2 -T]
20 B=[I1*omega1; I2*omega2]
21 V=A \ B
22 omega=V(1) //rad/s
23 t=V(2) //s
24 //Calculating the kinetic energy lost during the
    operation
25 E=I1*I2*(omega1-omega2)^2/(2*(I1+I2)) //N-m
26 //Results :
27 printf("\n\n The speed after engagement, omega = %.1
    f rad/s.\n",omega)
28 printf(" The time taken , t = %.2f s.\n",t)
29 printf(" The kinetic energy lost during the
    operation , E = %d N-m.\n\n",E)

```

---

**Scilab code Exa 3.13** To find the acceleration

```
1 //To find the acceleration
2 clc
3 //Given:
4 M=75 //kg
5 r=0.3 //m
6 G=6
7 IA=100,IB=5 //kg-m2
8 eta=90/100 //%
9 //Solution:
10 //Calculating the equivalent mass of the geared
    system
11 me=1/r2*(IA+G2*IB) //kg
12 //Calculating the total equivalent mass to be
    accelerated
13 Me=me+M //kg
14 //Calculating the acceleration when it is allowed to
    fall freely
15 F=M*9.81 //Accelerating force provided by the pull
    of gravity , N
16 a=F/Me //m/s2
17 //Calculating the equivalent mass of the geared
    system when the efficiency is 90%
18 me1=1/r2*(IA+G2*IB/eta) //kg
19 //Calculating the total equivalent mass to be
    accelerated
20 Me1=me1+M //kg
21 //Calculating the acceleration when the efficiency
    is 90%
22 F1=M*9.81 //Accelerating force by the pull of
    gravity , N
23 a1=F1/Me1 //m/s2
24 //Results:
```



```

25 printf("\n\n The acceleration of the mass M if it is
    allowed to fall freely , a = %.3f m/s ^2.\n",a)
26 printf(" The acceleration of the mass M when the
    efficiency of the gearing system is 0.9, a = %.3f
    m/s ^2.\n\n",a1)

```

---

**Scilab code Exa 3.18** To find the torque

```

1 //To find the torque
2 clc
3 //Given:
4 d=1.5 , r=d/2 , d1=1 , kM=90/1000 , kI=225/1000 , kD=600/1000 ,
    kP=450/1000 //m
5 NM=900 , N1=275 , ND=50 //rpm
6 mM=200 , mI=375 , mD=2250 , mP=200 , m1=1150 , m2=650 //kg
7 FI=150 , FD=1125 , FP=150 //N-m
8 F1=500 , F2=350 //N
9 a=0.9 //m/s ^2
10 //Solution:
11 //Calculating the speed of guide pulley
12 NP=ND*d/d1 //rpm
13 //Calculating the gear ratio for intermediate gear
    and motor
14 G1=N1/NM
15 //Calculating the gear ratio for drum and motor
16 G2=ND/NM
17 //Calculating the gear ratio for the guide pulley
    and motor
18 G3=NP/NM
19 //Calculating the mass moment of inertia of the
    motor
20 IM=mM*kM^2 //kg-m^2
21 //Calculating the mass moment of inertia of the
    intermediate gear
22 II=mI*kI^2 //kg-m^2

```

```

23 //Calculating the mass moment of inertia of the drum
    and shaft
24 ID=mD*kD^2 //kg-m^2
25 //Calculating the mass moment of inertia of the
    guide pulley
26 IP=mP*kP^2 //kg-m^2
27 //Calculating the angular acceleration of the drum
28 alphaD=a/r //rad/s^2
29 //Calculating the angular acceleration of the motor
30 alphaM=alphaD*NM/ND //rad/s^2
31 //Calculating the equivalent mass moment of inertia
    of the system
32 I=IM+G1^2*II+G2^2*ID+2*G3^2*IP //kg-m^2
33 //Calculating the torque at motor to accelerate the
    system
34 T1=I*alphaM //N-m
35 //Calculating the torque at motor to overcome
    friction at intermediate gear, drum and two guide
    pulleys
36 T2=G1*FI+G2*FD+2*G3*FP //N-m
37 //Calculating the tension in the rising rope between
    the pulley and drum
38 Q1=m1*9.81+m1*a+F1 //N
39 //Calculating the tension in the falling rope
    between the pulley and drum
40 Q2=m2*9.81-m2*a-F2 //N
41 //Calculating the torque at drum
42 TD=(Q1-Q2)*r //N-m
43 //Calculating the torque at motor to raise and lower
    cages and ropes and to overcome frictional
    resistance
44 T3=G2*TD //N-m
45 //Calculating the total motor torque required
46 T=T1+T2+T3 //N-m
47 //Results:
48 printf("\n\n The total motor torque required , T = %
    .1 f N-m.\n\n",T)

```

---

**Scilab code Exa 3.19** To find velocities and loss of KE

```
1 //To find velocities and loss of KE
2 clc
3 //Given:
4 m1=50 , m2=25 //kg
5 u1=3 , u2=1.5 //m/s
6 //Solution:
7 //When the impact is inelastic
8 //Calculating the common velocity after impact
9 v=(m1*u1+m2*u2)/(m1+m2) //m/s
10 //Calculating the loss of kinetic energy during
    impact
11 EL=m1*m2/(2*(m1+m2))*(u1-u2)^2 //N-m
12 //When the impact is elastic
13 //Calculating the velocity of the first sphere
    immediately after impact
14 v1=2*v-u1 //m/s
15 //Calculating the velocity of the second sphere
    immediately after impact
16 v2=2*v-u2 //m/s
17 //Calculating the loss of kinetic energy
18 EL1=0
19 //When the coefficient of restitution , e=0.6
20 e=0.6
21 //Calculating the velocity of the first sphere
    immediately after impact
22 v12=(1+e)*v-e*u1 //m/s
23 //Calculating the velocity of the second sphere
    immediately after impact
24 v22=(1+e)*v-e*u2 //m/s
25 //Calculating the loss of kinetic energy
26 EL2=m1*m2/(2*(m1+m2))*(u1-u2)^2*(1-e^2) //N-m
27 //Results :
```

```

28 printf("\n\n The common velocity after impact when
    the impact is inelastic , v = %.1f m/s.\n",v)
29 printf(" The loss of kinetic energy during impact ,
    EL = %.2f N-m.\n",EL)
30 printf(" The velocity of the first sphere
    immediately after impact when the impact is
    elastic , v1 = %d m/s.\n",v1)
31 printf(" The velocity of the second sphere
    immediately after impact , v2 = %.1f m/s.\n",v2)
32 printf(" The loss of kinetic energy , EL = %d.\n",EL1
    )
33 printf(" The velocity of the first sphere
    immediately after impact When the coefficient of
    restitution is 0.6 , v1 = %.1f m/s.\n",v12)
34 printf(" The velocity of the second sphere
    immediately after impact , v2 = %.1f m/s.\n",v22)
35 printf(" The loss of kinetic energy during impactm
    EL = %d N-m.\n\n",EL2)

```

---

**Scilab code Exa 3.20** To find the speed and energy dissipated

```

1 //To find the speed and energy dissipated
2 clc
3 //Given:
4 m1=15*1000 ,m2=5*1000 //kg
5 u1=20*1000/3600 ,u2=12*1000/3600 //m/s
6 s=1000*10^3 //N/m
7 e=0.5
8 //Solution:
9 //Calculating the common speed
10 v=(m1*u1+m2*u2)/(m1+m2) //m/s
11 //Calculating the strain energy stored in one spring
12 SE=mulhf('1/2*s ', 'x^2 ') //Strain energy , N-m
13 //Calculating the strain energy stored in four
    buffer springs

```

```

14 SE4=mulf('4*1/2*s','x^2') //Strain energy, N-m
15 //Calculating the difference in kinetic energies
    before impact and during impact
16 d=m1*m2/(2*(m1+m2))*(u1-u2)^2 //Difference in
    kinetic energies, N-m
17 //Equating the difference between kinetic energies
    to the strain energy stored in the springs
18 x=sqrt(d*2/(4*s))*1000 //mm
19 //Calculating the speed of the loaded wagon
    immediately after impact ends
20 v11=2*v-u1 //m/s
21 //Calculating the speed of the empty wagon
    immediately after impact ends
22 v21=2*v-u2 //m/s
23 //Calculating the speeds of the wagons taking into
    account the coefficient of restitution, e=0.5
24 v12=(1+e)*v-e*u1 //m/s
25 v22=(1+e)*v-e*u2 //m/s
26 //Calculating the amount of energy dissipated during
    impact
27 EL=m1*m2/(2*(m1+m2))*(u1-u2)^2*(1-e^2) //N-m
28 //Results:
29 printf("\\n\\n The magnitude of common speed, v = %d m
    /s.\\n",v)
30 printf(" The maximum deflection of each buffer
    spring during impact, x = %d mm.\\n",x)
31 printf(" The speed of the loaded wagon immediately
    after the impact ends, v1 = %.2f m/s.\\n",v11)
32 printf(" The speed of the empty wagon immediately
    after the impact ends, v2 = %.2f m/s.\\n",v21)
33 printf(" When coefficient of restitution is taken
    into account, v1 = %.3f m/s.\\n",v12)
34 printf(" When coefficient of restitution is taken
    into account, v2 = %.3f m/s.\\n",v22)
35 printf(" The amount of energy dissipated during
    impact, EL = %d N-m.\\n\\n",EL)

```

---

Scilab code Exa 3.21 To find strain energy twist and speed

```
1 //To find strain energy , twist and speed
2 clc
3 //Given:
4 IA=22.5,IB=67.5 //kg-m^2
5 q=225 //N-m/rad
6 NA=150,NB=0 //rpm
7 //Calculating the angular speed of the flywheel
8 omegaA=2*pi*NA/60 //rad/s
9 //Calculating the angular speed of both the
   flywheels at the instant their speeds are equal
10 omega=IA*omegaA/(IA+IB) //rad/s
11 //Calculating the kinetic energy of the system at
   that instant
12 E2=1/2*(IA+IB)*omega^2 //N-m
13 //Calculating the kinetic energy of the flywheel A
14 E1=1/2*IA*omegaA^2 //N-m
15 //Calculating the strain energy stored in the spring
16 E=E1-E2 //Strain energy stored in the spring , N-m
17 //Calculating the maximum twist of the spring
18 theta=sqrt(E*2/q) //radians
19 thetad=theta*180/pi //Maximum twist , degrees
20 //Calculating the speed of each flywheel when the
   spring regains its initial unstrained condition
21 N=60*omega/(2*pi)
22 NA1=2*N-NA //rpm
23 NB1=2*N-NB //rpm
24 //Results:
25 printf("\n\n The strain energy stored in the spring
   is %d N-m.\n",E)
26 printf(" The maximum twist of the spring , theta = %
   .1f degrees.\n",thetad)
27 printf(" The speed of flywheel A when the spring
```

```
    regains its initial unstrained condition , NA1 =  
    %d rpm, i.e. %d rpm in the opposite direction.\n"  
    ,NA1,-NA1)  
28 printf(" The speed of flywheel B when the spring  
    regains its initial unstrained condition , NB1 =  
    %d rpm.\n",NB1)
```

---

# Chapter 4

## Simple Harmonic Motion

Scilab code Exa 4.1 To Find the Velocity and Acceleration

```
1 //To Find the Velocity and Acceleration
2 clc
3 //Given:
4 N=120 //rpm
5 r=1,x=0.75 //m
6 //Solution:
7 //Calculating Angular Velocity
8 omega=2*%pi*N/60 //rad/s
9 //Calculating Velocity of the Piston
10 v=omega*sqrt(r^2-x^2) //m/s
11 //Calculating Acceleration of the Piston
12 a=omega^2*x
13 //Results:
14 printf("\\n\\n The Velocity of the Piston , v = %.2 f m/
    s.\\n",v)
15 printf(" The Acceleration of the Piston , a = %.2 f m/
    s ^ 2.\\n\\n",a)
```

---

Scilab code Exa 4.2 To Find the Angular Velocity Time and Acceleration



```

1 //To Find the Angular Velocity , Time and
  Acceleration
2 clc
3 //Given:
4 x1=.75 , x2=2 //m
5 v1=11 , v2=3 //m/s
6 //Solution:
7 //We have ,  $11 = \omega \cdot \sqrt{r^2 - .75^2}$  and  $3 = \omega \cdot \sqrt{r^2 - 2^2}$ .
8 //These upon solving yield  $r^2 - (121/\omega^2) - 0.5625 = 0$  and  $r^2 - (9/\omega^2) - 4 = 0$ .
9 //Take  $r^2 = x$  and  $(1/\omega^2) = y$  and the equation
  become  $x - 121y = 0.5625$  and  $x - 9y = 4$ .
10 //Variables Matrix
11 A=[1 -121; 1 -9]
12 //Constants Matrix
13 B=[.5625; 4]
14 V=A \ B
15 //Calculating Amplitude of the Particle
16 r=sqrt(V(1)) //m
17 //Calculating Angular Velocity of the Particle
18 omega=sqrt(1/V(2)) //rad/s
19 //Calculating Periodic Time
20 tp=2*%pi/omega //seconds
21 //Calculating Maximum Acceleration
22 amax=omega^2*r //m/s^2
23 //Results:
24 printf(" \n\n The Angular Velocity , omega = %.1f rad/
  s.\n" , omega)
25 printf(" The Periodic Time , tp = %.1f s.\n" , tp)
26 printf(" The Maximum Acceleration , amax = %.2f m/s
  ^2.\n\n" , amax)

```

---

**Scilab code Exa 4.3** To Find the Frequency and Velocity

```

1 //To Find the Frequency and Velocity
2 clc
3 //Given:
4 m=60 //kg
5 r=0.0125,x=0.005 //m
6 //Solution:
7 //Calculating the Extension of the Spring
8 delta=(.25/1.5)*60*10^-3 //m
9 //Calculating the Frequency of the System
10 n=1/(2*pi)*sqrt(9.81/delta) //Hz
11 //Calculating the Angular Velocity of the Mass
12 omega=**sqrt(9.81/delta) //rad/s
13 //Calculating the Linear Velocity of the Mass
14 v=omega*sqrt(r^2-x^2)
15 //Results:
16 printf("\\n\\n The Frequency of Natural Vibration , n =
    %.2f Hz.\\n",n)
17 printf(" The Velocity of the Mass , v = %.2f m/s.\\n\\n
    ",v)

```

---

#### Scilab code Exa 4.4 To Find the Frequency of Oscillation

```

1 //To Find the Frequency of Oscillation
2 clc
3 //Given:
4 m=1,m1=2.5 //kg
5 s=1.8*10^3 //N/m
6 l=(300+300)*10^-3 //m
7 //Solution:
8 //Calculating the Mass Moment of Inertia of the
    System
9 IA=(m*l^2/3)+(m1*l^2) //kg-m^2
10 //Calculating the Ratio of Alpha to Theta
11 //delta=0.3*theta
12 //Restoring Force=s*delta=540*theta

```

```

13 //Restoring torque about A=540*theta*0.3=162*theta N
    -m ... ( i)
14 //Torque about A= IA*alpha=1.02*alpha N-m
    ... ( ii)
15 //Equating (i) and (ii), 1.02*alpha=162*theta
16 alphabytheta=162/1.02
17 //Calculating the Frequency of Oscillation
18 n=1/(2*pi)*sqrt(alphabytheta)
19 //Results:
20 printf("\n\n The Frequency of Oscillation , n = %.2f
    Hz.\n\n",n)

```

---

#### Scilab code Exa 4.5 To Find the Moment of Inertia

```

1 //To Find the Moment of Inertia
2 clc
3 //Given:
4 m=85 //kg
5 h=0.1 //m
6 //Solution:
7 //Calculating the Frequency of Oscillation
8 n=100/145 //Hz
9 //Calculating the Equivalent Length of Simple
    Pendulum
10 L=(1/(2*pi)/.69*sqrt(9.81))^2
11 //Calculating the Radius of Gyration
12 kG=sqrt((L-h)*h)
13 //Calculating the Moment of Inertia of the Flywheel
    through the Centre of Gravity
14 I=m*kG^2 //kg-m^2
15 //Results:
16 printf("\n\n The Moment of Inertia of the Flywheel
    Through its c.g., I = %.1f kg-m^2.\n\n",I)

```

---

**Scilab code Exa 4.6** To Find the Moment of Inertia

```
1 //To Find the Moment of Inertia
2 clc
3 //Given:
4 m=60 //kg
5 d1=75 , d2=102 //mm
6 //Solution:
7 //Calculating the Frequencies of Oscillation
8 n1=100/190 , n2=100/165 //Hz
9 //Calculating the Equivalent Lengths of Simple
  Pendulum
10 L1=9.81/(2*%pi*n1)^2 //m
11 L2=9.81/(2*%pi*n2)^2 //m
12 //Calculating Distance of c.g. from the Small and
  Big End Centres (h1 and h2), and the Radius of
  Gyration
13 function y=f(x)
14     h1=x(1)
15     h2=x(2)
16     kG=x(3)
17     y(1)=L1*h1-h1^2-kG^2
18     y(2)=L2*h2-h2^2-kG^2
19     y(3)=h1+h2-1
20 endfunction
21 z=fsolve([1,1,1],f)
22 h1=z(1) , h2=z(2) , kG=z(3)
23 //Calculating the Mass Moment of Inertia of the Rod
24 I=m*kG^2 //kg-m^2
25 //Results:
26 printf(" \n\n The Moment of Inertia of the Rod, I =
  %d kg-m^2.\n" ,I)
27 printf(" The C.G is at a Distance of h1 = %.3f m
  from the Small End Centre.\n\n" ,h1)
```

---

### Scilab code Exa 4.7 To Find the Time of Swing

```
1 //To Find the Time of Swing
2 clc
3 //Given:
4 l=1.2 //m
5 theta=3*%pi/180 //rad
6 //Solution:
7 //Calculating the Distance Between the Knife Edge
   and C.G. of the Rod
8 h=1.2/2-.05 //m
9 //Calculating the Radius of Gyration of the Rod
   About C.G.
10 kG=1/sqrt(12) //m
11 //Calculating the Time of Swing of the Rod
12 tp=2*%pi*sqrt((kG^2+h^2)/(9.81*h)) //seconds
13 //Calculating the Minimum Time of Swing
14 tpmin=2*%pi*sqrt((2*kG)/9.81) //seconds
15 //Calculating Angular Velocity
16 omega=2*%pi/tp //rad/s
17 //Calculating Maximum Angular Velocity
18 omegamax=omega*theta //rad/s
19 //Calculating Maximum Angular Acceleration
20 alphamax=omega^2*theta //rad/s^2
21 //Results:
22 printf("\\n\\n The Time of Swing of the Rod, tp = %.2 f
   seconds.\\n",tp)
23 printf(" The Minimum Time of Swing, tp(min) = %.2 f
   seconds.\\n",tpmin)
24 printf(" The Maximum Angular Velocity, omega(max) =
   %.4 f rad/s.\\n",omegamax)
25 printf(" The Maximum Angular Acceleration, alpha(max)
   ) = %.3 f rad/s^2.\\n\\n", alphamax)
```

---

**Scilab code Exa 4.8** To Find Centre of Percussion and Impulse

```
1 //To Find Centre of Percussion and Impulse
2 clc
3 //Given:
4 m=30 //kg
5 OG=1.05 ,h=OG ,AG=0.15 //m
6 //Solution:
7 //Calculating the Frequency of Oscillation
8 n=20/43.5 //Hz
9 //Calculating the Equivalent Length of Simple
  Pendulum
10 L=9.81/(2*%pi*n)^2 //m
11 //Calculating the Distance of Centre of Percussion (
  C) from the Centre of Gravity (G)
12 CG=L-OG //m
13 //Calculating the Distance of Centre of Percussion (
  C) from the Knife Edge A
14 AC=AG-CG //m
15 //Calculating the Radius of Gyration of the Pendulum
  About O
16 k0=sqrt(L*h) //m
17 h1=h*(1-cos(60*%pi/180)) //m
18 //Calculating the Angular Velocity of the Pendulum
19 omega=sqrt(2*m*9.81*h1/(m*k0^2)) //rad/s
20 OA=OG+AG
21 //Calculating the Velocity of Striking
22 v=omega*(OA) //Velocity of Striking
23 //Calculating the Angular Velocity of the Pendulum
  Immediately After Impact
24 I=m*k0^2
25 LKE=55 //Loss of Kinetic Energy , N-m
26 omega1=sqrt(omega^2-LKE*2/I)
27 //Calculating the Impulses at Knife Edge A and at
```

```

    Pivot O (P and Q)
28 CLM=m*h*(omega-omega1) //Change of Linear Momentum
29 CAM=m*(k0^2-h^2)*(omega-omega1) //Change of Angular
    Momentum
30 //P+Q=Change of Linear Momentum and, 0.15P-1.05Q=
    Change of Angular Momentum.
31 //i.e., P+Q=CLM and 0.15P-1.05Q=CAM
32 //Variables Matrix
33 A=[1 1; 0.15 -1.05]
34 B=[CLM; CAM]
35 V=A \ B
36 P=V(1)
37 Q=V(2)
38 //Calculating the Change in Axis Reaction When the
    Pendulum is Vertical
39 CAR=m*(omega^2-omega1^2)*h //Change in Axis Reaction
    , N
40 //Results:
41 printf("\n\n The Distance of Centre of Percussion ,
    AC = %.3f m.\n",AC)
42 printf(" The Velocity of Striking = %.2f m/s.\n",v)
43 printf(" The Impulse at the Knife Edge, P = %.1f N-s
    .\n",P)
44 printf(" The Impulse at the Pivot , Q = %.2f N-s.\n",
    Q)
45 printf(" The Change in Axis Reaction When the
    Pendulum is Vertical = %d N.\n\n",CAR)

```

---

**Scilab code Exa 4.9** To Find the Radius of Gyration

```

1 //To Find the Radius of Gyration
2 clc
3 //Given:
4 m=1.5 //kg
5 l=1.25 , x=120*10^-3 , y=x //m

```

```

6 //Solution:
7 //Calculating the Frequency of Oscillation
8 n=20/40 //Hz
9 //Calculating the Radius of Gyration of the
   Connecting Rod
10 kG=1/(2*pi*n)*sqrt(9.81*x*y/l) //m
11 //Calculating the Moment of Inertia of the
   Connecting Rod
12 I=m*kG^2 //kg-m^2
13 //Results:
14 printf("\n\n The Radius of Gyration , kG = %d mm.\n",
   kG*1000)
15 printf(" The Mass Moment of Inertia , I = %.3f kg-m
   ^2.\n\n",I)

```

---

**Scilab code Exa 4.10** To Find the Radius of Gyration

```

1 //To Find the Radius of Gyration
2 clc
3 //Given:
4 l=2.5 , r=250*10^-3 //m
5 //Solution:
6 //Calculating the Frequency of Oscillation
7 n=50/170 //Hz
8 //Calculating the Radius of Gyration of the Wheel
9 kG=r/(2*pi*n)*sqrt(9.81/l) //m
10 //Results:
11 printf("\n\n The Radius of Gyration , kG = %d mm.\n\n",
   kG*10^3)

```

---

**Scilab code Exa 4.11** To Find the Mass Moment of Inertia

```

1 //To Find the Mass Moment of Inertia

```



```

2  clc
3  //Given:
4  m1=5.5,m2=1.5 //kg
5  l=1.25,r=125*10^-3 //m
6  //Solution:
7  //Calculating the Frequency of Oscillation
8  n=10/30 //Hz
9  //Calculating the Radius of Gyration About an Axis
   Through the c.g.
10 kG=r/(2*%pi*n)*sqrt(9.81/l) //m
11 //Calculating the Mass Moment of Inertia About an
   Axis Through its c.g.
12 m=m1+m2 //Total Mass, kg
13 I=m*kG^2 //kg-m^2
14 //Results:
15 printf("\\n\\n The Mass Moment of Inertia About an
   Axis Through its c.g., I = %.3f kg-m^2.\\n\\n",I)

```

---

# Chapter 5

## Simple Mechanisms

Scilab code Exa 5.1 To find the time ratio

```
1 //To find the time ratio
2 clc
3 //Given:
4 AC=300, CB1=120 //mm
5 //Solution:
6 //Refer Fig. 5.28
7 //Calculating the sine of inclination of the slotted
   bar with the vertical
8 sineCAB1=CB1/AC
9 //Calculating the inclination of the slotted bar
   with the vertical
10 angleCAB1=asin(sineCAB1)*180/%pi //degrees
11 //Calculating the angle alpha
12 alpha=2*(90-angleCAB1) //degrees
13 //Calculating the ratio of time of cutting stroke to
   time of return stroke
14 r=(360-alpha)/alpha //Ratio of time of cutting
   stroke to time of return stroke
15 //Results:
16 printf("\\n\\n The ratio of the time of cutting stroke
   to the time of return stroke is %.2f.\\n\\n",r)
```

---

**Scilab code Exa 5.2** To find the time ratio

```
1 //To find the time ratio
2 clc
3 //Given:
4 AC=240 , CB1=120 , AP1=450 //mm
5 //Solution:
6 //Refer Fig. 5.29
7 //Calculating the sine of inclination of the slotted
   bar with the vertical
8 sineCAB1=CB1/AC
9 //Calculating the inclination of the slotted bar
   with the vertical
10 angleCAB1=asin(sineCAB1)*180/%pi //degrees
11 //Calculating the angle alpha
12 alpha=2*(90-angleCAB1) //degrees
13 //Calculating the time ratio of cutting stroke to
   the return stroke
14 r=(360-alpha)/alpha //Time ratio of cutting stroke
   to the return stroke
15 //Calculating the length of the stroke
16 R1R2=2*AP1*sin(%pi/2-alpha/2*%pi/180) //mm
17 //Results:
18 printf("\\n\\n The time ratio of cutting stroke to the
   return stroke is %d.\\n",r)
19 printf(" The length of the stroke , R1R2 = P1P2 = %d
   mm.\\n\\n" ,R1R2)
```

---

**Scilab code Exa 5.3** To find the dimensions of AC and AP

```
1 //To find the dimensions of AC and AP
```

```

2  clc
3  //Given:
4  //Refer Fig. 5.30 and Fig. 5.31
5  BC=30 ,R1R2=120 //mm
6  r=1.7 //Time ratio of working stroke to the return
      stroke
7  //Solution:
8  //Calculating the angle alpha
9  alpha=360/(1.7+1) //degrees
10 //Calculating the length of the link AC
11 B1C=BC
12 AC=B1C/cosd(alpha/2) //mm
13 //Calculating the length of the link AP
14 AP1=R1R2/(2*cosd(alpha/2)) //mm
15 AP=AP1
16 //Results:
17 printf(" \n\n The length of AC = %.1f mm.\n",AC)
18 printf(" The length of AP = %.1f mm.\n\n",AP)

```

---

**Scilab code Exa 5.4** To find the time ratio

```

1  //To find the time ratio
2  clc
3  //Given:
4  CD=50 ,CA=75 ,PA=150 ,PR=135 //mm
5  //Solution:
6  //Refer Fig. 5.32 and Fig. 5.33
7  //Calculating the cosine of angle beta
8  CA2=CA
9  cosbeta=CD/CA2
10 //Calculating the angle beta
11 beta=2*acos(cosbeta)*180/%pi //degrees
12 //Calculating the ratio of time of cutting stroke to
      time of return stroke
13 r=(360-beta)/beta //Ratio of time of cutting stroke

```

```
        to time of return stroke
14 //Calculating the length of effective stroke
15 R1R2=87.5 //mm
16 //Results:
17 printf("\n\n The ratio of time of cutting stroke to
        time of return stroke is %.3f.\n",r)
18 printf(" The length of effective stroke , R1R2 = %.1 f
        mm.\n\n",R1R2)
```

---

# Chapter 6

## Velocity in Mechanisms Instantaneous Centre Method

Scilab code Exa 6.1 To find the angular velocity

```
1 //To find the angular velocity
2 clc
3 //Given:
4 NAB=100 //rpm
5 AB=300/1000,BC=360/1000,CD=BC //m
6 //Solution:
7 //Refer Fig. 6.9
8 //Calculating the angular speed of link AB
9 omegaAB=2*%pi*NAB/60 //rad/s
10 //Calculating the velocity of point B on link AB
11 vB=omegaAB*AB //m/s
12 //Calculating the angular velocity of link BC
13 //By measurement from instantaneous centre diagram,
    Fig. 6.10,
14 I13B=500/1000 //m
15 omegaBC=vB/I13B //rad/s
16 //Results:
17 printf("\\n\\n The angular velocity of the link BC,
    omegaBC = %.3f rad/s.\\n\\n",omegaBC)
```

---

**Scilab code Exa 6.2** To find velocity and angular velocity

```
1 //To find velocity and angular velocity
2 clc
3 //Given:
4 omegaOB=10 //rad/s
5 OB=100/1000 //m
6 //Solution:
7 //Refer Fig. 6.12
8 //Calculating the velocity of the crank OB
9 vOB=omegaOB*OB //m/s
10 vB=vOB
11 //By measurement from the instantaneous centre
    diagram, Fig. 6.13,
12 I13A=460/1000, I13B=560/1000 //m
13 //Calculating the velocity of slider A
14 vA=vB/I13B*I13A
15 //Calculating the angular velocity of the connecting
    rod AB
16 omegaAB=vB/I13B //rad/s
17 //Results:
18 printf("\\n\\n The velocity of slider A, vA = %.2f m/s
    .\\n",vA)
19 printf(" The angular velocity of connecting rod AB,
    omegaAB = %.2f rad/s.\\n\\n",omegaAB)
```

---

**Scilab code Exa 6.3** To find velocity and angular velocity

```
1 //To find velocity and angular velocity
2 clc
3 //Given:
```

```

4 NOA=120 //rpm
5 OA=200/1000 //m/s
6 //Results:
7 //Refer Fig. 6.15
8 //Calculating the angular velocity of the crank OA
9 omegaOA=2*pi*NOA/60 //rad/s
10 //Calculating the velocity of crank OA
11 vOA=omegaOA*OA //m/s
12 vA=vOA
13 //By measurement from the instantaneous centre
    diagram, Fig. 6.16,
14 I13A=840/1000, I13B=1070/1000, I14B=400/1000, I14C
    =200/1000, I15C=740/1000, I15D=500/1000 //m
15 //Calculating the velocity of point B
16 vB=vA/I13A*I13B //m/s
17 //Calculating the velocity of point C
18 vC=vB/I14B*I14C //m/s
19 //Calculating the velocity of point B
20 vD=vC/I15C*I15D //m/s
21 //Calculating the angular velocity of the link AB
22 omegaAB=vA/I13A //rad/s
23 //Calculating the angular velocity of the link BC
24 omegaBC=vB/I14B //rad/s
25 //Calculating the angular velocity of the link CD
26 omegaCD=vC/I15C //rad/s
27 //Results:
28 printf("\n\n The velocity of point B, vB = %.1f m/s
    .\n", vB)
29 printf(" The velocity of point C, vC = %.1f m/s.\n",
    vC)
30 printf(" The velocity of point D, vD = %.2f m/s.\n",
    vD)
31 printf(" The angular velocity of the link AB,
    omegaAB = %.2f rad/s.\n", omegaAB)
32 printf(" The angular velocity of the link BC,
    omegaBC = %d rad/s.\n", omegaBC)
33 printf(" The angular velocity of the link CD,
    omegaCD = %.2f rad/s.\n\n", omegaCD)

```



---

**Scilab code Exa 6.4** To find the velocity

```
1 //To find the velocity
2 clc
3 //Given:
4 omega01A=100 //rad/s
5 O1A=100/1000 //m
6 //Solution:
7 //Refer Fig. 6.18
8 //Calculating the linear velocity of crank O1A
9 v01A=omega01A*O1A //m/s
10 vA=v01A
11 //By measurement from the instantaneous centre
    diagram, Fig. 6.19,
12 I13A=910/1000, I13B=820/1000, I15B=130/1000, I15D
    =50/1000, I16D=200/1000, I16E=400/1000 //m
13 //Calculating the velocity of point B
14 vB=vA/I13A*I13B //m/s
15 //Calculating the velocity of point D
16 vD=vB/I15B*I15D //m/s
17 //Calculating the velocity of point E
18 vE=vD/I16D*I16E //m/s
19 //Results:
20 printf(" \n\n The velocity of point B, vB = %.2 f m/s
    .\n", vB)
21 printf(" The velocity of point D, vD = %.2 f m/s.\n",
    vD)
22 printf(" The velocity of point E, vE = %.2 f m/s.\n",
    vE)
```

---

**Scilab code Exa 6.5** To find the velocity

```

1 //To find the velocity
2 clc
3 //Given:
4 N01A=400 //rpm
5 O1A=16/1000 //m
6 //Solution:
7 //Refer Fig. 6.21
8 //Calculating the angular velocity of the crank O1A
9 omega01A=2*%pi*N01A/60 //rad/s
10 //Calculating the linear velocity of the crank O1A
11 v01A=omega01A*O1A //m/s
12 vA=v01A
13 //By measurement from the instantaneous centre
    diagram, Fig. 6.22,
14 I13A=41/1000, I13B=50/1000, I14B=23/1000, I14C=28/1000,
    I15C=65/1000, I15D=62/1000 //m
15 //Calculating the velocity of point B
16 vB=vA/I13A*I13B //m/s
17 //Calculating the velocity of point C
18 vC=vB/I14B*I14C //m/s
19 //Calculating the velocity of of the needle at D
20 vD=vC/I15C*I15D //m/s
21 //Results:
22 printf("\\n\\n The velocity of the needle at D, vD = %
    .2 f m/s.\\n\\n", vD)

```

---

**Scilab code Exa 6.6** To find the velocity of ram

```

1 //To find velocity of ram
2 clc
3 //Given:
4 NOA=120 //rpm
5 //Solution:
6 //Refer Fig. 6.24
7 //Calculating the angular speed of crank OA

```

```
8 omega0A=2*%pi*N0A/60 //rad/s
9 //By measurement from the instantaneous centre
  diagram, Fig. 6.25,
10 I12I26=65/1000 //m
11 //Calculating the velocity of the ram
12 vD=omega0A*I12I26 //m/s
13 //Results:
14 printf("\n\n The velocity of ram D, vD = %.3f m/s.\n
  \n",vD)
```

---

# Chapter 7

## Velocity in Mechanisms Relative Velocity Method

Scilab code Exa 7.1 To find the angular velocity

```
1 //To find the angular velocity
2 clc
3 //Given:
4 NBA=120 //rpm
5 AB=40/1000 , CD=80/1000 //m
6 //Solution:
7 //Refer Fig. 7.7
8 //Calculating the angular velocity of the crank AB
9 omegaBA=2*%pi*NBA/60 //rad/s
10 //Calculating the velocity of B with respect to A
11 vBA=omegaBA*AB //m/s
12 vB=vBA
13 //By measurement from the velocity diagram , Fig.
    7.7(b) ,
14 vCD=0.385 //m/s
15 vC=vCD
16 //Calculating the angular velocity of link CD
17 omegaCD=vCD/CD //rad/s
18 //Results :
```

```

19 printf("\n\n The angular velocity of link CD,
    omegaCD = %.1f rad/s, clockwise about D.\n\n",
    omegaCD)

```

---

**Scilab code Exa 7.2** To find velocities angular velocities and position

```

1 //To find velocities , angular velocities and
  position
2 clc
3 //Given:
4 NB0=180 //rpm
5 OB=0.5 , PB=2 , d0=50/1000 , dB=60/1000 , dC=30/1000 //m
6 //Solution:
7 //Refer Fig. 7.8
8 //Calculating the angular velocity of the crank BO
9 omegaB0=2*%pi*NB0/60 //rad/s
10 //Calculating the linear velocity of B with respect
    to O
11 vB0=omegaB0*OB //m/s
12 vB=vB0
13 //By measurement from the velocity diagram , Fig.
    7.8(b) ,
14 vP=8.15 , vPB=6.8 , vE=8.5 , bg=5 , bp=vPB , vG=8 //m/s
15 //Calculating the angular velocity of the connecting
    rod PB
16 omegaPB=vPB/PB //rad/s
17 //Calculating the velocity of rubbing at the pin of
    crank-shaft
18 vCS=d0/2*omegaB0 //Velocity of rubbing at the pin of
    crank-shaft , m/s
19 //Calculating the velocity of rubbing at the pin of
    crank
20 vC=dB/2*(omegaB0+omegaPB) //Velocity of rubbing at
    the pin of crank , m/s
21 //Calculating the velocity of rubbing at the pin of

```

```

    cross-head
22 vPCH=dC/2*omegaPB //Velocity of rubbing at the pin
    of cross-head, m/s
23 //Calculating the position of point G on the
    connecting rod
24 BG=bg/bp*PB //m
25 //Results:
26 printf("\n\n The velocity of piston P, vP = %.2f m/s
    .\n",vP)
27 printf(" The angular velocity of connecting rod,
    omegaPB = %.1f rad/s, anticlockwise.\n",omegaPB)
28 printf(" The velocity of point E on the connecting
    rod, vE = %.1f m/s.\n",vE)
29 printf(" The velocity of rubbing at the pin of crank
    -shaft is %.2f m/s.\n",vCS)
30 printf(" The velocity of rubbing at the pin of crank
    is %.4f m/s.\n",vC)
31 printf(" The velocity of rubbing at the pin of cross
    -head is %.3f m/s.\n",vPCH)
32 printf(" The position of point G on the connecting
    rod, BG = %.2f m.\n",BG)
33 printf(" The linear velocity of point G, vG = %d m/s
    .\n\n",vG)

```

---

**Scilab code Exa 7.3** To find the velocity

```

1 //To find the velocity
2 clc
3 //Given:
4 NAO=600 //rpm
5 OA=28/1000, BD=46/1000 //m
6 //Solution:
7 //Refer Fig. 7.10
8 //Calculating the angular velocity of crank AO
9 omegaAO=2*pi*NAO/60 //rad/s

```

```

10 //Calculating the velocity of A with respect to O
11 vAO=omegaAO*OA //m/s
12 vA=vAO
13 //By measurement from the velocity diagram, Fig.
    7.10(b),
14 vD=1.6, vDB=1.7 //m/s
15 //Calculating the angular velocity of D with respect
    to B
16 omegaBD=vDB/BD //rad/s
17 //Results:
18 printf("\n\n The velocity of the slider D, vD = %.1f
    m/s.\n", vD)
19 printf(" The angular velocity of the link BD,
    omegaBD = %.2f rad/s, clockwise sbout B.\n\n",
    omegaBD)

```

---

**Scilab code Exa 7.4** To find velocity angular velocity and rubbing speed

```

1 //To find velocity , angular velocity and rubbing
    speed
2 clc
3 //Given:
4 NBA=120 //rpm
5 AB=150/1000, DC=450/1000, BC=450/1000, dC=50/1000, rC=dC
    /2 //m
6 //Sloution:
7 //Refer Fig. 7.12
8 //Calculating the angular velocity of the crank AB
9 omegaBA=2*pi*NBA/60 //rad/s
10 //Calculating the linear velocity of B with respect
    to A
11 vBA=omegaBA*AB //m/s
12 vB=vBA
13 //By measurement from the velocity diagram, Fig.
    7.12(b),

```

```

14 vF=0.7 , vCD=2.25 , vCB=2.25 //m/s
15 //Calculating the angular velocity of DC
16 omegaDC=vCD/DC //rad/s
17 //Calculating the angular velocity of BC
18 omegaCB=vCB/BC //rad/s
19 //Calculating the rubbing speed at the pin C
20 vr=(omegaCB-omegaDC)*rC //The rubbing speed at the
    pin C,m/s
21 //Results:
22 printf("\n\n The velocity of block F, vF = %.1f m/s
    .\n" ,vF)
23 printf(" The angular velocity of DC, omegaDC = %d
    rad/s, anticlockwise about D.\n",omegaDC)
24 printf(" The rubbing speed at the pin C is %d m/s.\n
    \n" ,vr)

```

---

**Scilab code Exa 7.5** To find velocity and angular velocity

```

1 //To find velocity and angular velocity
2 clc
3 //Given:
4 NAO=120 //rpm
5 OA=100/1000 , CE=350/1000 //m
6 //Solution:
7 //Refer Fig. 7.13
8 //Calculating the angular speed of the crank OA
9 omegaAO=2*pi*NAO/60 //rad/s
10 //Calculating the velocity of A with respect to O
11 vAO=omegaAO*OA //m/s
12 vA=vAO
13 //By measurement from the velocity diagram , Fig.
    7.14(b) ,
14 vF=0.53 , od=1.08 , vCE=0.44 //m/s
15 //Calculating the angular velocity of CE
16 omegaCE=vCE/CE //rad/s

```



```

17 //Results:
18 printf("\n\n The velocity of F, vF = %.2f m/s.\n",vF
   )
19 printf(" The velocity of sliding of CE in the
   trunnion is %.2f m/s.\n",od)
20 printf(" The angular velocity of CE, omegaCE = %.2f
   rad/s, clockwise about E.\n\n",omegaCE)

```

---

**Scilab code Exa 7.6** To find the absolute velocity

```

1 //To find the absolute velocity
2 clc
3 //Given:
4 NCO=120 //rpm
5 OC=125/1000 //m
6 //Solution:
7 //Refer Fig. 7.15
8 //Calculating the angular velocity of the crank CO
9 omegaCO=2*%pi*NCO/60 //rad/s
10 //Calculating the linear velocity of C with respect
   to O
11 vCO=omegaCO*OC //m/s
12 vC=vCO
13 //By measurement from the velocity diagram, Fig.
   7.16(b),
14 vCO=1.57, vE=0.7 //m/s
15 //Results:
16 printf("\n\n The absolute velocity of point E of the
   lever, vE = %.1f m/s.\n\n",vE)

```

---

**Scilab code Exa 7.7** To find linear and angular velocity

```

1 //To find linear and angular velocity

```

```

2  clc
3  //Given:
4  NB01=40 //rpm
5  O1O2=800/1000 , O1B=300/1000 , O2D=1300/1000 , DR=400/1000
   //m
6  //Solution:
7  //Refer Fig. 7.18
8  //Calculating the angular speed of the crank BO
9  omegaB01=2*%pi*NB01/60 //rad/s
10 //Calculating the velocity of B with respect to O1
11 vB01=omegaB01*O1B //m/s
12 vB=vB01
13 //By measurement from the velocity diagram , Fig.
   7.18(b) ,
14 vR=1.44 , vD02=1.32 //m/s
15 vD=vD02
16 //Calculating the angular velocity of the link O2D
17 omegaD02=vD02/O2D //rad/s
18 //Results:
19 printf(" \n\n The velocity of the ram R, vR = %.2 f m/
   s.\n" ,vR)
20 printf(" The angular velocity of the link O2D,
   omegaDO2 = %.3 f rad/s, anticlockwise about O2.\n\n"
   ,omegaD02)

```

---

**Scilab code Exa 7.8** To find speed and time ratio

```

1  //To find speed and time ratio
2  clc
3  //Given:
4  NA01=60 //rpm
5  O1A=85 , rQ=50 //mm
6  //Solution:
7  //Refer Fig. 7.20 and Fig. 7.21
8  //Calculating the angular velocity of AO1

```

```

9  omegaA01=2*%pi*NA01/60 //rad/s
10 //Calculating the velocity of A with respect to O1
11 vA01=omegaA01*O1A //mm/s
12 vA=vA01
13 //By measurement from the velocity diagram, Fig.
    7.20(b),
14 vD02=410 //mm/s
15 O2D=264 //mm
16 angleB102B2=60*%pi/180 //rad
17 funcprot(0) //To vary the Scilab function 'beta'
18 alpha=120,beta=240 //degrees
19 //Calculating the angular velocity of the quadant Q
20 omegaQ=vD02/O2D //rad/s
21 //Calculating the linear speed of the rack
22 vR=omegaQ*rQ //mm/s
23 //Calculating the ratio of times of lowering and
    raising the rack
24 r=beta/alpha
25 //Calculating the length of stroke of the rack
26 L=rQ*angleB102B2 //mm
27 //Results:
28 printf("\n\n The linear speed of the rack , vR = %.1f
    mm/s.\n",vR)
29 printf(" The ratio of times of lowering and raising
    the rack is %d.\n",r)
30 printf(" The length of the stroke of the rack is %.2
    f mm.\n\n",L)

```

---

**Scilab code Exa 7.9** To find velocity and angular velocity

```

1 //To find velocity and angular velocity
2 clc
3 //Given:
4 NPO=120 //rpm
5 OQ=100/1000 , OP=200/1000 , RQ=150/1000 , RS=500/1000 //m

```

```

6 //Solution:
7 //Refer Fig. 7.22
8 //Calculating the angular speed of the crank PO
9 omegaPO=2*%pi*NPO/60 //rad/s
10 //Calculating the velocity of P with respect to O
11 vPO=omegaPO*OP //m/s
12 vP=vPO
13 //By measurement from the velocity diagram, Fig.
    7.23(b),
14 vS=0.8, vSR=0.96, vTP=0.85 //m/s
15 //Calculating the angular velocity of link RS
16 omegaRS=vSR/RS //rad/s
17 //Results:
18 printf("\n\n The velocity of the slider S (cutting
    tool), vS = %.1f m/s.\n", vS)
19 printf(" The angular velocity of the link RS,
    omegaRS = %.2f rad/s, clockwise about R.\n",
    omegaRS)
20 printf(" The velocity of the sliding block T on the
    slotted lever QT, vTP = %.2f m/s.\n\n", vTP)

```

---

**Scilab code Exa 7.10** To find velocity and resisting torque

```

1 //To find velocity and resisting torque
2 clc
3 //Given:
4 NAD=100 //rpm
5 TA=50 //N-m
6 DA=300/1000, CB=360/1000, AB=CB, DC=600/1000 //m
7 eta=70/100 //%
8 //Solution:
9 //Refer Fig. 7.25
10 //Calculating the angular velocity of the crank AD
11 omegaAD=2*%pi*NAD/60 //rad/s
12 //Calculating the velocity of A with respect to D

```

```

13 vAD=omegaAD*DA //m/s
14 vA=vAD
15 //By measurement from the velocity diagram, Fig.
    7.25(b),
16 vBC=2.25 //m/s
17 vB=vBC
18 //Calculating the angular velocity of the driven
    link CB
19 omegaBC=vBC/CB //rad/s
20 //Calculating the actual mechanical advantage
21 omegaA=omegaAD, omegaB=omegaBC
22 MAactual=eta*omegaA/omegaB
23 //Calculating the resisting torque
24 TB=eta*TA*omegaA/omegaB //N-m
25 //Results:
26 printf("\n\n The velocity of the point B, vB = %.2 f
    m/s.\n", vB)
27 printf(" The angular velocity of the driven link CB,
    omegaBC = %.2 f rad/s.\n", omegaBC)
28 printf(" The actual mechanical advantage, M.A.(
    actual) = %.2 f.\n", MAactual)
29 printf(" The resisting torque, TB = %.1 f N-m.\n\n",
    TB)

```

---

**Scilab code Exa 7.11** To find velocity ratio

```

1 //To find velocity ratio
2 clc
3 //Given:
4 WC=2.5*1000, WD=4*1000 //N
5 OA=175/1000, AB=180/1000, AD=500/1000, BC=325/1000 //m
6 //Solution:
7 //Refer Fig. 7.26
8 //Assuming the speed of crank OA to be 'N'
9 //Calculating the angular velocity of crank OA

```

```

10 omegaAO=mulf('2*%pi/60', 'N')
11 //Calculating the velocity of A with respect to O
12 vAO=mulf('omegaAO', 'OA')
13 vA=vAO
14 //Assume the vector oa (i.e. velocity of A) as 20 m/
    s
15 N=20/(2*%pi/60*OA) //mm
16 //By measurement from the velocity diagram, Fig.
    7.27(b),
17 vC=35, vD=21 //mm
18 //Calculating the velocity ratio between C and the
    ram D
19 r=vC/vD //The velocity ratio between C and the ram D
20 //Calculating the efficiency of the machine
21 eta=(WD*vD)/(WC*vC)*100 //%
22 //Results:
23 printf("\n\n The velocity ratio between C and the
    ram D is %.2f.\n", r)
24 printf(" The efficiency of the machine, eta = %d %s
    .\n\n", eta, '%')

```

---

**Scilab code Exa 7.12** To find velocity angular velocity and torque

```

1 //To find velocity , angular velocity and torque
2 clc
3 //Given:
4 NAO=180 //rpm
5 OA=180/1000, CB=240/1000, AB=360/1000, BD=540/1000 //m
6 FD=2*1000 //N
7 DA=30/1000, DD=DA, rA=DA/2, rD=DD/2 //m
8 //Solution:
9 //Refer Fig. 7.28
10 //Calculating the angular velocity of the crank OA
11 omegaAO=2*%pi*NAO/60 //rad/s
12 //Calculating the velocity of A with respect to O

```

```

13 vA0=omegaA0*OA
14 vA=vA0
15 //By measurement fro the velocity diagram , Fig.
    7.29(b)
16 vD=2.05 , vBA=0.9 , vBC=2.8 , vDB=2.4 //m/s
17 //Calculating the angular velocity of the link AB
18 omegaAB=vBA/AB //rad/s
19 //Calculating the angular velocity of the link CB
20 omegaCB=vBC/CB //rad/s
21 //Calculating the angular velocity of the link BD
22 omegaBD=vDB/BD //rad/s
23 //Calculating the relative angular velocity at A
24 rvA=omegaCB-omegaAB+omegaBD //The relative angular
    velocity at A, rad/s
25 //Calculating the relative angular velocity at D
26 rvD=omegaBD //The relative angular velocity at D,
    rad/s
27 //Calculating the velocity of rubbing on the pin A
28 vrA=rvA*rA*1000 //The velocity of rubbing on the pin
    A, mm/s
29 //Calculating the velocity of rubbing on the pin D
30 vrD=rvD*rD*1000 //The velocity of rubbing on the pin
    D, mm/s
31 //Calculating the torque applied to crank OA
32 TA=FD*vD/omegaA0 //N-m
33 //Results:
34 printf("\n\n The velocity of slider D, vD = %.2f m/s
    .\n",vD)
35 printf(" The angular velocity of the link AB,
    omegaAB = %.1f rad/s, anticlockwise about A.\n",
    omegaAB)
36 printf(" The angular velocity of the link CB,
    omegaCB = %.2f rad/s, anticlockwise about C.\n",
    omegaCB)
37 printf(" The angular velocity of the link BD,
    omegaBD = %.2f rad/s, clockwise about B.\n",
    omegaBD)
38 printf(" The velocity of rubbing on the pin A is %d

```

```

    mm/s.\n",vrA)
39 printf(" The velocity of rubbing on the pin D is %d
    mm/s.\n",vrD)
40 printf(" The torque applied to the crank OA, TA = %
    .1f N-m.\n\n",TA)

```

---

**Scilab code Exa 7.13** To find the velocities

```

1 //To find the velocities
2 clc
3 //Given:
4 NBA=180 //rpm
5 AB=0.45, BD=1.5, BC=0.9, CE=BC //m
6 FD=500, FE=750 //N
7 //Solution:
8 //Refer Fig. 7.31
9 //Calculating the angular velocity of the crank AB
10 omegaBA=2*pi*NBA/60 //rad/s
11 //Calculating the velocity of B with respect to A
12 vBA=omegaBA*AB //m/s
13 vB=vBA
14 //By measurement from the velocity diagram, Fig.
    7.31(b),
15 vD=9.5, vE=1.7 //m/s
16 //Calculating the power input
17 Pi=FD*vD-FE*vE //N-m/s
18 //Calculating the turning moment at A
19 TA=Pi/omegaBA //N-m
20 //Results:
21 printf("\n\n The velocity of slider D, vD = %.1f m/s
    .\n",vD)
22 printf(" The velocity of slider E, vE = %.1f m/s.\n"
    ,vE)
23 printf(" The turning moment at A, TA = %.1f N-m.\n\n"
    ",TA)

```





# Chapter 8

## Acceleration in Mechanisms

**Scilab code Exa 8.1** To find linear and angular velocity and acceleration

```
1 //To find linear and angular velocity and
   acceleration
2 clc
3 //Given:
4 NB0=300 //rpm
5 OB=150/1000 ,BA=600/1000 //m
6 //Solution:
7 //Refer Fig. 8.4
8 //Calculating the angular velocity of BO
9 omegaB0=2*%pi*NB0/60 //rad/s
10 //Calculating the linear velocity of B with respect
    to O
11 vB0=omegaB0*OB //m/s
12 vB=vB0
13 //By measurement from the velocity diagram , Fig.
    8.4(b) ,
14 vAB=3.4 ,vD=4.1 //m/s
15 //Calculating the radial component of the
    acceleration of B with respect of O
16 arB0=vB0^2/OB //m/s^2
17 aB=arB0
```

```

18 //Calculating the radisla component of the
    accaleration of A with respect to B
19 arAB=vAB^2/BA //m/s^2
20 //By measurement from the acceleration diagram, Fig.
    8.4(c),
21 aD=117, adashAB=103 //m/s^2
22 //Calculating the angular velocity of the connecting
    rod
23 omegaAB=vAB/BA //rad/s^2
24 //Calculating the angular acceleration of the
    connecting rod
25 alphaAB=adashAB/BA //rad/s^2
26 //Results:
27 printf("\n\n The linear velocity of the midpoint of
    the connecting rod, vD = %.1f m/s.\n",vD)
28 printf(" The linear acceleration of the midpoint of
    the connecting rod, aD = %d m/s^2.\n",aD)
29 printf(" The angular velocity of the connecting rod,
    omegaAB = %.2f rad/s, anticlockwise about B.\n",
    omegaAB)
30 printf(" The angular acceleration of the connecting
    rod, alphaAB = %.2f rad/s^2, clockwise about B.\n
    \n",alphaAB)

```

---

**Scilab code Exa 8.2** To find linear and angular velocity and acceleration

```

1 //To find linear and angular velocity and
    acceleration
2 clc
3 //Given:
4 omegaBC=75 //rad/s
5 alphaBC=1200 //rad/s^2
6 CB=100/1000,BA=300/1000 //m/
7 //Solution:
8 //Refer Fig. 8.5

```

```

9 //Calculating the linear velocity of B with respect
  to C
10 vBC=omegaBC*CB //m/s
11 //Calculating the tangential component of the
  acceleration of B with respect to C
12 alphasBC=alphaBC*CB //m/s^2
13 //By measurement from the velocity diagram, Fig.
  8.6(b),
14 vG=6.8,vAB=4 //m/s
15 //Calculating the angular velocity of AB
16 omegaAB=vAB/BA //rad/s
17 //Calculating the radial component of the
  acceleration of B with respect to C
18 arBC=vBC^2/CB //m/s^2
19 //Calculating the radial component of the
  acceleration of A with respect to B
20 arAB=vAB^2/BA //m/s^2
21 //By measurement from the acceleration diagram, Fig.
  8.6(c),
22 arBC=120,arAB=53.3,aG=414,atAB=546 //m/s^2
23 //Calculating the angular acceleration of AB
24 alphaAB=atAB/BA //rad/s^2
25 //Results:
26 printf("\n\n The velocity of G, vG = %.1f m/s.\n",vG
  )
27 printf(" The angular velocity of AB, omegaAB = %.1f
  rad/s, clockwise.\n",omegaAB)
28 printf(" The acceleration of G, aG = %d m/s^2.\n",aG
  )
29 printf(" The angular acceleration of AB, alphaAB =
  %d rad/s^2.\n\n",alphaAB)

```

---

**Scilab code Exa 8.3** To find linear and angular acceleration

```

1 //To find linear and angular acceleration

```

```

2  clc
3  //Given:
4  vC=1, vCD=vC //m/s
5  aC=2.5 //m/s^2
6  AB=3, BC=1.5 //m
7  //Solution:
8  //Refer Fig. 8.8
9  //By measurement from the velocity diagram, Fig.
   8.8(b),
10 vBA=0.72, vBC=0.72 //m/s
11 //Calculating the radial component of acceleration
   of B with respect to C
12 arBC=vBC^2/BC //m/s^2
13 //Calculating the radial component of acceleration
   of B with respect to A
14 arBA=vBA^2/AB //m/s^2
15 //By measurement from the acceleration diagram, Fig.
   8.8(c),
16 aCD=2.5, aC=aCD, arBC=0.346, arBA=0.173, atBA=1.41, atBC
   =1.94, vectorbb=1.13, vectorab=0.9 //m/s^2
17 //Calculating the angular acceleration of AB
18 alphaAB=atBA/AB //rad/s^2
19 //Calculating the angular acceleration of BC
20 alphaBC=atBC/BC //rad/s^2
21 //Results:
22 printf(" \n\n The magnitude of vertical component of
   the acceleration of the point B is %.2f m/s^2.\n"
   ,vectorbb)
23 printf(" The magnitude of horizontal component of
   the acceleration of the point B is %.1f m/s^2.\n"
   ,vectorab)
24 printf(" The angular acceleration of the link AB,
   alphaAB = %.2f rad/s^2.\n",alphaAB)
25 printf(" The angular acceleration of the link BC,
   alphaBC = %.1f rad/s^2.\n\n",alphaBC)

```

---

**Scilab code Exa 8.4** To find the angular velocity and acceleration

```
1 //To find the angular velocity and acceleration
2 clc
3 //Given:
4 omegaQP=10 //rad/s
5 PQ=62.5/1000, QR=175/1000, RS=112.5/1000, PS=200/1000
   //m
6 //Solution:
7 //Refer Fig. 8.9
8 //Calculating the velocity of Q with respect to P
9 vQP=omegaQP*PQ //m/s
10 vQ=vQP
11 //By measurement from the velocity diagram, Fig.
   8.9(b),
12 vRQ=0.333, vRS=0.426, vR=vRS //m/s
13 //Calculating the angular velocity of link QR
14 omegaQR=vRQ/QR //rad/s
15 //Calculating the angular velocity of link RS
16 omegaRS=vRS/RS //rad/s
17 //Calculating the radial component of the
   acceleration of Q with respect to P
18 arQP=vQP^2/PQ //m/s^2
19 aQP=arQP, aQ=aQP
20 //Calculating the radial component of the
   acceleration of R with respect to Q
21 arRQ=vRQ^2/QR //m/s^2
22 //Calculating the radial component of the
   acceleration of R with respect to S
23 arRS=vRS^2/RS //m/s^2
24 aRS=arRS, aR=aRS
25 //By measurement from the acceleration diagram, Fig.
   8.9(c),
26 atRQ=4.1, atRS=5.3 //m/s^2
```

```

27 //Calculating the angular acceleration of link QR
28 alphaQR=atRQ/QR //rad/s^2
29 //Calculating the angular acceleration of link RS
30 alphaRS=atRS/RS //rad/s^2
31 //Results:
32 printf("\n\n The angular velocity of link QR,
    omegaQR = %.1f rad/s, anticlockwise.\n",omegaQR)
33 printf(" The angular velocity of link RS, omegaRS =
    %.2f rad/s, clockwise.\n",omegaRS)
34 printf(" The angular acceleration of link QR,
    alphaQR = %.2f rad/s^2, anticlockwise.\n",alphaQR
    )
35 printf(" The angular acceleration of link RS,
    alphaRS = %.1f rad/s^2, anticlockwise.\n\n",
    alphaRS)

```

---

**Scilab code Exa 8.5** To find angular velocities and accelerations

```

1 //To find angular velocities and accelerations
2 clc
3 //Given:
4 omegaAP1=10 //rad/s
5 alphaAP1=30 //rad/s^2
6 P1A=300/1000 , P2B=360/1000 , AB=P2B //m
7 //Solution:
8 //Refer Fig. 8.10
9 //Calculating the velocity of A with respect to P1
10 vAP1=omegaAP1/P1A //m/s
11 vA=vAP1
12 //By measurement from the velocity diagram , Fig.
    8.11(b) ,
13 vBP2=2.2 , vBA=2.05 //m/s
14 //Calculating the angular velocity of P2B
15 omegaP2B=vBP2/P2B //rad/s
16 //Calculating the angular velocity of AB

```

```

17 omegaAB=vBA/AB //rad/s
18 //Calculating the tangential component of the
    acceleration of A with respect to P1
19 atAP1=alphaAP1*P1A //m/s^2
20 //Calculating the radial component of the
    acceleration of A with respect to P1
21 arAP1=vAP1^2/P1A //m/s^2
22 //Calculating the radial component of the
    acceleration of B with respect to A
23 arBA=vBA^2/AB //m/s^2
24 //Calculating the radial component of B with respect
    to P2
25 arBP2=vBP2^2/P2B //m/s^2
26 //By measurement from the acceleration diagram, Fig.
    8.11(c),
27 aBP2=29.6, aB=aBP2, atBA=13.6, atBP2=26.6 //m/s^2
28 //Calculating the angular acceleration of P2B
29 alphaP2B=atBP2/P2B //rad/s^2
30 //Calculating the angular acceleration of AB
31 alphaAB=atBA/AB //rad/s^2
32 //Results:
33 printf("\n\n The velocity of P2B, vBP2 = %.1f m/s.\n
    ",vBP2)
34 printf(" The angular velocity of P2B, omegaP2B = %.1
    f rad/s, clockwise.\n",omegaP2B)
35 printf(" The angular velocity of AB, omegaAB = %.1f
    rad/s, anticlockwise.\n",omegaAB)
36 printf(" The acceleration of the joint B, aB = %.1f
    m/s^2.\n",aB)
37 printf(" The angular acceleration of P2B, alphaP2B =
    %.1f rad/s^2, anticlockwise.\n",alphaP2B)
38 printf(" The angular acceleration of AB, alphaAB = %
    .1f rad/s^2, anticlockwise.\n\n",alphaAB)

```

---

**Scilab code Exa 8.6** To find velocities and accelerations



```

1 //To find velocities and accelerations
2 clc
3 //Given:
4 NAO=20 //rpm
5 OA=300/1000 , AB=1200/1000 , BC=450/1000 , CD=BC //m
6 //Solution:
7 //Refer Fig. 8.13
8 //Calculating the angular velocity of crank AO
9 omegaAO=2*%pi*NAO/60 //rad/s
10 //Calculating the linear velocity of A with respect
    to O
11 vAO=omegaAO*OA //m/s
12 vA=vAO
13 //By measurement from the velocity diagram , Fig.
    8.13(b) ,
14 vB=0.4 , vD=0.24 , vDC=0.37 , vBA=0.54 //m/s
15 //Calculating the angular velocity of CD
16 omegaCD=vDC/CD //rad/s
17 //Calculating the radial component of the
    acceleration of A with respect to O
18 arAO=vAO^2/OA //m/s^2
19 //Calculating the radial component of the
    acceleration of B with respect to A
20 arBA=vBA^2/AB //m/s^2
21 //Calculating the radial component of the
    acceleration of D with respect to C
22 arDC=vDC^2/CD //m/s^2
23 //By measurement from the acceleration diagram , Fig.
    8.13(c) ,
24 aD=0.16 , atDC=1.28 //m/s^2
25 //Calculating the angular acceleration of CD
26 alphaCD=atDC/CD //rad/s^2
27 //Results:
28 printf(" \n\n Velocity of sliding at B, vB = %.1f m/s
    .\n" , vB)
29 printf(" Velocity of sliding at D, vD = %.2f m/s.\n"
    , vD)
30 printf(" Angular velocity of CD, omegaCD = %.2f rad/

```

```

    s.\n",omegaCD)
31 printf(" Linear acceleration of D, aD = %.2f m/s^2.\n",aD)
32 printf(" Angular acceleration of CD, alphaCD = %.2f rad/s^2, clockwise.\n\n",alphaCD)

```

---

**Scilab code Exa 8.7** To find linear and angular acceleration

```

1 //To find linear and angular acceleration
2 clc
3 //Given:
4 NAO=180 //rpm
5 OA=150/1000 , AB=450/1000 , PB=240/1000 , CD=660/1000 //m
6 //solution:
7 //Refer Fig. 8.15
8 //Calculating the angular speed of crank AO
9 omegaAO=2*pi*NAO/60 //rad/s
10 //Calculating the velocity of A with respect to O
11 vAO=omegaAO*OA //m/s
12 vA=vAO
13 //By measurement from the velocity diagram, Fig.
14 // 8.15(b),
15 vD=2.36 , vDC=1.2 , vBA=1.8 , vBP=1.5 //m/s
16 //Calculating the radial component of the
17 // acceleration of B with respect to A
18 arAO=vBA^2/AB //m/s^2
19 //Calculating the radial component of the
20 // acceleration of B with respect to A
21 arBA=vBA^2/AB //m/s^2
22 //Calculating the radial component of the
23 // acceleration of B with respect to P
24 arBP=vBP^2/PB //m/s^2
25 //Calculating the radial component of D with respect
26 // to C
27 arDC=vDC^2/CD //m/s^2

```

```

23 //By measurement from the acceleration diagram , Fig.
    8.15(c) ,
24 aD=69.6 , atDC=17.4 //m/s^2
25 //Calculating the angular acceleration of CD
26 alphaCD=atDC/CD //rad/s^2
27 //Results :
28 printf("\n\n Acceleration of slider D, aD = %.1f m/s
    ^2.\n" , aD)
29 printf(" Angular acceleration of link CD, alphaCD =
    %.1f rad/s^2.\n\n" , alphaCD)

```

---

**Scilab code Exa 8.8** To find linear and angular velocities and accelerations

```

1 //To find linear and angular velocities and
    accelerations
2 clc
3 //Given :
4 NAO=180 //rpm
5 OA=180/1000 , CB=240/1000 , AB=360/1000 , BD=540/1000 //m
6 alphaAO=50 //rad/s^2
7 //Solution :
8 //Refer Fig. 8.17
9 //Calculating the angular speed of crank AO
10 omegaAO=2*pi*NAO/60 //rad/s
11 //Calculating the velocity of A with respect to O
12 vAO=omegaAO*OA //m/s
13 vA=vAO
14 //By measurement from the velocity diagram , Fig.
    8.17(b) ,
15 vBA=0.9 , vBC=2.4 , vDB=2.4 , vD=2.05 //m/s
16 //Calculating the angular velocity of BD
17 omegaBD=vDB/BD //rad/s
18 //Calculating the tangential component of the
    acceleration of A with respect to O
19 atAO=alphaAO*OA //m/s^2

```

```

20 //Calculating the radial component of the
    acceleration of A with respect to O
21 arA0=vA0^2/OA //m/s^2
22 //Calculating the radial component of the
    acceleration of B with respect to A
23 arBA=vBA^2/AB //m/s^2
24 //Calculating the radial component of the
    acceleration of B with respect to C
25 arBC=vBC^2/AB //m/s^2
26 //Calculating the radial component of the
    acceleration of D with respect to B
27 arDB=vDB^2/BD //m/s^2
28 //By measurement from the acceleration diagram, Fig.
    8.17(c),
29 aD=13.3, atDB=38.5 //m/s^2
30 //Calculating the angular acceleration of BD
31 alphaBD=atDB/BD //rad/s^2
32 //Results:
33 printf("\n\n Velocity of slider D, vD = %.2 f m/s.\n"
    ,vD)
34 printf(" Angular velocity of BD, omegaBD = %.1 f rad/
    s.\n",omegaBD)
35 printf(" Acceleration of slider D, aD = %.1 f m/s^2.\n"
    ,aD)
36 printf(" Angular acceleration of BD, alphaBD = %.1 f
    rad/s^2, clockwise.\n\n",alphaBD)

```

---

**Scilab code Exa 8.9** To find velocity and accelerations

```

1 //To find velocity and accelerations
2 clc
3 //Given:
4 omegaA01=100 //rad/s
5 O1A=100/1000, AC=700/1000, BC=200/1000, BD=150/1000, O2D
    =200/1000, O2E=400/1000, O3C=200/1000 //m

```

```

6 //Solution:
7 //Refer Fig. 8.19
8 //Calculating the linear velocity of A with respect
  to O1
9 vA01=omegaA01/O1A //m/s
10 vA=vA01
11 //By measurement from the velocity diagram, Fig.
  8.19(b),
12 vCA=7, vC03=10, vC=vC03, vDB=10.2, vD02=2.8, vD=vD02, vE
  =5.8, vE02=vE //m/s
13 //Calculating the radial component of the
  acceleration of A with respect to O1
14 arA01=vA01^2/O1A //m/s^2
15 aA01=arA01, aA=aA01
16 //Calculating the radial component of the
  acceleration of C with respect to A
17 arCA=vCA^2/AC //m/s^2
18 //Calculating the radial component of the
  acceleration of C with respect to O3
19 arC03=vC03^2/O3C //m/s^2
20 //Calculating the radial component of the
  acceleration of D with respect to B
21 arDB=vDB^2/BD //m/s^2
22 //Calculating the radial component of the
  acceleration of D with respect to O2
23 arD02=vD02^2/O2D //m/s^2
24 //Calculating the radial component of the
  acceleration of E with respect to O2
25 arE02=vE02^2/O2E //m/s^2
26 //By measurement from the acceleration diagram, Fig.
  8.19(c),
27 aE=1200, atD02=610 //m/s^2
28 aE02=aE
29 aB=440 //Acceleration of point B, m/s^2
30 //Calculating the angular acceleration of the bell
  crank lever
31 alpha=atD02/O2D //The angular acceleration of the
  bell crank lever, rad/s^2

```

```

32 //Results:
33 printf("\n\n Velocity of the point E on the bell
    crank lever , vE = %.1f m/s.\n",vE)
34 printf(" Acceleration of point B = %d m/s ^2.\n",aB)
35 printf(" Acceleration of point E, aE = %d m/s ^2.\n",
    aE)
36 printf(" Angular acceleration of the bell crank
    lever = %d rad/s ^2, anticlockwise.\n\n",alpha)

```

---

**Scilab code Exa 8.10** To find velocity torque and acceleration

```

1 //To find velocity , torque and acceleration
2 clc
3 //Given:
4 NAO=100 //rpm
5 OA=150/1000 , AB=600/1000 , BC=350/1000 , CD=150/1000 , DE
    =500/1000 //m
6 dA=50/1000 , dB=dA , rA=dA/2 , rB=dB/2 //m
7 pF=0.35 //N/mm^2
8 DF=250 //mm
9 //Solution:
10 //Refer Fig. 8.21
11 //Calculating the angular speed of the crank AO
12 omegaAO=2*pi*NAO/60 //rad/s
13 //Calculating the velocity of A with respect to O
14 vAO=omegaAO*OA //m/s
15 vA=vAO
16 //By measurement from the velocity diagram , Fig.
    8.21(b) ,
17 vBA=1.65 , vBC=0.93 , vB=vBC , vED=0.18 , vEO=0.36 , vE=vEO , vF
    =vE //m/s
18 //Calculating the velocity of D with respect to C
19 vDC=vBC*CD/BC //m/s
20 //Calculating the angular velocity of B with respect
    to A

```

```

21 omegaBA=vBA/AB //rad/s
22 //Calculating the angular velocity of B with respect
    to C
23 omegaBC=vBC/BC //rad/s
24 //Calculating the rubbing velocity of pin at A
25 vrA=(omegaAO-omegaBA)*rA //The rubbing velocity of
    pin at A, m/s
26 //Calculating the rubbing velocity of pin at B
27 vrB=(omegaBA+omegaBC)*rB //The rubbing velocity of
    pin at B, m/s
28 //Calculating the force at the pump piston at F
29 FF=pF*pi/4*DF^2 //N
30 //Calculating the force required at the crankshaft A
31 FA=FF*vF/vA //N
32 //Calculating the torque required at the crankshaft
33 TA=FA*OA //N-m
34 //Calculating the radial component of the
    acceleration of A with respect to O
35 arAO=vAO^2/OA //m/s^2
36 //Calculating the radial component of the
    acceleration of B with respect to A
37 arBA=vBA^2/AB //m/s^2
38 //Calculating the radial component of the
    acceleration of B with respect to C
39 arBC=vBC^2/BC //m/s^2
40 //Calculating the radial component of the
    acceleration of E with respect to D
41 arED=vED^2/DE //m/s^2
42 //By measurement from the acceleration diagram, Fig.
    8.21(c),
43 aBC=9.2, aB=aBC, aBA=9, aE=3.8 //m/s^2
44 //Calculating the acceleration of D
45 aD=aBC*CD/BC //m/s^2
46 //Results:
47 printf("\n\n The velocity of the cross-head E, vE =
    %.2f m/s.\n",vE)
48 printf(" The rubbing velocity of pin at A = %.3f m/s
    .\n",vrA)

```

```

49 printf(" The rubbing velocity of pin at B = %.3f m/s
   .\n",vrB)
50 printf(" The torque required at the crankshaft , TA =
   %d N-m.\n",TA)
51 printf(" The acceleration of the crosshead E, aE = %
   .1f m/s ^2.\n\n",aE)

```

---

**Scilab code Exa 8.11** To find velocity and acceleration

```

1 //To find velocity and acceleration
2 clc
3 //Given:
4 NAO=150 //rpm
5 OA=150/1000 , AB=550/1000 , AC=450/1000 , DC=500/1000 , BE
   =350/1000 //m
6 //Solution:
7 //Refer Fig. 8.23
8 //Calculating the angular speed of the crank AO
9 omegaAO=2*pi*NAO/60 //rad/s
10 //Calculating the linear velocity of A with respect
   to O
11 vAO=omegaAO*OA //m/s
12 vA=vAO
13 //By measurement from the velocity diagram , Fig.
   8.23(b) ,
14 vCA=0.53 , vCD=1.7 , vC=vCD , vEB=1.93 , vE=1.05 //m/s
15 //Calculating the radial component of the
   acceleration of A with respect to O
16 arAO=vAO^2/OA //m/s^2
17 aA=arAO
18 //Calculating the radial component of the
   acceleration of C with respect to A
19 arCA=vCA^2/AC //m/s^2
20 //Calculating the radial component of the
   acceleration of C with respect to D

```



```

21 arCD=vCD^2/DC //m/s^2
22 //Calculating the radial component of the
    acceleration of E with respect to B
23 arEB=vEB^2/BE //m/s^2
24 //By measurement from the acceleration diagram, Fig.
    8.23(c),
25 aE=3.1 //m/s^2
26 //Results:
27 printf("\n\n Velocity of the ram E, vE = %.2f m/s.\n
    ",vE)
28 printf(" Acceleration of the ram E, aE = %.1f m/s
    ^2.\n\n",aE)

```

---

**Scilab code Exa 8.12** To find the velocity and acceleration

```

1 //To find the velocity and acceleration
2 clc
3 //Given:
4 NDC=1140 //rpm
5 AB=80/1000,CD=40/1000,BE=150/1000,DE=BE,EP=200/1000
    //m
6 //Solution:
7 //Refer Fig. 8.25
8 //Calculating the angular speed of the link CD
9 omegaDC=2*%pi*NDC/60 //rad/s
10 //Calculating the velocity of D with respect to C
11 vDC=omegaDC*CD //m/s
12 vD=vDC
13 //Calculating the angular speed of the larger wheel
14 omegaBA=omegaDC*CD/AB //rad/s
15 //Calculating the velocity of B with respect to A
16 vBA=omegaBA*AB //m/s
17 vB=vBA
18 //By measurement from the velocity diagram, Fig.
    8.25(b),

```

```

19 vEB=8.1 , vED=0.15 , vPE=4.7 , vP=0.35 //m/s
20 //Calculating the radial component of the
    acceleration of B with respect to A
21 arBA=vBA^2/AB //m/s^2
22 //Calculating the radial component of the
    acceleration of D with respect to C
23 arDC=vDC^2/CD //m/s^2
24 //Calculating the radial component of the
    acceleration of E with respect to B
25 arEB=vEB^2/BE //m/s^2
26 //Calculating the radial component of the
    acceleration of E with respect to D
27 arED=vED^2/DE //m/s^2
28 //Calculating the radial component of the
    acceleration of P with respect to E
29 arPE=vPE^2/EP //m/s^2
30 //By measurement from the acceleration diagram , Fig.
    8.25(c) ,
31 aP=655 //m/s^2
32 //Results:
33 printf("\n\n Velocity of P, vP = %.2f m/s.\n" ,vP)
34 printf(" Acceleration of the piston P, aP = %d m/s
    ^2.\n\n" ,aP)

```

---

**Scilab code Exa 8.13** To find velocity and acceleration

```

1 //To find velocity and acceleration
2 clc
3 //Given:
4 NBA=120 //rpm
5 AB=150/1000 , OC=700/1000 , CD=200/1000 //m
6 //Solution:
7 //Refer Fig. 8.29
8 //Calculating the angular speed of the crank AB
9 omegaAB=2*%pi*NBA/AB //rad/s

```

```

10 //Calculating the velocity of B with respect to A
11 vBA=omegaBA*AB //m/s
12 //By measurement from the velocity diagram, Fig.
    8.29(b),
13 vD=2.15, vBBdash=1.05, vDC=0.45, vBdash0=1.55, vC0=2.15
    //m/s
14 Bdash0=0.52 //m
15 //Calculating the angular velocity of the link OC or
    OB'
16 omegaC0=vC0/OC //rad/s
17 omegaBdash0=omegaC0 //rad/s
18 //Calculating the radial component of the
    acceleration of B with respect to A
19 arBA=omegaAB^2/AB //m/s^2
20 //Calculating the coriolis component of the
    acceleration of slider B with respect to the
    coincident point B'
21 acBBdash=2*omegaC0*vBBdash //m/s^2
22 //Calculating the radial component of the
    acceleration of D with respect to C
23 arDC=vDC^2/CD //m/s^2
24 //Calculating the radial component of the
    acceleration of B' with respect to O
25 arBdash0=vBdash0^2/Bdash0 //m/s^2
26 //By measurement fro the acceleration diagram, Fig.
    8.29(c),
27 aD=8.4, atBdash0=6.4 //m/s^2
28 //Calculating the angular acceleration of the
    slotted lever
29 alpha=atBdash0/Bdash0 //The angular acceleration of
    the slotted lever, rad/s^2
30 //Results:
31 printf("\n\n Velocity of the ram D, vD = %.2f m/s.\n
    ",vD)
32 printf(" Acceleration of the ram D, aD = %.1f m/s
    ^2.\n",aD)
33 printf(" Angular acceleration of the slotted lever =
    %.1f rad/s^2, anticlockwise.\n\n",alpha)

```

---

**Scilab code Exa 8.14** To find the acceleration

```
1 //To find the acceleration
2 clc
3 //Given:
4 NBA=200 //rpm
5 AB=75/1000 , PQ=375/1000 , QR=500/1000 //m
6 //Solution:
7 //Refer Fig. 8.31
8 //Calculating the angular velocity of the crank AB
9 omegaBA=2*%pi*NBA/60 //rad/s
10 //Calculating the velocity of B with respect to A
11 vBA=omegaBA*AB //m/s
12 //By measurement from the velocity diagram , Fig.
    8.31(b) ,
13 vR=1.6 , vBdashB=1.06 , vBdashP=1.13 , vRQ=0.4 , vQP=1.7 //m
    /s
14 PBdash=248/1000 //m
15 //Calculating the angular velocity of the link PQ
16 omegaPQ=vQP/PQ //rad/s
17 //Calculating the radial component of the
    acceleration of B with respect to A
18 arBA=omegaBA^2*AB //m/s^2
19 //Calculating the coriolis component of the
    acceleration of B with respect to coincident
    point B'
20 acBBdash=2*omegaPQ*vBdashB //m/s^2
21 //Calculating the radial component of the
    acceleration of R with respect to Q
22 arRQ=vRQ^2/QR //m/s^2
23 //Calculating the radial component of the
    acceleration of B' with respect to P
24 arBdashP=vBdashP^2/PBdash //m/s^2
25 //By measurement from the acceleration diagram , Fig.
```

```

      8.31(d) ,
26  aR=22 , aBBdash=18 //m/s^2
27  //Results:
28  printf("\n\n Velocity of the tool-box R, vR = %.1f m
      /s.\n" ,vR)
29  printf(" Acceleration of the tool-box R, aR = %d m/s
      ^2.\n" ,aR)
30  printf(" The acceleration of sliding of the block B
      along the slotted lever PQ, aBBdash = %d m/s^2.\n
      \n" ,aBBdash)

```

---

**Scilab code Exa 8.15** To find linear and angular acceleration

```

1  //To find linear and angular acceleration
2  clc
3  //Given:
4  NAO=30 //rpm
5  OA=150/1000 , OC=100/1000 , CD=125/1000 , DR=500/1000 //m
6  //Solution:
7  //Refer Fig. 8.33
8  //Calculating the angular speed of the crank OA
9  omegaAO=2*pi*NAO/60 //rad/s
10 //Calculating the velocity of A with respect to O
11 vAO=omegaAO*OA //m/s
12 vA=vAO
13 //By measurement from the velocity diagram , Fig.
      8.33(b) ,
14 vBC=0.46 , vAB=0.15 , vRD=0.12 //m/s
15 CB=240/1000 //m
16 //Calculating the angular velocity of the link BC
17 omegaBC=vBC/CB //rad/s
18 //Calculating the radial component of the
      acceleration of A with respect to O
19 arAO=vAO^2/OA //m/s^2
20 //Calculating the coriolis component of the

```

```

        acceleration of A with respect to coincident
        point B
21 acAB=2*omegaBC*vAB //m/s^2
22 //Calculating the radial component of the
        acceleration of B with respect to C
23 arBC=vBC^2/CB //m/s^2
24 //Calculating the radial component of the
        acceleration of R with respect to D
25 arRD=vRD^2/DR //m/s^2
26 //By measurement from the acceleration diagram, Fig.
        8.33(c),
27 aR=0.18, atBC=0.14 //m/s^2
28 //Calculating the angular acceleration of the
        slotted lever CA
29 alphaCA=atBC/CB //rad/s^2
30 alphaBC=alphaCA
31 //Results:
32 printf("\n\n Acceleration of the sliding block R, aR
        = %.2f m/s^2.\n", aR)
33 printf(" Angular acceleration of the slotted lever
        CA, alphaCA = %.3f rad/s^2, anticlockwise.\n\n",
        alphaCA)

```

---

**Scilab code Exa 8.16** To find linear and angular acceleration

```

1 //To find linear and angular acceleration
2 clc
3 //Given:
4 AB=125/1000 //m
5 NCO=300 //rpm
6 //Solution:
7 //Refer Fig. 8.35
8 //By measurement from the space diagram, Fig. 8.35(a
    ),
9 OC=85/1000 //m

```

```

10 //Calculating the angular velocity of the link CO
11 omegaC0=2*%pi*NC0/60 //rad/s
12 //Calculating the velocity of C with respect to O
13 vC0=omegaC0*OC //m/s
14 vC=vC0
15 //By measurement from the velocity diagram , Fig.
    8.35(b) ,
16 vBC=0.85 , vBA=2.85 , vB=vBA //m/s
17 //Calculating the radial component of of the
    acceleration of C with respect to O
18 arC0=vC0^2/OC //m/s^2
19 //Calculating the coriolis component of of
    acceleration of the piston B with respect to the
    cylinder or the coincident point C
20 acBC=2*omegaC0*vBC //m/s^2
21 //Calculating the radial component of of the
    acceleration of B with respect to A
22 arBA=vBA^2/AB //m/s^2
23 //By measurement from the acceleration diagram , Fig.
    8.35(d) ,
24 aBC=73.2 , atBA=37.6 //m/s^2
25 //Calculating the angular acceleration of the
    connecting rod AB
26 alphaAB=atBA/AB //rad/s^2
27 //Results :
28 printf("\n\n Acceleration of the piston inside the
    cylinder , aBC = %.1f m/s^2.\n" , aBC)
29 printf(" Angular acceleration of the connecting rod
    AB, alphaAB = %d rad/s^2, clockwise.\n\n" , alphaAB
    )

```

---

**Scilab code Exa 8.17** To find velocities and acceleration

```

1 //To find velocities and acceleration
2 clc

```

```

3 //Given:
4 NAO=100 //rpm
5 OA=50/1000 , AB=350/1000 , DE=250/1000 , EF=DE , CB=125/1000
   //m
6 //Solution:
7 //Refer Fig. 8.37
8 //Calculating the angular velocity of the crank AO
9 omegaAO=2*%pi*NAO/60 //rad/s
10 //Calculating the velocity of A with respect to O
11 vAO=omegaAO*OA //m/s
12 vA=vAO
13 //By measurement from the velocity diagram , Fig.
   8.37(b) ,
14 vBA=0.4 , vBC=0.485 , vB=vBC , vSD=0.265 , vQS=0.4 , vED=0.73 ,
   vFE=0.6 , vF=0.27 //m/s
15 DS=85/1000 //m
16 //Calculating the angular velocity of the link DE
17 omegaDE=vED/DE //rad/s
18 //Calculating the velocity of sliding of the link DE
   in the swivel block
19 vS=vQS //m/s
20 //Calculating the radial component of the
   acceleration of A with respect to O
21 arAO=vAO^2/OA //m/s^2
22 //Calculating the radial component of the
   acceleration of B with respect to A
23 arBA=vBA^2/AB //m/s^2
24 //Calculating the radial component of the
   acceleration of B with respect to C
25 arBC=vBC^2/CB //m/s^2
26 //Calculating the radial component of the
   acceleration of S with respect to D
27 arSD=vSD^2/DS //m/s^2
28 //Calculating the coriolis component of the
   acceleration of Q with respect to S
29 acQS=2*omegaDE*vQS //m/s^2
30 //Calculating the radial component of the
   acceleration of F with respect to E

```



```

31 arFE=vFE^2/EF //m/s^2
32 //By measurement from the acceleration diagram, Fig.
    8.37(d),
33 arQS=1.55 //m/s^2
34 //Results:
35 printf("\n\n Velocity of the slider block F, vF = %
    .2f m/s.\n",vF)
36 printf(" Angular velocity of the link DE, omegaDE =
    %.2f rad/s, anticlockwise.\n",omegaDE)
37 printf(" Velocity of sliding of the link DE in the
    swivel block, vS = %.1f m/s.\n",vS)
38 printf(" Acceleration of sliding of the link DE in
    the trunnion, arQS = %.2f m/s^2.\n\n",arQS)

```

---

# Chapter 9

## Mechanisms with Lower Pairs

**Scilab code Exa 9.1** To find inclination of track arm

```
1 //To find inclination of track arm
2 clc
3 //Given:
4 c=1.2,b=2.7 //m
5 //Solution:
6 //Calculating the inclination of the track arm to
   the longitudinal axis
7 alpha=atan(c/(2*b))*180/%pi //degrees
8 //Results:
9 printf("\\n\\n Inclination of the track arm to the
   longitudinal axis , alpha = %.1f degrees.\\n\\n",
   alpha)
```

---

**Scilab code Exa 9.3** To find the angle turned

```
1 //To find the angle turned
2 clc
3 //Given:
```

```

4 alpha=18*%pi/180 //radians
5 //Solution:
6 //Maximum velocity is possible when
7 theta1=0,theta2=180 //degrees
8 //Calculating the angle turned by the driving shaft
  when the velocity ratio is unity
9 theta3=acos(sqrt((1-cos(alpha))/(sin(alpha)^2)))
  *180/%pi //degrees
10 theta4=180-theta3 //degrees
11 //Results:
12 printf("\n\n Angle turned by the driving shaft when
  the velocity ratio is maximum, theta = %d degrees
  or %d degrees.\n",theta1,theta2)
13 printf(" Angle turned by the driving shaft when the
  velocity ratio is unity, theta = %.1f degrees or
  %.1f degrees.\n\n",theta3,theta4)

```

---

**Scilab code Exa 9.4** To find the greatest permissible angle

```

1 //To find the greatest permissible angle
2 clc
3 //Given:
4 N=500 //rpm
5 //Solution:
6 //Calculating the angular velocity of the driving
  shaft
7 omega=2*%pi*N/60 //rad/s
8 //Calculating the total fluctuation of speed of the
  driven shaft
9 q=12/100*omega //rad/s
10 //Calculating the greatest permissible angle between
  the centre lines of the shafts
11 alpha=acos((- (q/omega)+sqrt(0.12^2+4))/2)*180/%pi //
  degrees
12 //Results:

```

```
13 printf("\n\n Greatest permissible angle between the
    centre lines of the shafts , alpha = %.2f degrees
    .\n\n",alpha)
```

---

**Scilab code Exa 9.5** To find speeds and permissible angle

```
1 //To find speeds and permissible angle
2 clc
3 //Given:
4 N=1200,q=100 //rpm
5 //Solution:
6 //Calculating the greatest permissible angle between
    the centre lines of the shafts
7 alpha=acos((- (100/1200)+sqrt(0.083^2+4))/2)*180/%pi
    //degrees
8 //Calculating the maximum speed of the driven shaft
9 N1max=N/cosd(alpha) //rpm
10 //Calculating the minimum speed of the driven shaft
11 N1min=N*cosd(alpha) //rpm
12 //Results:
13 printf("\n\n Greatest permissible angle between the
    centre lines of the shafts , alpha = %.1f degrees
    .\n",alpha)
14 printf(" Maximum speed of the driven shaft , N1(max)
    = %d rpm.\n",N1max)
15 printf(" Minimum speed of the driven shaft , N1(min)
    = %d rpm.\n\n",N1min)
```

---

**Scilab code Exa 9.7** To find speeds of shafts

```
1 //To find speeds of shafts
2 clc
3 //Given:
```

```

4 alpha=20 //degrees
5 NA=500 //rpm
6 //Solution:
7 //Calculating the maximum speed of the intermediate
  shaft
8 NBmax=NA/cosd(alpha) //rpm
9 //Calculating the minimum speed of the intermediate
  shaft
10 NBmin=NA*cosd(alpha) //rpm
11 //Calculating the maximum speed of the driven shaft
12 NCmax=NBmax/cosd(alpha) //rpm
13 //Calculating the minimum speed of the driven shaft
14 NCmin=NBmin*cosd(alpha) //rpm
15 //Results:
16 printf("\n\n Maximum speed of the intermediate shaft
  , NB(max) = %.1f rad/s.\n",NBmax)
17 printf(" Minimum speed of the intermediate shaft , NB
  (min) = %.2f rad/s.\n",NBmin)
18 printf(" Maximum speed of the driven shaft , NC(max)
  = %.2f rad/s.\n",NCmax)
19 printf(" Minimum speed of the driven shaft , NC(min)
  = %.1f rad/s.\n",NCmin)

```

---

# Chapter 10

## Friction

Scilab code Exa 10.1 To find weight and coefficient of friction

```
1 //To find weight and coefficient of friction
2 clc
3 //Given:
4 theta=30 //degrees
5 P1=180 //Pulling force , N
6 P2=220 //Pushing force , N
7 //Solution:
8 //Resolving the forces horizontally for the pull of
   180N
9 F1=P1*cosd(theta) //N
10 //Resolving the forces for the push of 220 N
11 F2=P2*cosd(theta) //N
12 //Calculating the coefficient of friction
13 //For the pull of 180N,  $F1=\mu*W-90*\mu$ , or  $F1/\mu-W$ 
   =-90 .....( i)
14 //For the push of 220N,  $F2=W*\mu+110*\mu$ , or  $F2/\mu+W$ 
   =110 .....( ii)
15 A=[F1 -1; F2 -1]
16 B=[-90; 110]
17 V=A \ B
18 mu=1/V(1)
```

```

19 W=V(2)
20 //Results:
21 printf("\n\n The weight of the body, W = %d N.\n",W)
22 printf(" The coefficient of friction , mu = %.4f.\n\n
    ",mu)

```

---

**Scilab code Exa 10.2** To find weight and coefficient of friction

```

1 //To find weight and coefficient of friction
2 clc
3 //Given:
4 P1=1500,P2=1720 //N
5 alpha1=12,alpha2=15 //degrees
6 //Solution:
7 //Refer Fig. 10.10
8 //Effort applied parallel to the plane,  $P1=W*(\sin(\alpha1)+\mu*\cos(\alpha1))$ , or  $P1/W-\mu*\cos(\alpha1)=\sin(\alpha1)$  .....( i)
9 //Effort applied parallel to the plane,  $P2=W*(\sin(\alpha2)+\mu*\cos(\alpha2))$ , or  $P2/W-\mu*\cos(\alpha2)=\sin(\alpha2)$  .....( ii)
10 A=[P1 -cosd(alpha1); P2 -cosd(alpha2)]
11 B=[sind(alpha1); sind(alpha2)]
12 V=A \ B
13 W=1/V(1)
14 mu=V(2)
15 //Results:
16 printf("\n\n Coefficient of friction , mu = %.3f.\n",
    mu)
17 printf(" Weight of the body, W = %d N.\n\n",W)

```

---

**Scilab code Exa 10.3** To estimate the power

```

1 //To estimate the power
2 clc
3 //Given:
4 W=75*1000 //W
5 v=300 //mm/min
6 p=6,d0=40 //mm
7 mu=0.1
8 //Solution:
9 //Calculating the mean diameter of the screw
10 d=(d0-p/2)/1000 //m
11 //Calculating the helix angle
12 alpha=atan(p/(%pi*d*1000)) //radians
13 //Calculating the force required at the
    circumference of the screw
14 phi=atan(mu) //Limiting angle of friction , radians
15 P=W*tan(alpha+phi) //N
16 //Calculating the torque required to overcome the
    friction
17 T=P*d/2 //N-m
18 //Calculating the speed of the screw
19 N=v/p //rpm
20 //Calculating the angular speed
21 omega=2*%pi*N/60 //rad/s
22 //Calculating the power of the motor
23 Power=T*omega/1000 //Power of the motor, kW
24 //Results:
25 printf("\\n\\n Power of the motor required = %.3f kW.\\
    n\\n",Power)

```

---

Scilab code Exa 10.4 To find work done

```

1 //To find work done
2 clc
3 //Given:
4 p=12,d=40 //mm

```



```

5 mu=0.16
6 W=2500 //N
7 //Solution:
8 //Work done in drawing the wagons together against a
    steady load of 2500 N:
9 //Calculating the helix angle
10 alpha=atan(p/(%pi*d)) //radians
11 //Calculating the effort required at the
    circumference of the screw
12 phi=atan(mu) //Limiting angle of friction , radians
13 P=W*tan(alpha+phi) //N
14 //Calculating the torque required to overcome
    friction between the screw and nut
15 T=P*d/(2*1000) //N-m
16 //Calculating the number of turns required
17 N=240/(2*p)
18 //Calculating the work done
19 W1=T*2*%pi*N //Work done, N-m
20 //Work done in drawing the wagons together when the
    load increases from 2500 N to 6000 N:
21 W2=W1*(6000-2500)/2500 //Work done, N-m
22 //Results:
23 printf("\n\n Work done in drawing the wagons
    together against a steady load of 2500 N = %.1f N
    -m.\n",W1)
24 printf(" Work done in drawing the wagons together
    when the load increases from 2500 N to 6000 N = %
    .1f N-m.\n\n",W2)

```

---

**Scilab code Exa 10.5** To find the torque required

```

1 //To find the torque required
2 clc
3 //Given:
4 D=150/1000 //m

```

```

5 ps=2*10^6 //N/m^2
6 d0=50,p=6 //mm
7 mu=0.12
8 //Solution:
9 //Calculating the load on the valve
10 W=ps*pi/4*D^2 //N
11 //Calculating the mean diameter of the screw
12 d=(d0-p/2)/1000 //m
13 //Calculating the helix angle
14 alpha=atan(p/(pi*d*1000))
15 //Calculating the force required to turn the handle
16 phi=atan(mu) //Limiting angle of friction , radians
17 P=W*tan(alpha+phi) //N
18 //Calculating the torque required to turn the handle
19 T=P*d/2 //N-m
20 //Results:
21 printf("\n\n The torque required to turn the handle,
      T = %.1f N-m.\n\n",T)

```

---

**Scilab code Exa 10.6** To find force required

```

1 //To find force required
2 clc
3 //Given:
4 dc=22.5,p=5,D=50,R=D/2,l=500 //mm
5 mu=0.1,mu1=0.16
6 W=10*1000 //N
7 //Solution:
8 //Calculating the mean diameter of the screw
9 d=dc+p/2 //mm
10 //Calculating the helix angle
11 alpha=p/(pi*d) //radians
12 //Calculating the force required at the
    circumference of the screw
13 phi=atan(mu) //Limiting angle of friction , radians

```

```

14 P=W*tan(alpha+phi) //N
15 //Calculating the total torque required
16 T=P*d/2+mu1*W*R //N-mm
17 //Calculating the force required at the end of a
    spanner
18 P1=T/l //N
19 //Results:
20 printf("\n\n Force required at the end of a spanner ,
    P1 = %.2f N.\n\n",P1)

```

---

**Scilab code Exa 10.7** To find diameter of hand wheel

```

1 //To find diameter of hand wheel
2 clc
3 //Given:
4 d=50,p=12.5,D=60,R=D/2 //mm
5 W=10*1000,P1=100 //N
6 mu=0.15,mu1=0.18
7 //Solution:
8 //Calculating the helix angle
9 alpha=atan(p/(%pi*d)) //radians
10 //Calculating the tangential force required at the
    circumference of the screw
11 phi=atan(mu) //Limiting angle of friction , radians
12 P=W*tan(alpha+phi) //N
13 //Calculating the total torque required to turn the
    hand wheel
14 T=P*d/2+mu1*W*R //N-mm
15 //Calculating the diameter of the hand wheel
16 D1=T/(2*P1*1000)*2 //m
17 //Results:
18 printf("\n\n Diameter of the hand wheel , D1 = %.3f m
    .\n\n",D1)

```

---

Scilab code Exa 10.8 To find the power required

```
1 //To find the power required
2 clc
3 //Given:
4 d0=55 , D2=60 , R2=D2/2 , D1=90 , R1=D1/2 //mm
5 p=10/1000 //m
6 W=400 //N
7 mu=0.15
8 v=6 //Cutting speed , m/min
9 //Solution:
10 //Calculating the mean diameter of the screw
11 d=d0-p/2 //mm
12 //Calculating the helix angle
13 alpha=p/(%pi*d) //radians
14 //Calculating the force required at the
    circumference of the screw
15 phi=atan(mu) //Limiting angle of friction , radians
16 P=W*tan(alpha+phi) //N
17 //Calculating the mean radius of the flat surface
18 R=(R1+R2)/2 //mm
19 //Calculating the torque required
20 T=(P*d/2+mu1*W*R)/1000 //N-m
21 //Calculating the speed of the screw
22 N=v/p //rpm
23 //Calculating the angular speed
24 omega=2*%pi*N/60 //rad/s
25 //Calculating the power required to operate the nut
26 Power=T*omega/1000 //Power required to operate the
    nut , kW
27 //Results:
28 printf("\\n\\n Power required to operate the nut = %.3
    f kW.\\n\\n" ,Power)
```

---

Scilab code Exa 10.9 To find the force applied

```
1 //To find the force applied
2 clc
3 //Given:
4 d=50/1000, l=0.7 //m
5 p=10 //mm
6 mu=0.15
7 W=20*1000 //N
8 //Solution:
9 //Calculating the helix angle
10 alpha=atan(p/(%pi*d*1000)) //radians
11 //Force required to raise the load:
12 //Calculating the force required at the
    circumference of the screw
13 phi=atan(mu) //Limiting angle of friction , radians
14 P1=W*tan(alpha+phi) //N
15 //Calculating the force required at the end of the
    lever
16 P11=P1*d/(2*l) //N
17 //Calculating the force required at the
    circumference of the screw
18 P2=W*(phi-alpha) //N
19 //Force required to lower the load:
20 //Calculating the force required at the end of the
    lever
21 P21=P2*d/(2*l) //N
22 //Results:
23 printf("\\n\\n Force required at the end of the lever
    to raise the load , P1 = %d N.\\n",P11)
24 printf(" Force required at the end of the lever to
    lower the load , P1 = %d N.\\n\\n",P21)
```

---

**Scilab code Exa 10.10** To find ratio of torques and efficiency

```
1 //To find ratio of torques and efficiency
2 clc
3 //Given:
4 d=50,p=12.5 //mm
5 mu=0.13
6 W=25*1000 //N
7 //Solution:
8 //Calculating the helix angle
9 alpha=atan(p/(%pi*d)) //radians
10 //Calculating the force required on the screw to
    raise the load
11 phi=atan(mu) //Limiting angle of friction , radians
12 P1=W*(alpha+phi) //N
13 //Calculating the torque required on the screw to
    raise the load
14 T1=P1*d/2 //N-mm
15 //Calculating the force required on the screw to
    lower the load
16 P2=W*tan(phi-alpha) //N
17 //Calculating the torque required to lower the load
18 T2=P2*d/2 //N
19 //Calculating the ratio of the torques required
20 r=T1/T2 //Ratio of the torques required , N-mm
21 //Calculating the efficiency of the machine
22 eta=tan(alpha)/tan(alpha+phi)*100 //%
23 //Results:
24 printf("\n\n Torque required on the screw to raise
    the load , T1 = %d N-mm.\n",T1)
25 printf(" Ratio of the torque required to raise the
    load to the torque required to lower the load = %
    .1f.\n",r)
26 printf(" Efficiency of the machine, eta = %.1f %c.\n
```

\n", eta, "%")

---

**Scilab code Exa 10.11** To find work done and efficiency

```
1 //To find work done and efficiency
2 clc
3 //Given:
4 p=10, d=50, D2=60, R2=D2/2, D1=10, R1=D1/2 //mm
5 W=20*1000 //N
6 mu=0.08, mu1=mu
7 //Solution:
8 //Calculating the helix angle
9 alpha=atan(p/(%pi*d)) //radians
10 //Calculating the force required at the
    circumference of the screw to lift the load
11 phi=atan(mu) //Limiting angle of friction, radians
12 P=W*tan(alpha+phi) //N
13 //Calculating the torque required to overcome
    friction at the screw
14 T=P*d/(2*1000) //N-m
15 //Calculating the number of rotations made by the
    screw
16 N=170/p
17 //When the load rotates with the screw:
18 //Calculating the work done in lifting the load
19 W1=T*2*%pi*N //Work done in lifting the load, N-m
20 //Calculating the efficiency of the screw jack
21 eta1=tan(alpha)/tan(alpha+phi)*100 //%
22 //When the load does not rotate with the screw:
23 //Calculating the mean radius of the bearing surface
24 R=(R1+R2)/2 //mm
25 //Calculating the torque required to overcome
    friction at the screw and the collar
26 T=(P*d/2+mu1*W*R)/1000 //N-m
27 //Calculating the work done by the torque in lifting
```

```

    the load
28 W2=T*2*%pi*N //Work done by the torque in lifting
    the load , N-m
29 //Calculating the torque required to lift the load ,
    neglecting friction
30 T0=(W*tan(alpha)*d/2)/1000 //N-m
31 //Calculating the efficiency of the screw jack
32 eta2=T0/T*100 //%
33 //Results:
34 printf("\n\n When the load rotates with the screw ,
    work done in lifting the load = %d N-m.\n",W1)
35 printf(" Efficiency of the screw jack , eta = %.1f %c
    .\n",eta1,"%")
36 printf(" When the load does not rotate with the
    screw , work done in lifting the load = %d N-m.\n"
    ,W2)
37 printf(" Efficiency of the screw jack , eta = %.1f %c
    .\n\n",eta2,"%")

```

---

**Scilab code Exa 10.12** To find length of lever

```

1 //To find length of lever
2 clc
3 //Given:
4 W=10*1000,P1=100 //N
5 p=12,d=50 //mm
6 mu=0.15
7 //Solution:
8 //Calculating the helix angle
9 alpha=atan(p/(%pi*d)) //radians
10 //Calculating the effort required at the
    circumference of the screw to raise the load
11 phi=atan(mu) //Limiting angle of friction , radians
12 P=W*tan(alpha+phi) //N
13 //Calculating the torque required to overcome

```



```

friction
14 T=P*d/2 //N-mm
15 //Calculating the length of the lever
16 l=T/P1 //mm
17 //Calculating the mechanical advantage
18 MA=W/P1
19 //Calculating the efficiency of the screw jack
20 eta=tan(alpha)/tan(alpha+phi)*100 //%
21 //Results:
22 printf("\n\n The length of the lever to be used, l =
    %.1f mm.\n",l)
23 printf(" Mechanical advantage obtained, M.A. = %d.\n
    ",MA)
24 if eta<50 then
25     printf(" The screw is a self locking screw.\n\n"
    );
26 else
27     printf(" The screw is not a self locking screw."
    );
28 end

```

---

**Scilab code Exa 10.13** To find the torque required

```

1 //To find the torque required
2 clc
3 //Given:
4 d=22,p=3 //mm
5 funcprot(0)
6 beta=60/2 //degrees
7 W=40*1000 //N
8 mu=0.15
9 //Solution:
10 //Calculating the helix angle
11 alpha=atan(p/(%pi*d)) //radians
12 //Calculating the virtual coefficient of friction

```

```

13 mu1=mu/cosd(beta)
14 //Calculating the force required at the
    circumference of the screw
15 phi1=atan(mu1) //Virtual limiting angle of friction ,
    radians
16 P=W*tan(alpha+phi1)
17 //Calculating the torque on one rod
18 T=P*d/(2*1000) //N-m
19 //Calculating the torque required on the nut
20 T1=2*T //N-m
21 //Results:
22 printf("\n\n The torque required on the nut, T1 = %
    .2f N-m.\n\n",T1)

```

---

**Scilab code Exa 10.14** To find the forcer

```

1 //To find the forcer
2 clc
3 //Given:
4 d=25 , p=5 , R=25 //mm
5 funcprot(0)
6 beta=27.5 //degrees
7 mu=0.1 , mu2=0.16
8 l=0.5 //m
9 W=10*1000 //N
10 //Solution:
11 //Calculating the virtual coefficient of friction
12 mu1=mu/cosd(beta)
13 //Calculating the helix angle
14 alpha=atan(p/(%pi*d)) //radians
15 //Calculating the force on the screw
16 phi1=atan(mu1) //Virtual limiting angle of frcition ,
    radians
17 P=W*tan(alpha+phi1) //N
18 //Calculating the total torque transmitted

```

```

19 T=(P*d/2+mu2*W*R)/1000 //N-m
20 //Calculating the force required at the end of a
    spanner
21 P1=T/l //N
22 //Results:
23 printf("\n\n Force required at the end of a spanner ,
    P1 = %.1f N.\n\n",P1)

```

---

**Scilab code Exa 10.15** To find power transmitted

```

1 //To find power transmitted
2 clc
3 //Given:
4 d=60,r=d/2 //mm
5 W=2000 //N
6 mu=0.03
7 N=1440 //rpm
8 //Solution:
9 //Calculating the angular speed of the shaft
10 omega=2*%pi*N/60 //rad/s
11 //Calculating the torque transmitted
12 T=mu*W*(r/1000) //N-m
13 //Calculating the power transmitted
14 P=T*omega //W
15 //Results:
16 printf("\n\n The power transmitted , P = %.1f W.\n\n"
    ,P)

```

---

**Scilab code Exa 10.16** To estimate power lost in friction

```

1 //To estimate power lost in friction
2 clc
3 //Given:

```

```

4 D=150/1000 ,R=D/2 //m
5 N=100 //rpm
6 W=20*1000 //N
7 mu=0.05
8 //Solution:
9 //Calculating the angular speed of the shaft
10 omega=2*%pi*N/60 //rad/s
11 //Calculating the total frictional torque for
    uniform pressure distribution
12 T=2/3*mu*W*R //N-m
13 //Calculating the power lost in friction
14 P=T*omega //W
15 //Results:
16 printf("\n\n Power lost in friction , P = %.1f W.\n\n
    ",P)

```

---

**Scilab code Exa 10.17** To find power absorbed in friction

```

1 //To find power absorbed in friction
2 clc
3 //Given:
4 W=20*1000 //N
5 alpha=120/2 //degrees
6 Pn=0.3 //N/mm^2
7 N=200 //rpm
8 mu=0.1
9 //Solution:
10 //Calculating the angular speed of the shaft
11 omega=2*%pi*N/60 //rad/s
12 //Calculating the inner radius of the bearing
    surface
13 r2=sqrt(W/(3*%pi*Pn)) //mm
14 //Calculating the outer radius of the bearing
    surface
15 r1=2*r2 //mm

```

```

16 //Calculating the total frictional torque assuming
    uniform pressure
17 T=2/3*mu*W*(1/sind(alpha))*(r1^3-r2^3)/(r1^2-r2^2)
    /1000 //N-m
18 //Calculating the power absorbed in friction
19 P=T*omega/1000 //kW
20 //Results:
21 printf("\n\n Power absorbed in friction , P = %.3f kW
    .\n\n",P)

```

---

**Scilab code Exa 10.18** To find power lost in friction

```

1 //To find power lost in friction
2 clc
3 //Given:
4 D=200/1000 ,R=D/2 //m
5 W=30*1000 //N
6 alpha=120/2 //degrees
7 mu=0.025
8 N=140 //rpm
9 //Solution:
10 //Calculating the angular speed of the shaft
11 omega=2*pi*N/60 //rad/s
12 //Power lost in friction assuming uniform pressure:
13 //Calculating the total frictional torque
14 T=2/3*mu*W*R*(1/sind(alpha)) //N-m
15 //Calculating the power lost in friction
16 P1=T*omega //Power lost in friction , W
17 //Power lost in friction assuming uniform wear:
18 //Calculating the total frictional torque
19 T=1/2*mu*W*R*(1/sind(alpha)) //N-m
20 //Calculating the power lost in friction
21 P2=T*omega //Power lost in friction , W
22 //Results:
23 printf("\n\n Power lost in friction assuming uniform

```

```

        pressure , P = %d W.\n",P1)
24 printf(" Power lost in friction assuming uniform
        wear , P = %.1f W.\n\n",P2)

```

---

**Scilab code Exa 10.19** To find power absorbed in friction

```

1 //To find power absorbed in friction
2 clc
3 //Given:
4 n=6
5 d1=600 ,r1=d1/2 ,d2=300 ,r2=d2/2 //mm
6 W=100*1000 //N
7 mu=0.12
8 N=90 //rpm
9 //Solution:
10 //Calculating the angular speed of the engine
11 omega=2*%pi*N/60 //rad/s
12 //Power absorbed in friction assuming uniform
    pressure:
13 //Calculating the total frictional torque
    transmitted
14 T=2/3*mu*W*(r1^3-r2^3)/(r1^2-r2^2)/1000 //N-m
15 //Calculating the power absorbed in friction
16 P1=T*omega/1000 //Power absorbed in friction
    assuming uniform pressure , kW
17 //Power absorbed in friction assuming uniform wear:
18 //Calculating the total frictional torque
    transmitted
19 T=1/2*mu*W*(r1+r2)/1000 //N-m
20 //Calculating the power absorbed in friction
21 P2=T*omega/1000 //Power absorbed in friction
    assuming uniform wear , kW
22 //Results:
23 printf("\n\n Power absorbed in friction assuming
    uniform pressure , P = %.1f kW.\n",P1)

```

```
24 printf(" Power absorbed in friction assuming uniform
wear , P = %.2 f kW.\n\n",P2)
```

---

**Scilab code Exa 10.20** To find power absorbed

```
1 //To find power absorbed
2 clc
3 //Given:
4 d1=400 , r1=d1/2 , d2=250 , r2=d2/2 //mm
5 p=0.35 //N/mm^2
6 mu=0.05
7 N=105 //rpm
8 W=150*1000 //N
9 //Solution:
10 //Calculating the angular speed of the shaft
11 omega=2*pi*N/60 //rad/s
12 //Calculating the total frictional torque
transmitted for uniform pressure
13 T=2/3*mu*W*(r1^3-r2^3)/(r1^2-r2^2)/1000 //N-m
14 //Calculating the power absorbed
15 P=T*omega/1000 //kW
16 //Calculating the number of collars required
17 n=W/(p*pi*(r1^2-r2^2))
18 //Results:
19 printf("\n\n Power absorbed , P = %.2 f kW.\n",P)
20 printf(" Number of collars required , n = %d.\n\n",n
+1)
```

---

**Scilab code Exa 10.21** To find diameter and number of collars

```
1 //To find diameter and number of collars
2 clc
3 //Given:
```

```

4 d2=300/1000 , r2=d2/2 //m
5 W=200*1000 //N
6 N=75 //rpm
7 mu=0.05
8 p=0.3 //N/mm^2
9 P=16*1000 //W
10 //Solution:
11 //Calculating the angular velocity of the shaft
12 omega=2*pi*N/60 //rad/s
13 //Calculating the total frictional torque
    transmitted
14 T=P/omega //N-m
15 //Calculating the external diameter of the collar
16 //We have ,  $T = \frac{2}{3} \mu W (r_1^3 - r_2^3) / (r_1^2 - r_2^2)$  , or
     $(2 \mu W) r_1^2 - (3 T - 2 \mu W r_2) r_1 + (2 \mu W r_2^2 - 3 T r_2) = 0$ 
17 A=2*mu*W , B=-(3*T-2*mu*W*r2) , C=2*mu*W*r2^2-3*T*r2
18 r1=(-B+sqrt(B^2-4*A*C))/(2*A)*1000 //mm
19 d1=2*r1 //mm
20 //Calculating the number of collars
21 n=W/(p*pi*(r1^2-(r2*1000)^2))
22 //Results:
23 printf("\n\n External diameter of the collar , d1 =
    %d mm.\n" , d1)
24 printf(" Number of collars , n = %d.\n\n" , n+1)

```

---

**Scilab code Exa 10.22** To find the pressure

```

1 //To find the pressure
2 clc
3 //Given:
4 W=4*1000 //N
5 r2=50 , r1=100 //mm
6 //Solution:
7 //Calculating the maximum pressure

```



```

8 pmax=W/(2*pi*r2*(r1-r2)) //N/mm^2
9 //Calculating the minimum pressure
10 pmin=W/(2*pi*r1*(r1-r2)) //N/mm^2
11 //Calculating the average pressure
12 pav=W/(pi*(r1^2-r2^2)) //N/mm^2
13 //Results:
14 printf("\n\n Maximum pressure , pmax = %.4 f N/mm^2.\n
      ", pmax)
15 printf(" Minimum pressure , pmin = %.4 f N/mm^2.\n",
      pmin)
16 printf(" Average pressure , pav = %.2 f N/mm^2.\n\n",
      pav)

```

---

**Scilab code Exa 10.23** To find power transmitted

```

1 //To find power transmitted
2 clc
3 //Given:
4 d1=300, r1=d1/2, d2=200, r2=d2/2 //mm
5 p=0.1 //N/mm^2
6 mu=0.3
7 N=2500 //rpm
8 n=2
9 //Solution:
10 //Calculating the radial speed of the clutch
11 omega=2*pi*N/60 //rad/s
12 //Calculating the intensity of pressure
13 C=p*r2 //N/mm
14 //Calculating the axial thrust
15 W=2*pi*C*(r1-r2) //N
16 //Calculating the mean radius of the friction
      surfaces for uniform wear
17 R=(r1+r2)/(2*1000) //m
18 //Calculating the torque transmitted
19 T=n*mu*W*R //N-m

```

```

20 //Calculating the power transmitted by a clutch
21 P=T*omega/1000 //kW
22 //Results:
23 printf("\n\n Power transmitted by a clutch , P = %.3 f
      kW.\n\n",P)

```

---

**Scilab code Exa 10.24** To find radii and axial thrust

```

1 //To find radii and axial thrust
2 clc
3 //Given:
4 n=2, mu=0.255
5 P=25*1000 //W
6 N=3000 //rpm
7 r=1.25 //Ratio of radii , r1/r2
8 p=0.1 //N/mm^2
9 //Solution:
10 //Calculating the angular speed of the clutch
11 omega = 2*pi*N/60 //rad/s
12 //Calculating the torque transmitted
13 T=P/omega*1000 //N-mm
14 //Calculating the inner radius
15 r2=(T/(n*mu*2*pi*0.1*(1.25-1)*(1.25+1)/2))^(1/3) //
      mm
16 //Calculating the outer radius
17 r1=r*r2 //mm
18 //Calculating the axial thrust to be provided by
      springs
19 C=0.1*r2 //Intensity of pressure , N/mm
20 W=2*pi*C*(r1-r2) //N
21 //Results:
22 printf("\n\n Outer radius of the frictional surface ,
      r1 = %d mm.\n",r1)
23 printf(" Inner radius of the frictional surface , r2
      = %d mm.\n",r2)

```

```

24 printf(" Axial thrust to be provided by springs , W =
      %d N.\n\n",W)

```

---

**Scilab code Exa 10.25** To find dimensions of clutch plate

```

1 //To find dimensions of clutch plate
2 clc
3 //Given:
4 P=7.5*1000 //W
5 N=900 //rpm
6 p=0.07 //N/mm^2
7 mu=0.25
8 n=2
9 //Solution:
10 //Calculating the angular speed of the clutch
11 omega=2*pi*N/60 //rad/s
12 //Calculating the torque transmitted
13 T=P/omega*1000 //N-mm
14 //Calculating the mean radius of the friction lining
15 R=(T/(pi/2*n*mu*p))^(1/3) //mm
16 //Calculating the face width of the friction lining
17 w=R/4 //mm
18 //Calculating the outer and inner radii of the
    clutch plate
19 //We have,  $w = r_1 - r_2$ , or  $r_1 - r_2 = w$ 
    .....( i)
20 //Also,  $R = (r_1 + r_2)/2$ , or  $r_1 + r_2 = 2*R$ 
    .....( ii)

21 A=[1 -1; 1 1]
22 B=[w; 2*R]
23 V=A \ B
24 r1=V(1)
25 r2=V(2)
26 //Results:
27 printf("\n\n Mean radius of the friction lining , R =

```

```

    %d mm.\n",R)
28 printf(" Face width of the friction lining , w = %.2 f
    mm.\n",w)
29 printf(" Outer radius of the clutch plate , r1 = %.3 f
    mm.\n",r1)
30 printf(" Inner radius of the clutch plate , r2 = %.3 f
    mm.\n\n",r2)

```

---

**Scilab code Exa 10.26** To find dimensions of clutch plate

```

1 //To find dimensions of clutch plate
2 clc
3 //Given:
4 P=100 //kW
5 N=2400 //rpm
6 T=500*1000 //N-mm
7 p=0.07 //N/mm^2
8 mu=0.3
9 Ns=8 //Number of springs
10 k=40 //Stiffness , N/mm
11 n=2
12 //Solution:
13 //Calculating the inner radius of the friction plate
14 r2=(T/(n*mu*2*pi*p*(1.25-1)*(1.25+1)/2))^(1/3) //mm
15 //Calculating the outer radius of the friction plate
16 r1=1.25*r2 //mm
17 //Calculating the total stiffness of the springs
18 s=k*Ns //N/mm
19 //Calculating the intensity of pressure
20 C=p*r2 //N/mm
21 //Calculating the axial force required to engage the
    clutch
22 W=2*pi*C*(r1-r2) //N
23 //Calculating the initial compression in the springs
24 IC=W/s //Initial compression in the springs , mm

```

```

25 //Results:
26 printf("\n\n Outer radius of the friction plate , r1
    = %.1f mm.\n",r1)
27 printf(" Inner radius of the friction plate , r2 = %d
    mm.\n",r2)
28 printf(" Initial compression in the springs = %.1f
    mm.\n\n",IC)

```

---

**Scilab code Exa 10.27** To find speed time and KE lost

```

1 //To find speed , time and KE lost
2 clc
3 //Given:
4 d1=220 , r1=d1/2 , d2=160 , r2=d2/2 //mm
5 W=570 //N
6 m1=800 , m2=1300 //kg
7 k1=200/1000 , k2=180/1000 //m
8 mu=0.35
9 N1=1250 //rpm
10 n=2
11 //Solution:
12 //Calculating the initial angular speed of the motor
    shaft
13 omega1=2*pi*N1/60 //rad/s
14 //Calculating the moment of inertia for the motor
    armature and shaft
15 I1=m1*k1^2 //kg-m^2
16 //Calculating the moment of inertia for the rotor
17 I2=m2*k2^2 //kg-m^2
18 //Calculating the final speed of the motor and rotor
19 omega2=0
20 omega3=(I1*omega1+I2*omega2)/(I1+I2) //rad/s
21 //Calculating the mean radius of the friction plate
22 R=(r1+r2)/(2*1000) //m
23 //Calculating the frictional torque

```

```

24 T=n*mu*W*R //N-m
25 //Calculating the angular acceleration of the rotor
26 alpha2=T/I2 //rad/s^2
27 //Calculating the time to reach the speed of omega3
28 omegaF=omega3, omegaI=omega2
29 t=(omegaF-omegaI)/alpha2 //seconds
30 //Calculating the angular kinetic energy before
    impact
31 E1=1/2*I1*omega1^2+1/2*I2*omega2^2 //N-m
32 //Calculating the angular kinetic energy after
    impact
33 E2=1/2*(I1+I2)*omega3^2 //N-m
34 //Calculating the kinetic energy lost during the
    period of slipping
35 E=E1-E2 //N-m
36 //Calculating the torque on armature shaft
37 T1=-60-T //N-m
38 //Calculating the torque on rotor shaft
39 T2=T //N-m
40 //Calculating the time of slipping assuming constant
    resisting torque:
41 //Considering armature shaft, omega3 = omega1+alpha1
    *t1, or omega3-(T1/I1)*t1 = omega1 .....( i
    )
42 //Considering rotor shaft, omega3 = alpha2*t1, or
    omega3-(T2/I2)*t1 = 0
    .....( ii)
43 A=[1 -T1/I1; 1 -T2/I2]
44 B=[omega1; 0]
45 V=A \ B
46 t11=V(2) //Time of slipping assuming constant
    resisting torque, seconds
47 //Calculating the time of slipping assuming constant
    driving torque:
48 //Calculating the torque on armature shaft
49 T1=60-T //N-m
50 t12=(omega2-omega1)/(T1/I1-T2/I2) //Time of slipping
    assuming constant driving torque, seconds

```

```

51 //Results :
52 printf("\n\n Final speed of the motor and rotor ,
        omega3 = %.2f rad/s.\n",omega3)
53 printf(" Time to reach the speed of %.2f rad/s, t =
        %.1f s.\n",omega3,t)
54 printf(" Kinetic energy lost during the period of
        slipping = %d N-m.\n",E)
55 printf(" Time of slipping assuming constant
        resisting torque, t1 = %.1f s.\n",t11)
56 printf(" Time of slipping assuming constant driving
        torque, t1 = %d s.\n\n",t12)

```

---

**Scilab code Exa 10.28** To find the power transmitted

```

1 //To find the power transmitted
2 clc
3 //Given:
4 n=4, mu=0.3
5 p=0.127 //N/mm2
6 N=500 //rpm
7 r1=125, r2=75 //mm
8 //Solution:
9 //Calculating the angular speed of the clutch
10 omega=2*pi*N/60 //rad/s
11 //Calculating the maximum intensity of pressure
12 C=p*r2 //N/mm
13 //Calculating the axial force required to engage the
    clutch
14 W=2*pi*C*(r1-r2) //N
15 //Calculating the mean radius of the friction
    surfaces
16 R=(r1+r2)/(2*1000) //m
17 //Calculating the torque transmitted
18 T=n*mu*W*R //N-m
19 //Calculating the power transmitted

```

```

20 P=T*omega/1000 //kW
21 //Results:
22 printf("\n\n Power transmitted , P = %.1f kW.\n\n",P)

```

---

**Scilab code Exa 10.29** To find maximum intensity of pressure

```

1 //To find maximum intensity of pressure
2 clc
3 //Given:
4 n1=3, n2=2, mu=0.3
5 d1=240, r1=d1/2, d2=120, r2=d2/2 //mm
6 P=25*1000 //W
7 N=1575 //rpm
8 //Solution:
9 //Calculating the angular speed of the shaft
10 omega=2*pi*N/60 //rad/s
11 //Calculating the torque transmitted
12 T=P/omega //N-m
13 //Calculating the number of pairs of friction
    surfaces
14 n=n1+n2-1
15 //Calculating the mean radius of friction surfaces
    for uniform wear
16 R=(r1+r2)/(2*1000) //m
17 //Calculating the axial force on each friction
    surface
18 W=T/(n*mu*R) //N
19 //Calculating the maximum axial intensity of
    pressure
20 p=W/(2*pi*r2*(r1-r2)) //N/mm^2
21 //Results:
22 printf("\n\n Maximum axial intensity of pressure , p
    = %.3f N/mm^2.\n\n",p)

```

---



**Scilab code Exa 10.30** To find maximum power transmitted

```
1 //To find maximum power transmitted
2 clc
3 //Given:
4 n1=3, n2=2, n=4, mu=0.3
5 d1=240, r1=d1/2, d2=120, r2=d2/2 //mm
6 P=25*1000 //W
7 N=1575 //rpm
8 //Solution:
9 //Calculating the angular speed of the shaft
10 omega=2*%pi*N/60 //rad/s
11 //Calculating the torque transmitted
12 T=P/omega //N-m
13 //Calculating the mean radius of the contact surface
    , for uniform pressure
14 R=2/3*(r1^3-r2^3)/(r1^2-r2^2)/1000 //m
15 //Calculating the total spring load
16 W1=T/(n*mu*R) //N
17 //Calculating the maximum power transmitted:
18 //Given:
19 ns=6 //Number of springs
20 c=8 //Contact surfaces of the spring
21 w=1.25 //Wear on each contact surface, mm
22 k=13*1000 //Stiffness of each spring, N/m
23 //Calculating the total wear
24 Tw=c*w/1000 //Total wear, m
25 //Calculating the reduction in spring force
26 Rs=Tw*k*ns //N
27 //Calculating the new axial load
28 W2=W1-Rs //N
29 //Calculating the mean radius of the contact
    surfaces for uniform wear
30 R=(r1+r2)/(2*1000) //m
```

```

31 //Calculating the torque transmitted
32 T=n*mu*W2*R //N-m
33 //Calculating the maximum power transmitted
34 P=T*omega/1000 //kw
35 //Results:
36 printf("\n\n Total spring load , W = %d N.\n",W1)
37 printf(" Maximum power that can be transmitted , P =
    %.2 f kW.\n\n",P)

```

---

**Scilab code Exa 10.31** To find dimensions and axial load

```

1 //To find dimensions and axial load
2 clc
3 //Given:
4 P=90*1000 //W
5 N=1500 //rpm
6 alpha=20 //degrees
7 mu=0.2
8 D=375, R=D/2 //mm
9 pn=0.25 //N/mm^2
10 //SOLution:
11 //Calculating the angular speed of the clutch
12 omega=2*pi*N/60 //rad/s
13 //Calculating the torque transmitted
14 T=P/omega*1000 //N-mm
15 //Calculating the width of the bearing surface
16 b=T/(2*pi*mu*pn*R^2) //mm
17 //Calculating the external and internal radii of the
    bearing surface
18 //We know that , r1+r2 = 2*R, and r1-r2 = b*sind(
    alpha)
19 A=[1 1; 1 -1]
20 B=[2*R; b*sind(alpha)]
21 V=A \ B
22 r1=V(1) //mm

```

```

23 r2=V(2) //mm
24 //Calculating the intensity of pressure
25 C=pn*r2 //N/mm
26 //Calculating the axial load required
27 W=2*pi*C*(r1-r2) //N
28 //Results:
29 printf("\n\n Width of the bearing surface , b = %.1f
      mm.\n",b)
30 printf(" External radius of the bearing surface , r1
      = %.1f mm.\n",r1)
31 printf(" Internal radius of the bearing surface , r2
      = %.1f mm.\n",r2)
32 printf(" Axial load required , W = %d N.\n\n",W)

```

---

**Scilab code Exa 10.32** To find axial force and face width

```

1 //To find axial force and face width
2 clc
3 //Given:
4 P=45*1000 //W
5 N=1000 //rpm
6 alpha=12.5 //degrees
7 D=500/1000 , R=D/2 //m
8 mu=0.2
9 pn=0.1 //N/mm^2
10 //Solution:
11 //Calculating the angular speed of the shaft
12 omega=2*pi*N/60 //rad/s
13 //Calculating the torque developed by the clutch
14 T=P/omega //N-m
15 //Calculating the normal load acting on the friction
      surface
16 Wn=T/(mu*R) //N
17 //Calculating the axial spring force necessary to
      engage the clutch

```

```

18 We=Wn*(sind(alpha)+mu*cosd(alpha)) //N
19 //Calculating the face width required
20 b=Wn/(pn*2*%pi*R*1000) //mm
21 //Results:
22 printf("\n\n Axial force necessary to engage the
        clutch , We = %d N.\n",We)
23 printf(" Face width required , b = %.1f mm.\n\n",b)

```

---

**Scilab code Exa 10.33** To find dimensions of contact surfaces

```

1 //To find dimensions of contact surfaces
2 clc
3 //Given:
4 alpha=30/2 //degrees
5 pn=0.35 //N/mm^2
6 P=22.5*1000 //W
7 N=2000 //rpm
8 mu=0.15
9 //Solution:
10 //Calculating the angular speed of the clutch
11 omega=2*%pi*N/60 //rad/s
12 //Calculating the torque transmitted by the clutch
13 T=P/omega*1000 //N-mm
14 //Calculating the mean radius of the contact surface
15 R=(T/(2*%pi*mu*pn/3))^(1/3) //mm
16 //Calculating the face width of the contact surface
17 b=R/3
18 //Calculating the outer and inner radii of the
        contact surface
19 //Refer Fig. 10.27
20 //We have,  $r_1 - r_2 = b \cdot \sin(\alpha)$ , and  $r_1 + r_2 = 2 \cdot R$ 
21 A=[1 -1; 1 1]
22 B=[b*sind(alpha); 2*R]
23 V=A \ B
24 r1=V(1) //mm

```

```

25 r2=V(2) //mm
26 //Results:
27 printf("\n\n Mean radius of the contact surface , R =
      %d mm.\n",R)
28 printf(" Outer radius of the contact surface , r1 = %
      .2 f mm.\n",r1)
29 printf(" Inner radius of the contact surface , r2 = %
      .2 f mm.\n\n",r2)

```

---

**Scilab code Exa 10.34** To find time required and energy lost

```

1 //To find time required and energy lost
2 clc
3 //Given:
4 D=75/1000, R=D/2 //m
5 alpha=15 //degrees
6 mu=0.3
7 W=180 //N
8 NF=1000 //rpm
9 m=13.5 //kg
10 k=150/1000 //m
11 //Solution:
12 //Calculating the angular speed of the flywheel
13 omegaF=2*pi*NF/60 //rad/s
14 //Calculating the torque required to produce
      slipping
15 T=mu*W*R*(1/sind(alpha)) //N-m
16 //Calculating the mass moment of inertia of the
      flywheel
17 IF=m*k^2 //kg-m^2
18 //Calculating the angular acceleration of the
      flywheel
19 alphaF=T/IF //rad/s^2
20 //Calculating the time required for the flywheel to
      attain full speed

```

```

21 tF=omegaF/alphaF //seconds
22 //Calculating the angle turned through by the motor
    and flywheel in time tF
23 theta=1/2*omegaF*tF //rad
24 //Calculating the energy lost in slipping of the
    clutch
25 E=T*theta //Energy lost in slipping of the clutch , N
    -m
26 //Results:
27 printf("\n\n Torque required to produce slipping , T
    = %.1f N-m.\n",T)
28 printf(" Time required for the flywheel to attain
    full speed , tF = %.1f s.\n",tF)
29 printf(" Energy lost in slipping of the clutch = %d
    N-m.\n\n",E)

```

---

**Scilab code Exa 10.35** To find mass and size of shoes

```

1 //To find mass and size of shoes
2 clc
3 //Given:
4 P=15*1000 //W
5 N=900 //rpm
6 n=4, mu=0.25
7 R=150/1000, r=120/1000 //m
8 theta=60 //degrees
9 p=0.1 //N/mm^2
10 //Solution:
11 //Calculating the angular speed of the clutch
12 omega=2*%pi*N/60 //rad/s
13 //Calculating the speed at which the engagement
    begins
14 omega1=3/4*omega //rad/s
15 //Calculating the torque transmitted at the running
    speed

```

```

16 T=P/omega //N-m
17 //Calculating the mass of the shoes
18 m=T/(n*mu*(omega^2*r-omega1^2*r)*R) //kg
19 //Calculating the contact length of shoes
20 l=(theta*pi/180)*R*1000 //mm
21 //Calculating the centrifugal force acting on each
    shoe
22 Pc=m*omega^2*r //N
23 //Calculating the inward force on each shoe exerted
    by the spring
24 Ps=m*omega1^2*r //N
25 //Calculating the width of the shoes
26 b=(Pc-Ps)/(l*p) //mm
27 //Results:
28 printf("\n\n Mass of the shoes , m = %.2 f kg.\n",m)
29 printf(" Width of the shoes , b = %.1 f mm.\n\n",b)

```

---

**Scilab code Exa 10.36** To find power transmitted

```

1 //To find power transmitted
2 clc
3 //Given:
4 n=4, mu=0.3
5 c=5, r=160 //mm
6 S=500 //N
7 D=400/1000, R=D/2 //m
8 m=8 //kg
9 s=50 //N/mm
10 N=500 //rpm
11 //Solution:
12 //Calculating the angular speed of the clutch
13 omega=2*pi*N/60 //rad/s
14 //Calculating the operating radius
15 r1=(r+c)/1000 //m
16 //Calculating the centrifugal force on each shoe

```

```
17 Pc=m*omega^2*r1 //N
18 //Calculating the inward force exerted by the spring
19 Ps=S+c*s //N
20 //Calculating the frictional force acting
    tangentially on each shoe
21 F=mu*(Pc-Ps) //N
22 //Calculating the total frictional torque
    transmitted by the clutch
23 T=n*F*R //N-m
24 //Calculating the power transmitted
25 P=T*omega/1000 //kW
26 //Results:
27 printf("\n\n Power transmitted , P = %.1f kW.\n\n",P)
```

---



# Chapter 11

## Belt Rope and Chain Drives

Scilab code Exa 11.1 To find speed of shaft

```
1 //To find speed of shaft
2 clc
3 //Given:
4 N1=150 //rpm
5 d1=750, d2=450, d3=900, d4=150 //mm
6 //Solution:
7 //Calculating the speed of the dynamo shaft when
   there is no slip
8 N41=N1*(d1*d3)/(d2*d4) //rpm
9 //Calculating the speed of the dynamo shaft whne
   there is a slip of 2% at each drive
10 s1=2, s2=2 //%
11 N42=N1*(d1*d3)/(d2*d4)*(1-s1/100)*(1-s2/100) //rpm
12 //Results:
13 printf("\\n\\n Speed of the dynamo shaft when there is
   no slip , N4 = %d rpm.\\n\\n",N41)
14 printf(" Speed of the dynamo shaft when there is a
   slip of 2%c at each drive , N4 = %d rpm.\\n\\n", "%",
   N42)
```

---

**Scilab code Exa 11.2** To find speed lost

```
1 //To find speed lost
2 clc
3 //Given:
4 d1=1, d2=2.25 //m
5 N1=200 //rpm
6 sigma1=1.4*10^6, sigma2=0.5*10^6, E=100*10^6 //N/m^2
7 //Solution:
8 //Calculating the speed of the driven pulley
9 N21=N1*(d1/d2) //rpm
10 //Calculating the speed of the shaft considering
    creep
11 N22=N1*(d1/d2)*(E+sqrt(sigma2))/(E+sqrt(sigma1)) //
    rpm
12 //Calculating the speed lost by the driven pulley
    due to creep
13 N1=N21-N22 //Speed lost by the driven pulley due to
    creep , rpm
14 //Results:
15 printf("\\n\\n Speed lost by the driven pulley due to
    creep = %.4f rpm.\\n\\n",N1)
```

---

**Scilab code Exa 11.3** To find radii of stepped pulleys

```
1 //To find radii of stepped pulleys
2 clc
3 //Given:
4 N1=160, N3=N1, N5=N3, N2=60, N4=80, N6=100 //rpm
5 x=720, r1=40 //mm
6 //Solution:
7 //For a crossed belt:
```

```

8 //Calculating the radii of pulleys 2, 3, 4, 5, and 6
9 r2=r1*(N1/N2) //mm
10 //For pulleys 3 and 4,  $r4 = r3*(N3/N4)$ , or  $r3*(N3/N4)$ 
     $-r4 = 0$ 
11 //For a crossed belt drive,  $r3+r4 = r1+r2$ 
12 A=[N3/N4 -1; 1 1]
13 B=[0; r1+r2]
14 V=A \ B
15 r3=V(1) //mm
16 r4=V(2) //mm
17 //For pulleys 5 and 6,  $r6 = r5*(N5/N6)$ , or  $r5*(N5/N6)$ 
     $-r6 = 0$ 
18 //For a crossed belt drive,  $r5+r6 = r1+r2$ 
19 A=[N5/N6 -1; 1 1]
20 B=[0; r1+r2]
21 V=A \ B
22 r5=V(1) //mm
23 r6=V(2) //mm
24 //Results:
25 printf("\n\n For a crossed belt, \n r2 = %.1fmm;\n",
    r2)
26 printf(" r3 = %.1 f mm;\n",r3)
27 printf(" r4 = %.1 f mm;\n",r4)
28 printf(" r5 = %.1 f mm;\n",r5)
29 printf(" r6 = %.1 f mm.\n\n",r6)
30 //For an open belt:
31 //Calculating the radii of pulleys 2, 3, 4, 5, and 6
32 r2=r1*(N1/N2) //mm
33 //Calculating the length of belt for an open belt
    drive
34 L=%pi*(r1+r2)+(r2-r1)^2/x+2*x //mm
35 //For pulleys 3 and 4,  $r4 = r3*(N3/N4)$ , or  $r3*(N3/N4)$ 
     $-r4 = 0$ 
36 //Since L is constant, for pulleys 3 and 4,  $\%pi*(r3+$ 
     $r4)+(r4-r3)^2/x+2*x-L = 0$ 
37 funcprot(0)
38 function y=f(a)
39     r3=a(1)

```

```

40     r4=a(2)
41     y(1)=r3*(N3/N4)-r4
42     y(2)=%pi*(r3+r4)+(r4-r3)^2/x+2*x-L
43 endfunction
44 z=fsolve([1,1],f)
45 r3=z(1) //mm
46 r4=z(2) //mm
47 //For pulleys 5 and 6, r6 = r5*(N5/N6), or r5*(N5/N6
    )-r6 = 0
48 //Since L is constant, for pulleys 5 and 6, %pi*(r5+
    r6)+(r6-r5)^2/x+2*x-L = 0
49 function y=f(a)
50     r5=a(1)
51     r6=a(2)
52     y(1)=r5*(N5/N6)-r6
53     y(2)=%pi*(r5+r6)+(r6-r5)^2/x+2*x-L
54 endfunction
55 z=fsolve([1,1],f)
56 r5=z(1) //mm
57 r6=z(2) //mm
58 //Results:
59 printf(" For an open belt ,\n r2 = %.1fmm,\n",r2)
60 printf(" r3 = %.1 f mm;\n",r3)
61 printf(" r4 = %.1 f mm;\n",r4)
62 printf(" r5 = %d mm;\n",r5)
63 printf(" r6 = %d mm.\n\n",r6)

```

---

**Scilab code Exa 11.4** To find the power transmitted

```

1 //To find the power transmitted
2 clc
3 //Given:
4 d=600/1000 //m
5 N=200 //rpm
6 mu=0.25

```

```

7 theta=160*%pi/180 //radians
8 T1=2500 //N
9 //Solution:
10 //Calculating the velocity of the belt
11 v=%pi*d*N/60 //m/s
12 //Calculating the tension in the slack side of the
    belt
13 T2=T1/exp(mu*theta) //N
14 //Calculating the power transmitted by the belt
15 P=(T1-T2)*v/1000 //kW
16 //Results:
17 printf("\n\n Power transmitted by the belt , P = %.2 f
    kW.\n\n",P)

```

---

**Scilab code Exa 11.5** To find force and power

```

1 //To find force and power
2 clc
3 //Given:
4 W=9*1000, T1=W //N
5 d=300/1000 //m
6 N=20 //rpm
7 mu=0.25
8 //Solution:
9 //Force required by the man:
10 //Calculating the angle of contact
11 theta=2.5*2*%pi //rad
12 //Calculating the force required by the man
13 T2=T1/exp(mu*theta) //N
14 //Power to raise the casting:
15 //Calculating the velocity of the rope
16 v=%pi*d*N/60 //m/s
17 //Calculating the power to raise the casting
18 P=(T1-T2)*v/1000 //kW
19 //Results:

```

```

20 printf("\n\n Force required by the man, T2 = %.2f N
    .\n\n",T2)
21 printf(" Power to raise the casting , P = %.3f kW.\n\n\
n",P)

```

---

**Scilab code Exa 11.6** To find length and power transmitted

```

1 //To find length and power transmitted
2 clc
3 //Given:
4 d1=450/1000, r1=d1/2, d2=200/1000, r2=d2/2, x=1.95
    //m
5 N1=200 //rpm
6 T1=1*1000 //N
7 mu=0.25
8 //Solution:
9 //Calculating the speed of the belt
10 v=%pi*d1*N1/60 //m/s
11 //Length of the belt:
12 //Calculating the length of the crossed belt
13 L=%pi*(r1+r2)+2*x+(r1+r2)^2/x //m
14 //Angle of contact between the belt and each pulley:
15 //Calculating the angle alpha
16 alpha=asin((r1+r2)/x)*180/%pi //degrees
17 //Calculating the angle of contact between the belt
    and each pulley
18 theta=(180+2*alpha)*%pi/180 //radians
19 //Power transmitted:
20 //Calculating the tension in the slack side of the
    belt
21 T2=T1/exp(mu*theta) //N
22 //Calculating the power transmitted
23 P=(T1-T2)*v/1000 //kW
24 //Results:
25 printf("\n\n Length of the belt , L = %.3f m.\n\n",L)

```

```

26 printf(" Angle of contact between the belt and each
    pulley , theta = %.3 f rad.\n\n",theta)
27 printf(" Power transmitted , P = %.2 f kW.\n\n",P)

```

---

Scilab code Exa 11.7 To find stress in the belt

```

1 //To find stress in the belt
2 clc
3 //Given:
4 N1=200, N2=300 //rpm
5 P=6*1000 //W
6 b=100, t=10 //mm
7 x=4, d2=0.5 //m
8 mu=0.3
9 //Solution:
10 //Stress in the belt for an open belt drive:
11 //Calculating the diameter of the larger pulley
12 d1=d2*(N2/N1) //m
13 //Calculating the velocity of the belt
14 v=%pi*d2*N2/60 //m/s
15 //Calculating the angle alpha for an open belt drive
16 alphao=asin((r1-r2)/x)*180/%pi //degrees
17 //Calculating the angle of contact on the smaller
    pulley
18 thetiao=(180-2*alphao)*%pi/180 //radians
19 //Calculating the tensions in the belt
20 //Ratio of the tensions in the belt , T1/T2 = exp(mu*
    thetiao), or T1-T2*exp(mu*thetiao) = 0
21 //Power transmitted , P = (T1-T2)*v, or T1-T2 = P/v
22 A=[1 -exp(mu*thetiao); 1 -1]
23 B=[0; P/v]
24 V=A \ B
25 T1o=V(1) //N
26 T2o=V(2) //N
27 //Calculating the stress in the belt

```

```

28 sigmao=T1o/(b*t) //MPa
29 //Stress in the belt for a cross belt drive:
30 //Calculating the angle alpha for a cross belt drive
31 alphac=asin((d1+d2)/(2*x))*180/%pi //degrees
32 //Calculating the angle of contact
33 thetac=(180+2*alphac)*%pi/180 //radians
34 //Calculating the tensions in the belt
35 //Ratio of the tensions in the belt, T1/T2 = exp(mu*
    thetac), or T1-T2*exp(mu*thetac) = 0
36 //Power transmitted, P = (T1-T2)*v, or T1-T2 = P/v
37 A=[1 -exp(mu*thetac); 1 -1]
38 B=[0; P/v]
39 V=A \ B
40 T1c=V(1) //N
41 T2c=V(2) //N
42 //Calculating the stress in the belt
43 sigmac=T1c/(b*t) //MPa
44 //Results:
45 printf("\n\n Stress in the belt for an open belt
    drive, sigma = %.3f MPa.\n\n",sigmao)
46 printf(" Stress in the belt for a cross belt drive,
    sigma = %.3f MPa.\n\n",sigmac)

```

---

**Scilab code Exa 11.8** To find width of the belt

```

1 //To find width of the belt
2 clc
3 //Given:
4 P=7.5*1000 //W
5 d=1.2, t=10/1000 //m
6 N=250 //rpm
7 theta=165*%pi/180 //radians
8 mu=0.3
9 sigma=1.5*10^6 //N/m^2
10 rho=1*10^3 //kg/m^3

```



```

11 //Solution:
12 //Calculating the velocity of the belt
13 v=%pi*d*N/60 //m/s
14 //Calculating the tensions in the belt
15 //Power transmitted, P = (T1-T2)*v, or T1-T2 = P/v
16 //Ratio of tensions in the belt, log(T1/T2) = mu*
    theta, or T1-T2*exp(mu*theta) = 0
17 A=[1 -1; 1 -exp(mu*theta)]
18 B=[P/v; 0]
19 V=A \ B
20 T1=V(1) //N
21 T2=V(2) //N
22 //Calculating the width of the belt
23 b=T1/(sigma*t-t*1*rho*v^2)*1000 //mm
24 //Results:
25 printf("\n\n Width of the belt, b = %.1f mm.\n\n",b)

```

---

**Scilab code Exa 11.9** To find width of the belt

```

1 //To find width of the belt
2 clc
3 //Given:
4 t=9.75/1000, d1=300/1000, x=3 //m
5 P=15*1000 //W
6 N1=900, N2=300 //rpm
7 rho=1000 //kg/m^3
8 sigma=2.5*10^6 //N/m^2
9 mu=0.3
10 //Solution:
11 //Calculating the diameter of the driven pulley
12 d2=d1*(N1/N2) //m
13 //Calculating the velocity of the belt
14 v=%pi*d1*N1/60 //m/s
15 //Calculating the angle alpha for an open belt drive
16 alpha=asin((d2-d1)/(2*x))*180/%pi //degrees

```

```

17 //Calculating the angle of lap
18 theta=(180-2*alpha)*%pi/180 //radians
19 //Calculating the tensions in the belt
20 //Ratio of tensions ,  $\log(T1/T2) = \mu*\theta$  , or  $T1-T2$ 
    *exp( $\mu*\theta$ ) = 0
21 //Power transmitted ,  $P = (T1-T2)*v$  , or  $T1-T2 = P/v$ 
22 A=[1 -exp( $\mu*\theta$ ); 1 -1]
23 B=[0; P/v]
24 V=A \ B
25 T1=V(1) //N
26 T2=V(2) //N
27 //Calculating the width of the belt
28 b=T1/(sigma*t-t*1*rho*v^2)*1000 //mm
29 //Results:
30 printf("\n\n Width of the belt , b = %d mm.\n\n",b)

```

---

**Scilab code Exa 11.10** To find greatest power transmitted

```

1 //To find greatest power transmitted
2 clc
3 //Given:
4 theta=120*%pi/180 //radians
5 b=100/1000 , t=6/1000 //m
6 rho=1000 //kg/m^3
7 mu=0.3
8 sigma=2*10^6 //N/m^2
9 //Solution:
10 //Speed of the belt for greatest power:
11 //Calculating the maximum tension in the belt
12 T=sigma*b*t //N
13 //Calculating the mass of the belt per metre length
14 l=1 //m
15 m=b*t*l*rho //kg/m
16 //Calculating the speed of the belt for greatest
    power

```

```

17 v=sqrt(T/(3*m)) //m/s
18 //Greatest power which the belt can transmit
19 //Calculating the centrifugal tension for maximum
    power to be transmitted
20 TC=T/3 //N
21 //Calculating the tension in the tight side of the
    belt
22 T1=T-TC //N
23 //Calculating the tension in the slack side of the
    belt
24 T2=T1/exp(mu*theta) //N
25 //Calculating the greatest power which the belt can
    transmit
26 P=(T1-T2)*v/1000 //kW
27 //Results:
28 printf("\n\n Speed of the belt for greatest power, v
    = %.2 f m/s.\n\n",v)
29 printf(" Greatest power which the belt can transmit,
    P = %.2 f kW.\n\n",P)

```

---

**Scilab code Exa 11.11** To find torque power and efficiency

```

1 //To find torque , power and efficiency
2 clc
3 //Given:
4 d1=1.2, r1=d1/2, d2=0.5, r2=d2/2, x=4 //m
5 m=0.9 //kg/m
6 T=2000 //N
7 mu=0.3
8 N1=200, N2=450 //rpm
9 //Solution:
10 //Calculating the velocity of the belt
11 v=%pi*d1*N1/60 //m/s
12 //Calculating the centrifugal tension
13 TC=m*v^2 //N

```

```

14 //Calculating the tension in the tight side of the
    belt
15 T1=T-TC //N
16 //Calculating the angle alpha for an open belt drive
17 alpha=asin((r1-r2)/x)*180/%pi //degrees
18 //Calculating the angle of lap on the smaller pulley
19 theta=(180-2*alpha)*%pi/180 //radians
20 //Calculating the tension in the slack side of the
    belt
21 T2=T1/exp(mu*theta) //N
22 //Calculating the torque on the shaft of larger
    pulley
23 TL=(T1-T2)*r1 //N-m
24 //Calculating the torque on the shaft of smaller
    pulley
25 TS=(T1-T2)*r2 //N-m
26 //Calculating the power transmitted
27 P=(T1-T2)*v/1000 //kW
28 //Power lost in friction:
29 //Calculating the input power
30 P1=TL*2*%pi*N1/(60*1000) //kW
31 //Calculating the output power
32 P2=TS*2*%pi*N2/(60*1000) //kW
33 //Calculating the power lost in friction
34 Pf=P1-P2 //Power lost in friction , kW
35 //Calculating the efficiency of the drive
36 eta=P2/P1*100 //%
37 //Results:
38 printf("\n\n Torque on the shaft of larger pulley ,
    TL = %.1f N-m.\n\n",TL)
39 printf(" Torque on the shaft of smaller pulley , TS =
    %d N-m.\n\n",TS)
40 printf(" Power transmitted , P = %.2f kW.\n\n",P)
41 printf(" Power lost in friction = %.2f kW.\n\n",Pf)
42 printf(" Efficiency of the drive , eta = %.1f %c.\n\n
    ",eta,"%")

```

---

**Scilab code Exa 11.12** To find power transmitted

```
1 //To find power transmitted
2 clc
3 //Given:
4 T0=2000 //N
5 mu0=0.3
6 theta=150*%pi/180 //radians
7 r2=200/1000, d2=2*r2 //m
8 N2=500//rpm
9 //Solution:
10 //Calculating the velocity of the belt
11 v=%pi*d2*N2/60 //m/s
12 //Calculating the tensions in the belt
13 //Initial tension,  $T_0 = (T_1+T_2)/2$ , or  $T_1+T_2 = 2*T_0$ 
14 //Ratio of the tensions in the belt,  $\log(T_1/T_2) = \mu$ 
    *theta, or  $T_1-T_2*\exp(\mu*\theta) = 0$ 
15 A=[1 1; 1 -exp(mu*theta)]
16 B=[2*T0; 0]
17 V=A \ B
18 T1=V(1) //N
19 T2=V(2) //N
20 //Calculating the power transmitted
21 P=(T1-T2)*v/1000 //kW
22 //Results:
23 printf("\n\n Power transmitted , P = %.1f kW.\n\n",P)
```

---

**Scilab code Exa 11.13** To find power transmitted

```
1 //To find power transmitted
2 clc
3 //Given:
```

```

4 x=4.8, d1=1.5, d2=1 //m
5 T0=3*1000 //N
6 m=1.5 //kg/m
7 mu=0.3
8 N2=400 //rpm
9 //Solution:
10 //Calculating the velocity of the belt
11 v=%pi*d2*N2/60 //m/s
12 //Calculating the centrifugal tension
13 TC=m*v^2 //N
14 //Calculating the angle alpha
15 alpha=asin((d1-d2)/(2*x))*180/%pi //degrees
16 //Calculating the angle of lap for the smaller
    pulley
17 theta=(180-2*alpha)*%pi/180 //radians
18 //Calculating the tensions in the belt
19 //Initial tension,  $T_0 = (T_1+T_2+2*TC)/2$ , or  $T_1+T_2 =$ 
     $2*(T_0-TC)$ 
20 //Ratio of tensions in the belt,  $\log(T_1/T_2) = \mu*$ 
     $\theta$ , or  $T_1-T_2*\exp(\mu*\theta) = 0$ 
21 A=[1 1; 1 -exp(mu*theta)]
22 B=[2*(T0-TC); 0]
23 V=A \ B
24 T1=V(1) //N
25 T2=V(2) //N
26 //Calculating the power transmitted
27 P=(T1-T2)*v/1000 //kW
28 //Results:
29 printf("\n\n Power transmitted , P = %.1f kW.\n\n",P)

```

---

**Scilab code Exa 11.14** To find diameter power and tension

```

1 //To find diameter , power and tension
2 clc
3 //Given:

```

```

4 x=1.2, d2=400/1000, t=5/1000, b=80/1000 //m
5 N1=350, N2=140 //rpm
6 mu=0.3
7 sigma=1.4*10^6 //N/m^2
8 //Solution:
9 //Calculating the diameter of the driving pulley
10 d1=d2*(N2/N1) //m
11 //Maximum power transmitted by the belting:
12 //Refer Fig. 11.18
13 //Calculating the angle alpha
14 alpha=asin((d2-d1)/(2*x))*180/%pi //degrees
15 //Calculating the angle of contact of the belt on
    the driving pulley
16 theta=(180-2*alpha)*%pi/180 //radians
17 //Calculating the maximum tension to which the belt
    can be subjected
18 T1=sigma*b*t //N
19 //Calculating the tension in the slack side of the
    belt
20 T2=T1/exp(mu*theta) //N
21 //Calculating the velocity of the belt
22 v=%pi*d1*N1/60 //m/s
23 //Calculating the power transmitted
24 P=(T1-T2)*v/1000 //kW
25 //Calculating the required initial belt tension
26 T0=(T1+T2)/2 //N
27 //Results:
28 printf("\n\n Diameter of the driving pulley, d1 = %
    .2 f m.\n\n",d1)
29 printf(" Maximum power transmitted by the belting, P
    = %.3 f kW.\n\n",P)
30 printf(" Required initial belt tension, T0 = %.1 f N
    .\n\n",T0)

```

---

**Scilab code Exa 11.15** To find width tension and length

```

1 //To find width , tension and length
2 clc
3 //Given:
4 d2=240/1000 , d1=600/1000 , x=3 //m
5 P=4*1000 //W
6 N2=300 //rpm
7 mu=0.3
8 T1s=10 //Safe working tension , N/mm width
9 //Solution:
10 //Minimum width of the belt:
11 //Calculating the velocity of the belt
12 v=%pi*d2*N2/60 //m/s
13 //Calculating the angle alpha for an open belt drive
14 alpha=asin((d1-d2)/(2*x))*180/%pi //degrees
15 //Calculating the angle of lap on the smaller pulley
16 theta=(180-2*alpha)*%pi/180 //radians
17 //Calculating the tensions in the belt
18 //Power transmitted , P = (T1-T2)*v , or T1-T2 = P/v
19 //Ratio of tensions , log(T1/T2) = mu*theta , or T1-T2
    *exp(mu*theta) = 0
20 A=[1 -1; 1 -exp(mu*theta)]
21 B=[P/v; 0]
22 V=A \ B
23 T1=V(1) //N
24 T2=V(2) //N
25 //Calculating the minimum width of the belt
26 b=T1/T1s //mm
27 //Calculating the initial belt tension
28 T0=(T1+T2)/2 //N
29 //Calculating the length of the belt required
30 L=%pi/2*(d1+d2)+2*x+(d1-d2)^2/(4*x) //m
31 //Results:
32 printf(" \n\n Minimum width of the belt , b = %.1f mm
    . \n\n" ,b)
33 printf(" Initial belt tension , T0 = %.1f N.\n\n" ,T0)
34 printf(" Length of the belt required , L = %.2f m.\n\n
    n" ,L)

```

---



**Scilab code Exa 11.16** To find power transmitted

```
1 //To find power transmitted
2 clc
3 //Given:
4 d1=400/1000 ,d2=250/1000 , x=2 , mu=0.4 //m
5 T=1200 //N
6 v=10 //m/s
7 //Solution:
8 //Power transmitted:
9 //Calculating the angle alpha for an open belt drive
10 alpha=asin((d1-d2)/(2*x))*180/%pi //degrees
11 //Calculating the angle of contact
12 theta=(180-2*alpha)*%pi/180 //radians
13 //Calculating the tension in the tight side of the
    belt
14 T1=T //Neglecting centrifugal tension , N
15 //Calculating the tension in the slack side of the
    belt
16 T2=T1/exp(mu*theta) //N
17 //Calculating the power transmitted
18 P=(T1-T2)*v/1000 //kW
19 //Results:
20 printf("\\n\\n Power transmitted , P = %.2f kW.\\n\\n" ,P)
21 //Power transmitted when initial tension is
    increased by 10%:
22 //Calculating the initial tension
23 T0=(T1+T2)/2 //N
24 //Calculating the increased initial tension
25 T0dash=T0+10/100*T0 //N
26 //Calculating the corresponding tensions in the belt
27 //We have , T0dash = (T1+T2)/2 , or T1+T2 = 2*T0dash
28 //Ratio of the tensions ,  $\log(T1/T2) = \mu * \theta$  , or
     $T1 - T2 * \exp(\mu * \theta) = 0$ 
```

```

29 A=[1 1; 1 -exp(mu*theta)]
30 B=[2*T0dash; 0]
31 V=A \ B
32 T1=V(1) //N
33 T2=V(2) //N
34 //Calculating the power transmitted
35 P1=(T1-T2)*v/1000 //kW
36 //Power transmitted when coefficient of friction is
    increased by 10%:
37 //Calculating the increased coefficient of friction
38 mudash=mu+10/100*mu
39 //Calculating the corresponding tensions in the belt
40 //Ratio of the tensions , log(T1/T2) = mudash*theta ,
    or T1-T2*exp(mudash*theta) = 0
41 //Initial tension , T0 = (T1+T2)/2, or T1+T2 = 2*T0
42 A=[1 -exp(mudash*theta); 1 1]
43 B=[0; 2*T0]
44 V=A \ B
45 T1=V(1) //N
46 T2=V(2) //N
47 //Calculating the power transmitted
48 P2=(T1-T2)*v/1000 //kW
49 //Results:
50 if P1>P2 then
51     printf(" Since the power transmitted by
        increasing the initial tension is more,
        therefore in order to increase the power
        transmitted we shall adopt the method of
        increasing the initial tension.\n\n")
52 else
53     printf(" Since the power transmitted by
        increasing the coefficient of friction is
        more, therefore in order to increase the
        power transmitted we shall adopt the method
        of increasing the coefficient of friction.\n\n")
54 end
55 //Percentage increase in power:\

```

```

56 //Calculating the percentage increase in power when
    the initial tension is increased
57 I1=(P1-P)/P*100 //Percentage increase in power when
    the initial tension is increased, %
58 //Calculating the percentage increase in power when
    coefficient of friction is increased
59 I2=(P2-P)/P*100 //Percentage increase in power when
    coefficient of friction is increased, %
60 //Results:
61 printf(" Percentage increase in power when the
    initial tension is increased = %.2f %c.\n\n",I1,"
    %")
62 printf(" Percentage increase in power when
    coefficient of friction is increased = %.1f %c.\n
    \n",I2,"%")

```

---

**Scilab code Exa 11.17** To find power and shaft speed

```

1 //To find power and shaft speed
2 clc
3 //Given:
4 funcprot(0)
5 beta=30/2 //degrees
6 alpha=750*10^-6 //mm^2
7 mu=0.12
8 rho=1.2*1000 //kg/m^3
9 sigma=7*10^6 //N/m^2
10 d=300/1000 //m
11 N=1500 //rpm
12 //Solution:
13 //Power transmitted:
14 //Calculating the velocity of the belt
15 v=%pi*d*N/60 //m/s
16 //Calculating the mass of the belt per metre length
17 l=1 //m

```

```

18 m=alpha*l*rho //kg/m
19 //Calculating the centrifugal tension
20 TC=m*v^2 //N
21 //Calculating the maximum tension in the belt
22 T=sigma*alpha //N
23 //Calculating the tension in the tight side of the
    belt
24 T1=T-TC //N
25 //Calculating the tension in the slack side of the
    belt
26 theta=%pi //Angle of contact , radians
27 T2=T1/exp(mu*theta*(1/sind(beta))) //N
28 //Calculating the power transmitted
29 P=(T1-T2)*v*2/1000 //kW
30 //Shaft speed:
31 //Calculating the belt speed for maximum power
    transmitted
32 v1=sqrt(T/(3*m)) //m/s
33 //Calculating the shaft speed for maximum power
    transmitted
34 N1=v1*60/(%pi*d) //rpm
35 //Results:
36 printf("\n\n Power transmitted , P = %.3f kW.\n\n",P)
37 printf(" Shaft speed at which the power transmitted
    would be maximum, N1 = %d rpm.\n\n",N1)

```

---

**Scilab code Exa 11.18** To find maximum power transmitted

```

1 //To find maximum power transmitted
2 clc
3 //Given:
4 funcprot(0)
5 beta=30/2 //degrees
6 t=20/1000, b=20/1000 //m
7 m=0.35 //kg/m

```

```

8 sigma=1.4*10^6 //N/m^2
9 theta=140*%pi/180 //radians
10 mu=0.15
11 //Solution:
12 //Calculating the maximum tension in the belt
13 T=sigma*b*t //N
14 //Calculating the velocity of the belt for maximum
    power to be transmitted
15 v=sqrt(T/(3*m)) //m/s
16 //Calculating the centrifugal tension
17 TC=T/3 //N
18 //Calculating the tension in the tight side of the
    belt
19 T1=T-TC //N
20 //Calculating the tension in the slack side of the
    belt
21 T2=T1/exp(mu*theta*(1/sind(beta))) //N
22 //Calculating the maximum power transmitted
23 P=(T1-T2)*v/1000 //kW
24 //Results:
25 printf("\n\n Maximum power transmitted , P = %.2f kW
    .\n\n",P)

```

---

**Scilab code Exa 11.19** To find number of V belts

```

1 //To find number of V-belts
2 clc
3 //Given:
4 P=90 //kW
5 N2=250 , N1=750 //rpm
6 d2=1, x=1.75 //m
7 v=1600/60 //m/s
8 a=375*10^-6 //m^2
9 rho=1000 //kg/m^3
10 sigma=2.5*10^6 //N/m^2

```

```

11 beta=35/2 //degrees
12 mu=0.25
13 //Solution:
14 //Calculating the diameter of the pulley on the
    motor shaft
15 d1=d2*(N2/N1) //m
16 //Calculating the mass of the belt per metre length
17 l=1 //m
18 m=a*l*rho //kg/m
19 //Calculating the centrifugal tension
20 TC=m*v^2 //N
21 //Calculating the maximum tension in the belt
22 T=sigma*a //N
23 //Calculating the tension in the tight side of the
    belt
24 T1=T-TC //N
25 //Refer Fig. 11.21
26 //Calculating the angle alpha
27 alpha=asin((d2-d1)/(2*x))*180/%pi //degrees
28 //Calculating the angle of lap on smaller pulley
29 theta=(180-2*alpha)*%pi/180 //radians
30 //Calculating the tension in the slack side of the
    belt
31 T2=T1/exp(mu*theta*(1/sind(beta))) //N
32 //Number of V-belts:
33 //Calculating the power transmitted per belt
34 P1=(T1-T2)*v/1000 //Power transmitted per belt , kW
35 //Calculating the number of V-belts
36 n=P/P1 //Number of V-belts
37 //Calculating the length each of belt for an open
    belt drive
38 L=%pi/2*(d2+d1)+2*x+(d2-d1)^2/(4*x) //m
39 //Results:
40 printf("\n\n Number of V-belts = %d.\n\n",n+1)
41 printf(" Length of each belt , L = %.3f m.\n\n",L)

```

---

**Scilab code Exa 11.20** To find number of ropes required

```
1 //To find number of ropes required
2 clc
3 //Given:
4 P=600 //kW
5 d=4 //m
6 N=90 //rpm
7 theta=160*%pi/180 //radians
8 funcprot(0)
9 beta=45/2 //degrees
10 mu=0.28
11 m=1.5 //kg/m
12 T=2400 //N
13 //Solution:
14 //Calculating the velocity of the rope
15 v=%pi*d*N/60 //m/s
16 //Calculating the centrifugal tension
17 TC=m*v^2 //N
18 //Calculating the tension in the tight side of the
    rope
19 T1=T-TC //N
20 //Calculating the tension in the slack side of the
    belt
21 T2=T1/exp(mu*theta*(1/sind(beta))) //N
22 //Calculating the power transmitted per rope
23 P1=(T1-T2)*v/1000 //Power transmitted per rope, kW
24 //Calculating the number of ropes
25 n=P/P1 //Number of ropes
26 //Results:
27 printf("\\n\\n Number of ropes required = %d.\\n\\n",n
    +1)
```

---

**Scilab code Exa 11.21** To find speed of pulley

```
1 //To find speed of pulley
2 clc
3 //Given:
4 d=3.6 //m
5 n=15 //Number of grooves
6 funcprot(0)
7 beta=45/2 //degrees
8 theta=170*%pi/180 //radians
9 mu=0.28
10 T=960 //N
11 m=1.5 //kg/m
12 //Solution:
13 //Speed of the pulley:
14 //Calculating the velocity of the rope
15 v=sqrt(T/(3*m)) //m/s
16 //Calculating the speed of the pulley
17 N=v*60/(%pi*d) //rpm
18 //Power transmitted
19 //Calculating the centrifugal tension for maximum
   power
20 TC=T/3 //N
21 //Calculating the tension in the tight side of the
   rope
22 T1=T-TC //N
23 //Calculating the tension in the slack side of the
   rope
24 T2=T1/exp(mu*theta*(1/sind(beta))) //N
25 //Calculating the power transmitted per rope
26 P1=(T1-T2)*v/1000 //Power transmitted per rope, kW
27 //Calculating the total power transmitted
28 P=P1*n //Total power transmitted, kW
29 //Results:
```



```

30 printf("\n\n Speed of the pulley for maximum power ,
      N = %.1f rpm.\n\n",N)
31 printf(" Power transmitted = %.2f kW.\n\n",P)

```

---

**Scilab code Exa 11.22** To find initial tension and diameter

```

1 //To find initial tension and diameter
2 clc
3 //Given:
4 PT=24 //kW
5 d=400/1000 //m
6 N=110 //rpm
7 funcprot(0)
8 beta=45/2 //degrees
9 theta=160*%pi/180 //radians
10 mu=0.28
11 n=10
12 //Solution:
13 //Initial tension:
14 //Calculating the power transmitted per rope
15 P=PT/n*1000 //W
16 //Calculating the velocity of the rope
17 v=%pi*d*N/60 //m/s
18 //Calculating the tensions in the rope
19 //Power transmitted,  $P = (T_1 - T_2) * v$ , or  $T_1 - T_2 = P/v$ 
20 //Ratio of tensions,  $\log(T_1/T_2) = \mu * \theta * (1/\sin(\beta))$ , or  $T_1 - T_2 * \exp(\mu * \theta * (1/\sin(\beta))) = 0$ 
21 A=[1 -1; 1 -exp(mu*theta*(1/sind(beta)))]
22 B=[P/v; 0]
23 V=A \ B
24 T1=V(1) //N
25 T2=V(2) //N
26 //Calculating the initial tension in each rope
27 T0=(T1+T2)/2 //N
28 //Diameter of each rope:

```

```

29 //Calculating the girth of rope
30 C=sqrt(T1/(122*10^3-53*v^2))*1000 //mm
31 //Calculating the diameter of each rope
32 d1=C/%pi //mm
33 //Results:
34 printf("\n\n Initial tension , T0 = %.2 f N.\n\n",T0)
35 printf(" Diameter of each rope , d1 = %.2 f mm.\n\n",
    d1)

```

---

**Scilab code Exa 11.23** To find pitch and length of chain

```

1 //To find pitch and length of chain
2 clc
3 //Given:
4 N1=240 , N2=120 //rpm
5 T1=20
6 d2=600/1000 , r2=d2/2 , x=800/1000 //m
7 //SOLution:
8 //Calculating the number of teeth on the drive
    sprocket
9 T2=T1*(N1/N2)
10 //Calculating the pitch of the chain
11 p=r2*2*sind(180/T2)*1000 //mm
12 //Length of the chain:
13 m=x*1000/p
14 //Calculating the multiplying factor
15 K=(T1+T2)/2+2*m+(1/sind(180/T1)-1/sind(180/T2))
    ^2/(4*m)
16 //Calculating the length of the chain
17 L=p*K/1000 //m
18 //Results:
19 printf("\n\n Number of teeth on the driven sprocket ,
    T2 = %d.\n\n",T2)
20 printf(" Pitch of the chain , p = %.1 f mm.\n\n",p)
21 printf(" Length of the chain , L = %.4 f m.\n\n",L)

```



# Chapter 12

## Toothed Gearing

Scilab code Exa 12.1 To find total load

```
1 //To find total load
2 clc
3 //Given:
4 P=120*1000 //W
5 d=250/1000, r=d/2 //m
6 N=650 //rpm
7 phi=20 //degrees
8 //Solution:
9 //Calculating the angular speed of the gear
10 omega=2*pi*N/60 //rad/s
11 //Calculating the torque transmitted
12 T=P/omega //N-m
13 //Calculating the tangential load on the pinion
14 FT=T/r //N
15 //Calculating the total load due to power
    transmitted
16 F=FT/(cosd(phi)*1000) //kN
17 //Results:
18 printf("\n\n Total load due to power transmitted , F
    = %.3f kN.\n\n",F)
```

---

**Scilab code Exa 12.2** To find addendum

```
1 //To find addendum
2 clc
3 //Given:
4 T=40, t=T
5 phi=20 //degrees
6 m=6 //mm
7 //Solution:
8 //Calculating the circular pitch
9 pc=%pi*m //mm
10 //Calculating the length of arc of contact
11 Lac=1.75*pc //Length of arc of contact, mm
12 //Calculating the length of path of contact
13 Lpc=Lac*cosd(phi) //Length of path of contact, mm
14 //Calculating the pitch circle radii of each wheel
15 R=m*T/2 //mm
16 r=R //mm
17 //Calculating the radius of the addendum circle of
    each wheel
18 RA=sqrt(R^2*(cosd(phi))^2+(Lpc/2+R*sind(phi))^2) //
    mm
19 //Calculating the addendum of the wheel
20 Ad=RA-R //Addendum of the wheel, mm
21 //Results:
22 printf("\\n\\n Addendum of the wheel = %.2f mm.\\n\\n",
    Ad)
```

---

**Scilab code Exa 12.3** To find length of path of contact

```
1 //To find length of path of contact
2 clc
```

```

3 //Given:
4 t=30, T=80
5 phi=20 //degrees
6 m=12 //mm
7 Addendum=10 //mm
8 //Solution:
9 //Length of path of contact:
10 //Calculating the pitch circle radius of pinion
11 r=m*t/2 //mm
12 //Calculating the pitch circle radius of gear
13 R=m*T/2 //mm
14 //Calculating the radius of addendum circle of
    pinion
15 rA=r+Addendum //mm
16 //Calculating the radius of addendum circle of gear
17 RA=R+Addendum //mm
18 //Calculating the length of path of approach
19 //Refer Fig. 12.11
20 KP=sqrt(RA^2-R^2*(cosd(phi))^2)-R*sind(phi) //mm
21 //Calculating the length of path of recess
22 PL=sqrt(rA^2-r^2*(cosd(phi))^2)-r*sind(phi) //mm
23 //Calculating the length of path of contact
24 KL=KP+PL //mm
25 //Calculating the length of arc of contact
26 Lac=KL/cosd(phi) //Length of arc of contact, mm
27 //Contact ratio:
28 //Calculating the circular pitch
29 Pc=%pi*m //mm
30 //Calculating the contact ratio
31 CR=Lac/pc //Contact ratio
32 //Results:
33 printf("\n\n Length of path of contact, KL = %.1f mm
    .\n\n",KL)
34 printf(" Length of arc of contact = %.2f mm.\n\n",
    Lac)
35 printf(" Contact ratio = %d.\n\n",CR)

```

---

**Scilab code Exa 12.4** To find angle and maximum velocity

```
1 //To find angle and maximum velocity
2 clc
3 //Given:
4 phi=20 //degrees
5 t=20, G=2
6 m=5 //mm
7 v=1.2 //m/s
8 addendum=1*m //mm
9 //Solution:
10 //Angle turned through by pinion when one pair of
    teeth is in mesh:
11 //Calculating the pitch circle radius of pinion
12 r=m*t/2 //mm
13 //Calculating the pitch circle radius of wheel
14 R=m*G*t/2 //mm
15 //Calculating the radius of addendum circle of
    pinion
16 rA=r+addendum //mm
17 //Calculating the radius of addendum circle of wheel
18 RA=R+addendum //mm
19 //Calculating the length of path of approach
20 KP=sqrt(RA^2-R^2*(cosd(phi))^2)-R*sind(phi) //mm
21 //Calculating the length of path of recess
22 PL=sqrt(rA^2-r^2*(cosd(phi))^2)-r*sind(phi) //mm
23 //Calculating the length of path of contact
24 KL=KP+PL //mm
25 //Calculating the length of arc of contact
26 Lac=KL/cosd(phi) //mm
27 //Calculating the angle turned by the pinion
28 angle=Lac*360/(2*%pi*r) //Angle turned by the pinion
    , degrees
29 //Maximum velocity of sliding:
```

```

30 //Calculating the angular speed of pinion
31 omega1=v*1000/r //rad/s
32 //Calculating the angular speed of wheel
33 omega2=v*1000/R //rad/s
34 //Calculating the maximum velocity of sliding
35 vS=(omega1+omega2)*KP //mm/s
36 //Results:
37 printf("\n\n Angle turned through by pinion when one
      pair of teeth is in mesh = %.2f degrees.\n\n",
      angle)
38 printf(" Maximum velocity of sliding , vS = %.1f mm/s
      .\n\n",vS)

```

---

**Scilab code Exa 12.5** To find velocity and angle turned

```

1 //To find velocity and angle turned
2 clc
3 //Given:
4 T=40, t=20
5 N1=2000 //rpm
6 phi=20 //degrees
7 addendum=5, m=5 //mm
8 //Solution:
9 //Calculating the angular velocity of the smaller
      gear
10 omega1=2*%pi*N1/60 //rad/s
11 //Calculating the angular velocity of the larger
      gear
12 omega2=omega1*t/T //rad/s
13 //Calculating the pitch circle radius of the smaller
      gear
14 r=m*t/2 //mm
15 //Calculating the pitch circle radius of the larger
      gear
16 R=m*T/2 //mm

```



```

17 //Calculating the radius of addendum circle of
    smaller gear
18 rA=r+addendum //mm
19 //Calculating the radius of addendum circle of
    larger gear
20 RA=R+addendum //mm
21 //Calculating the length of path of approach
22 KP=sqrt(RA^2-R^2*(cosd(phi))^2)-R*sind(phi) //mm
23 //Calculating the length of path of recess
24 PL=sqrt(rA^2-r^2*(cosd(phi))^2)-r*sind(phi) //mm
25 //Calculating the velocity of sliding at the point
    of engagement
26 vSK=(omega1+omega2)*KP //mm/s
27 //Calculating the velocity of sliding at the point
    of disengagement
28 vSL=(omega1+omega2)*PL //mm/s
29 //Angle through which the pinion turns:
30 //Calculating the length of path of contact
31 KL=KP+PL //mm
32 //Calculating the length of arc of contact
33 Lac=KL/cosd(phi) //Length of arc of contact, mm
34 //Calculating the circumference of pinion
35 C=2*pi*r //Circumference of pinion, mm
36 //Calculating the angle through which the pinion
    turns
37 angle=Lac*360/C //Angle through which the pinion
    turns, degrees
38 //Results:
39 printf("\n\n Velocity of sliding at the point of
    engagement, vSK = %d mm/s.\n\n",vSK)
40 printf(" Since the velocity of sliding is
    proportional to the distance of the contact point
    from the pitch point, therefore the velocity of
    sliding at the pitch point is zero.\n\n")
41 printf(" Velocity of sliding at the point of
    disengagement, vsL = %d mm/s.\n\n",vSL)
42 printf(" Angle through which the pinion turns = %.2 f
    degrees.\n\n",angle)

```

---

**Scilab code Exa 12.6** To find teeth angle and ratio

```
1 //To find teeth , angle and ratio
2 clc
3 //Given:
4 phi=20 //degrees
5 m=6, addendum=1*m //mm
6 t=17, T=49
7 //Solution:
8 //Number of pairs of teeth in contact:
9 //Calculating the pitch circle radius of pinion
10 r=m*t/2 //mm
11 //Calculating the pitch circle radius of gear
12 R=m*T/2 //mm
13 //Calculating the radius of addendum circle of
    pinion
14 rA=r+addendum //mm
15 //Calculating the radius of addendum circle of gear
16 RA=R+addendum //mm
17 //Calculating the length of path of approach
18 //Refer Fig. 12.11
19 KP=sqrt(RA^2-R^2*(cosd(phi))^2)-R*sind(phi) //mm
20 //Calculating the length of path of recess
21 PL=sqrt(rA^2-r^2*(cosd(phi))^2)-r*sind(phi) //mm
22 //Calculating the length of path of contact
23 KL=KP+PL //mm
24 //Calculating the length of arc of contact
25 Lac=KL/cosd(phi) //Length of arc of contact, mm
26 //Calculating the circular pitch
27 pc=%pi*m //mm
28 //Calculating the number of pairs of teeth in
    contact
29 n=Lac/pc //Number of pairs of teeth in contact
30 //Angle turned by the pinion and gear wheel when one
```

```

    pair of teeth is in contact:
31 //Calculating the angle turned through by the pinion
32 anglep=Lac*360/(2*pi*r) //Angle turned through by
    the pinion , degrees
33 //Calculating the angle turned through by the wheel
34 angleg=Lac*360/(2*pi*R) //Angle turned through by
    the gear wheel , degrees
35 //Ratio of sliding to rolling motion:
36 //At the instant when the tip of a tooth on the
    larger wheel is just making contact with its
    mating teeth
37 r1=((1+t/T)*KP)/r //Ratio of sliding velocity to
    rolling velocity
38 //At the instant when the tip of a tooth on a larger
    wheel is just leaving contact with its mating
    teeth
39 r2=((1+t/T)*PL)/r //Ratio of sliding velocity to
    rolling velocity
40 //Results:
41 printf("\n\n Number of pairs of teeth in contact =
    %d.\n\n",n+1)
42 printf(" Angle turned through by the pinion = %.1f
    degrees.\n\n",anglep)
43 printf(" Angle turned through by the gear wheel = %d
    degrees.\n\n",angleg)
44 printf(" At the instant when the tip of a tooth on
    the larger wheel is just making contact with its
    mating teeth , ratio of sliding velocity to
    rolling velocity = %.2f.\n\n",r1)
45 printf(" At the instant when the tip of a tooth on a
    larger wheel is just leaving contact with its
    mating teeth , ratio of sliding velocity to
    rolling velocity = %.3f.\n\n",r2)
46 printf(" Since at the pitch point , the sliding
    velocity is zero , , therefore the ratio of sliding
    velocity to rolling velocity is zero.\n\n")

```

---

Scilab code Exa 12.7 To find length of path of contact

```
1 //To find length of path of contact
2 clc
3 //Given:
4 t=18, T=72
5 phi=20 //degrees
6 m=4 //mm
7 addendump=8.5 //Addendum on pinion, mm
8 addendumg=3.5 //Addendum on gear, mm
9 //SOLution:
10 //Refer Fig. 12.12
11 //Calculating the pitch circle radius of the pinion
12 r=m*t/2 //mm
13 //Calculating the pitch circle radius of the gear
14 R=m*T/2 //mm
15 //Calculating the radius of addendum circle of the
    pinion
16 rA=r+addendump //mm
17 //Calculating the radius of addendum circle of the
    gear
18 RA=R-addendumg //mm
19 //Calculating the radius of the base circle of the
    pinion
20 O1M=r*cosd(phi) //mm
21 //Calculating the radius of the base circle of the
    gear
22 O2N=R*cosd(phi) //mm
23 //Calculating the length of path of approach
24 KP=R*sind(phi)-sqrt(RA^2-O2N^2) //mm
25 //Calculating the length of path of recess
26 PL=sqrt(rA^2-O1M^2)-r*sind(phi) //mm
27 //Calculating the length of the path of contact
28 KL=KP+PL //mm
```

```

29 //Results :
30 printf("\n\n Length of the path of contact , KL = %.2
    f mm.\n\n",KL)

```

---

**Scilab code Exa 12.8** To find path and arc of contact

```

1 //To find path and arc of contact
2 clc
3 //Given:
4 t=20, T=40
5 m=10 //mm
6 phi=20 //degrees
7 //Solution:
8 //Addendum height for each gear wheel:
9 //Calculating the pitch circle radius of the smaller
    gear wheel
10 r=m*t/2 //mm
11 //Calculating the pitch circle radius of the larger
    wheel
12 R=m*T/2 //mm
13 //Calculating the radius of addendum circle for the
    larger gear wheel
14 RA=sqrt((r*sind(phi)/2+R*sind(phi))^2+R^2*(cosd(phi)
    )^2) //mm
15 //Calculating the addendum height for larger gear
    wheel
16 addendumg=RA-R //mm
17 //Calculating the radius of addendum circle for the
    smaller gear wheel
18 rA=sqrt((R*sind(phi)/2+r*sind(phi))^2+r^2*(cosd(phi)
    )^2) //mm
19 //Calculating the addendum height for smaller gear
    wheel
20 addendump=rA-r //mm
21 //Calculating the length of the path of contact

```

```

22 Lpc=(r+R)*sind(phi)/2 //Length of the path of
    contact , mm
23 //Calculating the length of the arc of contact
24 Lac=Lpc/cosd(phi) //Length of the arc of contact , mm
25 //Contact ratio:
26 //Calculating the circular pitch
27 pc=%pi*m //mm
28 //Calculating the contact ratio
29 CR=Lpc/pc //Contact ratio
30 //Results:
31 printf("\n\n Addendum height for larger gear wheel =
    %.1f mm.\n\n",addendumg)
32 printf(" Addendum height for smaller gear wheel = %
    .1f mm.\n\n",addendumg)
33 printf(" Length of the path of contact = %.1f mm.\n\
    n",Lpc)
34 printf(" Length of the arc of contact = %.1f mm.\n\n
    ",Lac)
35 printf(" Contact ratio = %d.\n\n",CR+1)

```

---

**Scilab code Exa 12.9** To find number of teeth

```

1 //To find number of teeth
2 clc
3 //Given:
4 G=3
5 phi=20 //degrees
6 Aw=1 //module
7 //Solution:
8 //Calculating the minimum number of teeth for a gear
    ratio of 3:1
9 t1=(2*Aw)/(G*(sqrt(1+1/G*(1/G+2)*(sind(phi))^2)-1))
10 //Calculating the minimum number of teeth for equal
    wheel
11 t2=(2*Aw)/(sqrt(1+3*(sind(phi))^2)-1)

```

```

12 //Results :
13 printf("\n\n Minimum number of teeth for a gear
    ratio of 3:1, t = %d.\n\n",t1+1)
14 printf(" Minimum number of teeth for equal wheel, t
    = %d.\n\n",t2+1)

```

---

**Scilab code Exa 12.10** To find number of teeth

```

1 //To find number of teeth
2 clc
3 //Given:
4 G=4
5 phi=14.5 //degrees
6 //Solution:
7 //Least number of teeth on each wheel:
8 //Calculating the least number of teeth on the
    pinion
9 t=2*%pi/tand(phi)
10 //Calculating the least number of teeth on the gear
11 T=G*t
12 //Results :
13 printf("\n\n Least number of teeth on the pinion , t
    = %d.\n\n",t+1)
14 printf(" Least number of teeth on the gear , T = %d.\n
    \n",T+1)

```

---

**Scilab code Exa 12.11** To find addenda and velocity

```

1 //To find addenda and velocity
2 clc
3 //Given:
4 phi=16 //degrees
5 m=6 //mm

```

```

6 t=16, G=1.75, T=G*t
7 N1=240 //rpm
8 //Solution:
9 //Calculating the angular speed of the pinion
10 omega1=2*pi*N1/60 //rad/s
11 //Addenda on pinion and gear wheel:
12 //Calculating the addendum on pinion
13 addendump=m*t/2*(sqrt(1+T/t*(T/t+2)*(sind(phi))^2)
    -1) //Addendum on pinion, mm
14 //Calculating the addendum on wheel
15 addendumg=m*T/2*(sqrt(1+t/T*(t/T+2)*(sind(phi))^2)
    -1) //Addendum on wheel, mm
16 //Length of path of contact:
17 //Calculating the pitch circle radius of wheel
18 R=m*T/2 //mm
19 //Calculating the pitch circle radius of pinion
20 r=m*t/2 //mm
21 //Calculating the addendum circle radius of wheel
22 RA=R+addendump //mm
23 //Calculating the addendum circle radius of pinion
24 rA=r+addendumg //mm
25 //Calculating the length of path of approach
26 KP=sqrt(RA^2-R^2*(cosd(phi))^2)-R*sind(phi) //mm
27 //Calculating the length of path of recess
28 PL=sqrt(rA^2-r^2*(cosd(phi))^2)-r*sind(phi) //mm
29 //Calculating the length of path of contact
30 KL=KP+PL //mm
31 //Maximum velocity of sliding of teeth on either
    side of pitch point:
32 //Calculating the angular speed of gear wheel
33 omega2=omega1/G //rad/s
34 //Calculating the maximum velocity of sliding of
    teeth on the left side of pitch point
35 vmax1=(omega1+omega2)*KP //Maximum velocity of
    sliding of teeth on the left side of pitch point,
    mm/s
36 //Calculating the maximum velocity of sliding of
    teeth on the right side of pitch point

```



```

37 vmaxr=(omega1+omega2)*PL //Maximum velocity of
    sliding of teeth on the right side of pitch point
    , mm/s
38 //Results:
39 printf("\n\n Addendum on pinion = %.2f mm.\n\n",
    addendump)
40 printf(" Addendum on wheel = %.2f mm.\n\n", addendumg
    )
41 printf(" Length of path of contact , KL = %.2f mm.\n\n\
n", KL)
42 printf(" Maximum velocity of sliding of teeth on the
    left side of pitch point = %d mm/s.\n\n", vmaxl)
43 printf(" Maximum velocity of sliding of teeth on the
    right side of pitch point = %d mm/s.\n\n", vmaxr)

```

---

**Scilab code Exa 12.12** To find sliding velocities and contact ratio

```

1 //To find sliding velocities and contact ratio
2 clc
3 //Given:
4 phi=20 //degrees
5 t=30, T=50, m=4
6 N1=1000 //rpm
7 //Solution:
8 //Calculating the angular speed of thr pinion
9 omega1=2*pi*N1/60 //rad/s
10 //Sliding velocities at engagement and at
    disengagement of a pair of teeth:
11 //Calculating the addendum of the smaller gear
12 addendump=m*t/2*(sqrt(1+T/t*(T/t+2)*(sind(phi))^2)
    -1) //Addendum of the smaller gear, mm
13 //Calculating the addendum of the larger gear
14 addendumg=m*T/2*(sqrt(1+t/T*(t/T+2)*(sind(phi))^2)
    -1) //Addendum of the larger gear, mm
15 //Calculating the pitch circle radius of the smaller

```

```

    gear
16 r=m*t/2 //mm
17 //Calculating the radius of addendum circle of the
    smaller gear
18 rA=r+addendump //mm
19 //Calculating the pitch circle radius of the larer
    gear
20 R=m*T/2 //mm
21 //Calculating the radius of addendum circle of the
    larger gear
22 RA=R+addendumg //mm
23 //Calculating the path of approach
24 KP=sqrt(RA^2-R^2*(cosd(phi))^2)-R*sind(phi) //mm
25 //Calculating the path of recess
26 PL=sqrt(rA^2-r^2*(cosd(phi))^2)-r*sind(phi) //mm
27 //Calculating the angular speed of the larger gear
28 omega2=omega1*t/T //rad/s
29 //Calculating the sliding velocity at engagement of
    a pair of teeth
30 v1=(omega1+omega2)*KP //Sliding velocity at
    engagement of a pair of teeth, mm/s
31 //Calculating the sliding velocity at disengagement
    of a pair of teeth
32 v2=(omega1+omega2)*PL //Sliding velocity at
    disengagement of a pair of teeth, mm/s
33 //Contact ratio:
34 //Calculating the length of the arc of contact
35 Lac=(KP+PL)/cosd(phi) //mm
36 //Calculating the circular pitch
37 pc=%pi*m //Circular pitch, mm
38 //Calculating the contact ratio
39 CR=Lac/pc //Contact ratio
40 //Results:
41 printf("\n\n Sliding velocity at engagement of a
    pair of teeth = %.3f m/s.\n\n",v1/1000)
42 printf(" Sliding velocity at disengagement of a pair
    of teeth = %.3f m/s.\n\n",v2/1000)
43 printf(" Contact ratio = %d.\n\n",CR+1)

```

---

**Scilab code Exa 12.13** To find teeth and velocity

```
1 //To find teeth and velocity
2 clc
3 //Given:
4 G=3
5 m=6, AP=1*m, AW=AP //mm
6 phi=20 //degrees
7 N1=90 //rpm
8 //Solution:
9 //Calculating the angular speed of the pinion
10 omega1=2*%pi*N1/60 //rad/s
11 //Calculating the number of teeth on the pinion to
    avoid interference on it
12 t=2*AP/(sqrt(1+G*(G+2)*(sind(phi))^2)-1)
13 //Calculating the corresponding number of teeth on
    the wheel
14 T=G*t
15 //Length of path and arc of contact:
16 //Calculating the pitch circle radius of pinion
17 r=m*t/2 //mm
18 //Calculating the radius of addendum circle of
    pinion
19 rA=r+AP //mm
20 //Calculating the pitch circle radius of wheel
21 R=m*T/2 //mm
22 //Calculating the radius of addendum circle of wheel
23 RA=R+AW //mm
24 //Calculating the path of approach
25 KP=sqrt(RA^2-R^2*(cosd(phi))^2)-R*sind(phi) //mm
26 //Calculating the path of recess
27 PL=sqrt(rA^2-r^2*(cosd(phi))^2)-r*sind(phi) //mm
28 //Calculating the length of path of contact
29 KL=KP+PL //mm
```

```

30 //Calculating the length of arc of contact
31 Lac=KL/cosd(phi) //Length of arc of contact , mm
32 //Number of pairs of teeth in contact:
33 //Calculating the circular pitch
34 pc=%pi*m //mm
35 //Calculating the number of pairs of teeth in
    contact
36 n=Lac/pc //Number of pairs of teeth in contact
37 //Maximum velocity of sliding:
38 //Calculating the angular speed of wheel
39 omega2=omega1*t/T //rad/s
40 //Calculating the maximum velocity of sliding
41 vs=(omega1+omega2)*KP //mm/s
42 //Results:
43 printf("\n\n Number of teeth on the pinion to avoid
    interference , t = %d.\n\n",t+1)
44 printf(" Corresponding number of teeth on the wheel,
    T = %d.\n\n",T+1)
45 printf(" Length of path of contact , KL = %.2f mm.\n\n\
n",KL)
46 printf(" Length of arc of contact = %.2f mm.\n\n",
    Lac)
47 printf(" Number of pairs of teeth in contact = %d.\n
\n",n+1)
48 printf(" Maximum velocity of sliding , vs = %.2f mm/s
.\n\n",vs)

```

---

**Scilab code Exa 12.14** To find pressure angle and teeth

```

1 //To find pressure angle and teeth
2 clc
3 //Given:
4 T=20
5 d=125, r=d/2, OP=r, LH=6.25 //mm
6 //Calculating the least pressure angle to avoid

```

```

interference
7 phi=asin(sqrt(LH/r))*180/%pi //degrees
8 //Length of arc of contact:
9 //Calculating the length of path of contact
10 KL=sqrt((OP+LH)^2-(OP*cosd(phi))^2) //mm
11 //Calculating the length of arc of contact
12 Lac=KL/cosd(phi) //Length of arc of contact, mm
13 //Minimum number of teeth:
14 //Calculating the circular pitch
15 pc=%pi*d/T //mm
16 //Calculating the number of pairs of teeth in
    contact
17 n=Lac/pc //Number of pairs of teeth in contact
18 //Calculating the minimum number of teeth in contact
19 nmin=n //Minimum number of teeth in contact
20 //Results:
21 printf("\n\n Least pressure angle to avoid
    interference , phi = %.3f degrees.\n\n",phi)
22 printf(" Length of arc of contact = %.2f mm.\n\n",
    Lac)
23 printf(" Minimum number of teeth in contact = %d or
    %d pair.\n\n",nmin+1,(nmin+1)/2)

```

---

**Scilab code Exa 12.15** To find axial thrust

```

1 //To find axial thrust
2 clc
3 //Given:
4 L=175/1000, d2=100/1000, r2=d2/2 //m
5 theta=70 //degrees
6 G=1.5, T2=80
7 Tf=75 //Torque on faster wheel, N-m
8 funcprot(0)
9 //Solution:
10 //Spiral angles for each wheel:

```

```

11 //Calculating the number of teeth on slower wheel
12 T1=T2*G
13 //Calculating the pitch circle diameter of the
    slower wheel
14 d1=(L*2)-d2 //m
15 //Calculating the spiral angles
16 //We have,  $d2/d1 = (T2*\cos(\alpha1))/(T1*\cos(\alpha2))$ 
    , or  $T2*d1*\cos(\alpha1)-T1*d2*\cos(\alpha2) = 0$ 
    .....( i)
17 //Also,  $\alpha1+\alpha2 = \theta$ , or  $\alpha1+\alpha2-\theta = 0$ 
    .....(
    ii)
18 function y=f(x)
19     alpha1=x(1)
20     alpha2=x(2)
21     y(1)=T2*d1*cos(alpha1)-T1*d2*cos(alpha2)
22     y(2)=alpha1+alpha2-theta*pi/180
23 endfunction
24 z=fsolve([1,1],f)
25 alpha1=z(1)*180/pi //Spiral angle for slower wheel,
    degrees
26 alpha2=z(2)*180/pi //Spiral angle for faster wheel,
    degrees
27 //Axial thrust on each shaft:
28 //Calculating the tangential force at faster wheel
29 F2=Tf/r2 //N
30 //Calculating the normal reaction at the point of
    contact
31 RN=F2/cosd(alpha2) //N
32 //Calculating the axial thrust on the shaft of
    slower wheel
33 Fa1=RN*sind(alpha1) //N
34 //Calculating the axial thrust on the shaft of
    faster wheel
35 Fa2=RN*sind(alpha2) //N
36 //Results:
37 printf("\n\n Spiral angle for slower wheel, alpha1 =

```

```

    %.2f degrees.\n\n",alpha1)
38 printf(" Spiral angle for faster wheel, alpha2 = %.2
    f degrees.\n\n",alpha2)
39 printf(" Axial thrust on the shaft of slower wheel ,
    Fa1= %d N.\n\n",Fa1+1)
40 printf(" Axial thrust on the shaft of faster wheel ,
    Fa2 = %d N.\n\n",Fa2+1)

```

---

**Scilab code Exa 12.16** To find teeth distance and efficiency

```

1 //To find teeth , distance and efficiency
2 clc
3 //Given:
4 L=400/1000 //m
5 G=3
6 theta=50, phi=6 //degrees
7 pN=18 //mm
8 //Solution:
9 //Number of teeth on each wheel:
10 //Calculating the spiral angles of the driving and
    driven wheels
11 alpha1=theta/2 //degrees
12 alpha2=alpha1 //degrees
13 //Calculating the number of teeth on driver wheel
14 T1=L*1000*2*%pi/(pN*(1/cosd(alpha1)+G/cosd(alpha2)))
15 //Calculating the number of teeth on driven wheel
16 T2=G*T1
17 //Calculating the exact centre distance
18 L1=pN*T1/(2*%pi)*(1/cosd(alpha1)+G/cosd(alpha2)) //
    mm
19 //Calculating the efficiency of the drive
20 eta=(cosd(alpha2+phi)*cosd(alpha1))/(cosd(alpha1-phi)
    *cosd(alpha2))*100 //%
21 //Results:
22 printf("\n\n Number of teeth on driver wheel, T1 =

```

```

    %d.\n\n",T1+1)
23 printf(" Number of teeth on driven wheel, T2 = %d.\n
    \n", T2+1)
24 printf(" Exact centre distance, L1 = %.1f mm.\n\n",
    L1)
25 printf(" Efficiency of the drive, eta = %.1f %c.\n\n
    ",eta,"%")

```

---

**Scilab code Exa 12.17** To find angle teeth and efficiency

```

1 //To find angle , teeth and efficiency
2 clc
3 //Given:
4 pN=12.5, L=134 //mm
5 theta=80, phi=6 //degrees
6 G=1.25
7 //Solution:
8 funcprot(0)
9 //Spiral angle of each wheel:
10 //Calculating the spiral angles of wheels 1 and 2
11 //We have,  $d_2/d_1 = (T_2 \cos(\alpha_1))/(T_1 \cos(\alpha_2))$ 
    , or  $\cos(\alpha_1) - G \cos(\alpha_2) = 0$ 
    .....(i)
12 //Also,  $\alpha_1 + \alpha_2 = \theta$ , or  $\alpha_1 + \alpha_2 -$ 
    theta = 0
    .....(
    ii)
13 function y=f(x)
14     alpha1=x(1)
15     alpha2=x(2)
16     y(1)=cos(alpha1)-G*cos(alpha2)
17     y(2)=alpha1+alpha2-theta*pi/180
18 endfunction
19 z=fsolve([1,1],f)
20 alpha1=z(1)*180/pi //Spiral angle for slower wheel,

```



```

degrees
21 alpha2=z(2)*180/%pi //Spiral angle for faster wheel,
degrees
22 //Number of teeth on each wheel:
23 //Calculating the diameters of the wheels
24 d1=L, d2 = d1 //mm
25 //Calculating the number of teeth on wheel 1
26 T1=d1*%pi*cosd(alpha1)/pN
27 //Calculating the number of teeth on wheel 2
28 T2=T1/G
29 //Calculating the efficiency of the drive
30 eta=(cosd(alpha2+phi)*cosd(alpha1))/(cosd(alpha1-phi
) *cosd(alpha2))*100 //%
31 //Calculating the maximum efficiency
32 etamax=(cosd(theta+phi)+1)/(cosd(theta-phi)+1)*100
//%
33 //Results:
34 printf("\n\n Spiral angle for slower wheel, alpha1 =
%.2f degrees.\n\n",alpha1)
35 printf(" Spiral angle for faster wheel, alpha2 = %.2
f degrees.\n\n",alpha2)
36 printf(" Number of teeth on wheel 1, T1 = %d.\n\n",
T1+1)
37 printf(" Number of teeth on wheel 2, T2 = %d.\n\n",
T2+1)
38 printf(" Efficiency of the drive, eta = %d %c.\n\n",
eta+1,"%")
39 printf(" Maximum efficiency , etamax = %.1f %c.\n\n",
etamax,"%")

```

---

# Chapter 13

## Gear Trains

Scilab code Exa 13.1 To find speed of gear F

```
1 //To find speed of gear F
2 clc
3 //Given:
4 NA=975 //rpm
5 TA=20, TB=50, TC=25, TD=75, TE=26, TF=65
6 //Solution:
7 //Calculating the speed of gear F
8 NF=NA*(TA*TC*TE)/(TB*TD*TF) //rpm
9 //Results:
10 printf("\n\n Speed of gear F, NF = %d rpm.\n\n",NF)
```

---

Scilab code Exa 13.2 To design the gears

```
1 //To design the gears
2 clc
3 //Given:
4 x=600, pc=25 //mm
5 N1=360, N2=120 //rpm
```

```

6 //Solution:
7 //Calculating the pitch circle diameters of each
  gear
8 //Speed ratio ,  $N1/N2 = d2/d1$ , or  $N1*d1-N2*d2 = 0$ 
  .....( i)
9 //Centre distance between the shafts ,  $x = 1/2*(d1+d2$ 
  ), or  $d1+d2 = 600*2$  .....( ii)
10 A=[N1 -N2; 1 1]
11 B=[0; 600*2]
12 V=A \ B
13 d1=V(1) //mm
14 d2=V(2) //mm
15 //Calculating the number of teeth on the first gear
16 T1=round(%pi*d1/pc)
17 //Calculating the number of teeth on the second gear
18 T2=int(%pi*d2/pc+1)
19 //Calculating the pitch circle diameter of the first
  gear
20 d1dash=T1*pc/%pi //mm
21 //Calculating the pitch circle diameter of the
  second gear
22 d2dash=T2*pc/%pi //mm
23 //Calculating the exact distance between the two
  shafts
24 xdash=(d1dash+d2dash)/2 //mm
25 //Results:
26 printf("\n\n The number of teeth on the first and
  second gear must be %d and %d and their pitch
  circle diameters must be %.2f mm and %.1f mm
  respectively.\n\n",T1,T2,d1dash,d2dash)
27 printf(" The exact distance between the two shafts
  must be %.2f mm.\n\n",xdash)

```

---

**Scilab code Exa 13.3** To find the number of teeth

```

1 //To find the number of teeth
2 clc
3 //Given:
4 rAD=12 //Speed ratio , NA/ND
5 mA=3.125, mB=mA, mC=2.5, mD=mC, x=200 //mm
6 //Solution:
7 //Calculating the speed ratio between the gears A
  and B, and C and D
8 rAB=sqrt(rAD) //Speed ratio between the gears A and
  B
9 rCD=sqrt(rAB) //Speed ratio between the gears C and
  D
10 //Calculating the ratio of teeth on gear B to gear A
11 rtBA=rAB //Ratio of teeth on gear B to gear A
12 //Calculating the ratio of teeth on gear D to gear C
13 rtDC=rCD //Ratio of teeth on gear D to gear C
14 //Calculating the number of teeth on the gears A and
  B
15 //Distance between the shafts ,  $x = mA*TA/2+mB*TB/2$ ,
  or  $(mA/2)*TA+(mB/2)*TB = x$  .....( i )
16 //Ratio of teeth on gear B to gear A,  $TB/TA = \text{sqrt}$ 
  (12), or  $\text{sqrt}(12)*TA-TB = 0$  .....( ii )
17 A=[mA/2 mB/2; sqrt(12) -1]
18 B=[x; 0]
19 V=A \ B
20 TA=int(V(1))
21 TB=round(V(2))
22 //Calculating the number of teeth on the gears C and
  D
23 //Distance between the shafts ,  $x = mC*TC/2+mD*TD/2$ ,
  or  $(mC/2)*TC+(mD/2)*TD = x$  .....( iii )
24 //Ratio of teeth on gear D to gear C,  $TD/TC = \text{sqrt}$ 
  (12), or  $\text{sqrt}(12)*TC-TD = 0$  .....( iv )
25 A=[mC/2 mD/2; sqrt(12) -1]
26 B=[x; 0]
27 V=A \ B
28 TC=round(V(1))
29 TD=int(V(2))

```

```

30 //Results:
31 printf("\n\n Number of teeth on gear A, TA = %d.\n\n
    ",TA)
32 printf(" Number of teeth on gear B, TB = %d.\n\n",TB
    )
33 printf(" Number of teeth on gear C, TC = %d.\n\n",TC
    )
34 printf(" Number of teeth on gear D, TD = %d.\n\n",TD
    )

```

---

**Scilab code Exa 13.4** To find speed of gear B

```

1 //To find speed of gear B
2 clc
3 //Given:
4 TA=36, TB=45
5 NC=150 //rpm, anticlockwise
6 //Solution:
7 //Refer Fig. 13.7
8 //Algebraic method:
9 //Calculating the speed of gear B when gear A is
    fixed
10 NA=0, NC=150 //rpm
11 NB1=(-TA/TB)*(NA-NC)+NC //rpm
12 //Calculating the speed of gear B when gear A makes
    300 rpm clockwise
13 NA=-300 //rpm
14 NB2=(-TA/TB)*(NA-NC)+NC //rpm
15 //Results:
16 printf("\n\n Speed of gear B when gear A is fixed ,
    NB = %d rpm.\n\n",NB1)
17 printf(" Speed of gear B when gear A makes 300 rpm
    clockwise , NB = %d rpm.\n\n",NB2)

```

---

**Scilab code Exa 13.5** To find speed of gear C

```
1 //To find speed of gear C
2 clc
3 //Given:
4 TB=75, TC=30, TD=90
5 NA=100 //rpm, clockwise
6 //Solution:
7 //Refer Table 13.3
8 //Calculating the number of teeth on gear E
9 TE=TC+TD-TB
10 //Calculating the speed of gear C
11 y=-100
12 x=y*(TB/TE)
13 NC=y-x*(TD/TC) //rpm
14 //Results:
15 printf("\\n\\n Speed of gear C, NC = %d rpm,
        anticlockwise.\\n\\n",NC)
```

---

**Scilab code Exa 13.6** To find speed of gears B and C

```
1 //To find speed of gears B and C
2 clc
3 //Given:
4 TA=72, TC=32
5 NEF=18 //Speed of arm EF, rpm
6 //Solution:
7 //Refer Table 13.5
8 //Speed of gear C:
9 y=18 //rpm
10 x=y*(TA/TC)
11 NC=x+y //Speed of gear C, rpm
```

```

12 //Speed of gear B:
13 //Calculating the number of teeth on gear B
14 TB=(TA-TC)/2
15 //Calculating the speed of gear B
16 NB=y-x*(TC/TB) //Speed of gear B, rpm
17 //Solution:
18 printf("\n\n Speed of gear C = %.1f rpm.\n\n",NC)
19 printf(" Speed of gear B = %.1f rpm in the opposite
    direction of arm.\n\n",-NB)

```

---

**Scilab code Exa 13.7** To find revolutions of arm

```

1 //To find revolutions of arm
2 clc
3 //Given:
4 TA=40, TD=90
5 //Solution:
6 //Calculating the number of teeth on gears B and C
7 //From geometry of the Fig. 13.11,  $d_A+2*d_B=d_D$ .
8 //Since the number of teeth are proportional to
    their pitch circle diameters,
9 TB=(TD-TA)/2
10 TC=TB
11 //Refer Table 13.6
12 //Speed of arm when A makes 1 revolution clockwise
    and D makes half revolution anticlockwise:
13 //Calculating the values of x and y
14 //From the fourth row of the table,  $-x-y = -1$ , or  $x+$ 
     $y = 1$  .....(i)
15 //The gear D makes half revolution anticlockwise, i.
    e.,  $x*(TA/TD)-y = 1/2$  .....(ii)
16 A=[1 1; TA/TD -1]
17 B=[1; 1/2]
18 V=A \ B
19 x=V(1)

```

```

20 y=V(2)
21 //Calculating the speed of arm
22 varm=-y //Speed of arm, revolutions
23 //Results:
24 printf("\n\n Speed of arm when A makes 1 revolution
        clockwise and D makes half revolution
        anticlockwise = %.2f revolution anticlockwise.\n\n",
        varm)
25 //Speed of arm when A makes 1 revolution clockwise
        and D is stationary:
26 //Calculating the values of x and y
27 //From the fourth row of the table,  $-x-y = -1$ , or  $x+$ 
         $y = 1$  ..... (iii)
28 //The gear D is stationary, i.e.,  $x*(TA/TD)-y = 0$ 
        ..... (iv)

29 A=[1 1; TA/TD -1]
30 B=[1; 0]
31 V=A \ B
32 x=V(1)
33 y=V(2)
34 //Calculating the speed of arm
35 varm=-y //Speed of arm, revolutions
36 //Results:
37 printf(" Speed of arm when A makes 1 revolution
        clockwise and D is stationary = %.3f revolution
        clockwise.\n\n",-varm)

```

---

**Scilab code Exa 13.8** To find teeth and speed

```

1 //To find teeth and speed
2 clc
3 //Given:
4 TC=28, TD=26, TE=18, TF=TE
5 //Solution:
6 //The sketch is as in Fig. 13.12

```



```

7 //Number of teeth on wheels A and B:
8 //From geometry,  $d_A = d_C + 2*d_E$ , and  $d_B = d_D + 2*d_F$ 
9 //Since the number of teeth are proportional to
  their pitch circle diameters,
10  $T_A = T_C + 2*T_E$ 
11  $T_B = T_D + 2*T_F$ 
12 //Speed of wheel B when arm G makes 100 rpm
  clockwise and wheel A is fixed:
13 //Since the arm G makes 100 rpm clockwise, therefore
  from the fourth row of Table 13.7,
14  $y = -100$ 
15  $x = -y$ 
16 //Calculating the speed of wheel B
17  $N_{B1} = y + x*(T_A/T_C)*(T_D/T_B)$  //Speed of wheel B when arm
  G makes 100 rpm clockwise and wheel A is fixed,
  rpm
18 //Speed of wheel B when arm G makes 100 rpm
  clockwise and wheel A makes 10 rpm counter
  clockwise:
19 //Since the arm G makes 100 rpm clockwise, therefore
  from the fourth row of Table 13.7,
20  $y = -100$ 
21  $x = 10 - y$ 
22 //Calculating the speed of wheel B
23  $N_{B2} = y + x*(T_A/T_C)*(T_D/T_B)$  //Speed of wheel B when arm
  G makes 100 rpm clockwise and wheel A makes 10
  rpm counter clockwise, rpm
24 //Solution:
25 printf("\\n\\n Number of teeth on wheel A,  $T_A = %d$ .\\n
  \\n",  $T_A$ )
26 printf(" Number of teeth on wheel B,  $T_B = %d$ .\\n\\n",
   $T_B$ )
27 printf(" Speed of wheel B when arm G makes 100 rpm
  clockwise and wheel A is fixed =  $%.1f$  rpm,
  clockwise.\\n\\n",  $-N_{B1}$ )
28 printf(" Speed of wheel B when arm G makes 100 rpm
  clockwise and wheel A makes 10 rpm counter
  clockwise =  $%.1f$  rpm, counter clockwise.\\n\\n",  $N_{B2}$ )

```

)

---

**Scilab code Exa 13.9** To find number of teeth

```
1 //To find number of teeth
2 clc
3 //Given:
4 dD=224, m=4 //mm
5 //Solution:
6 //Refer Table 13.8
7 //Calculating the values of x and y
8 y=+1
9 x=+5-y
10 //Calculating the number of teeth on gear D
11 TD=dD/m
12 //Calculating the number of teeth on gear B
13 TB=y/x*TD
14 //Calculating the number of teeth on gear C
15 TC=(TD-TB)/2
16 //Results:
17 printf("\n\n Number of teeth on gear D, TD = %d.\n\n",TD)
18 printf(" Number of teeth on gear B, TB = %d.\n\n",TB)
19 printf(" Number of teeth on gear C, TC = %d.\n\n",TC)
```

---

**Scilab code Exa 13.10** To find angular velocities

```
1 //To find angular velocities
2 clc
3 //Given:
4 TC=50, TD=20, TE=35
```

```

5 NA=110 //rpm
6 //Solution:
7 //Calculating the number of teeth on internal gear G
8 TG=TC+TD+TE
9 //Speed of shaft B:
10 //Calculating the values of x and y
11 //From the fourth row of Table 13.9,  $y-x*(TC/TD)*(TE$ 
    //TG) = 0 .....( i)
12 //Also,  $x+y = 110$ , or  $y+x = 110$ 
    .....( ii)

13 A=[1 -(TC/TD)*(TE/TG); 1 1]
14 B=[0; 110]
15 V=A \ B
16 x=V(2)
17 y=V(1)
18 //Calculating the speed of shaft B
19 NB=round(+y) //Speed of shaft B, rpm
20 //Results:
21 printf("\n\n Number of teeth on internal gear G, TG
    = %d.\n\n",TG)
22 printf(" Speed of shaft B = %d rpm, anticlockwise.\n
    \n",NB)

```

---

**Scilab code Exa 13.11** To find angular velocities

```

1 //To find angular velocities
2 clc
3 //Given:
4 TA=12, TB=30, TC=14
5 NA=1, ND=5 //rps
6 //Solution:
7 //Number of teeth on wheels D and E:
8 //Calculating the number of teeth on wheel E
9 TE=TA+2*TB
10 //Calculating the number of teeth on wheel E

```

```

11 TD=TE-(TB-TC)
12 //Magnitude and direction of angular velocities of
    arm OP and wheel E:
13 //Calculating the values of x and y
14 //From the fourth row of Table 13.10,  $-x-y = -1$ , or
     $x+y = 1$  ..... ( i )
15 //Also,  $x*(TA/TB)*(TC/TD)-y = 5$ 
                                                    ..... ( ii )

16 A=[1 1; (TA/TB)*(TC/TD) -1]
17 B=[1; 5]
18 V=A \ B
19 x=V(1)
20 y=V(2)
21 //Calculating the angular velocity of arm OP
22 omegaOP=-y*2*pi //Angular velocity of arm OP, rad/s
23 //Calculating the angular velocity of wheel E
24 omegaE=(x*TA/TE-y)*2*pi //Angular velocity of wheel
    E, rad/s
25 //Results:
26 printf("\n\n Number of teeth on wheel E, TE = %d.\n\n",
    TE)
27 printf(" Number of teeth on wheel D, TD = %d.\n\n",
    TD)
28 printf(" Angular velocity of arm OP = %.3f rad/s,
    counter clockwise.\n\n",omegaOP)
29 printf(" Angular velocity of wheel E = %.2f rad/s,
    counter clockwise.\n\n",omegaE)

```

---

**Scilab code Exa 13.12** To find speed of shaft

```

1 //To find speed of shaft
2 clc
3 //Given:
4 TB=80, TC=82, TD=28
5 NA=500 //rpm

```

```

6 //Solution:
7 //Calculating the number of teeth on wheel E
8 TE=TB+TD-TC
9 //Calculating the values of x and y
10 y=800
11 x=-y*(TE/TB)*(TC/TD)
12 //Calculating the speed of shaft F
13 NF=x+y //Speed of shaft F, rpm
14 //Results:
15 printf("\n\n Speed of shaft F = %d rpm,
        anticlockwise.\n\n",NF)

```

---

**Scilab code Exa 13.14** To find number of teeth and speed

```

1 //To find number of teeth and speed
2 clc
3 //Given:
4 NA=300 //rpm
5 TD=40, TE=30, TF=50, TG=80, TH=40, TK=20, TL=30
6 //Solution:
7 //Refer Fig. 13.18 and Table 13.13
8 //Calculating the speed of wheel E
9 NE=NA*(TD/TE) //rpm
10 //Calculating the number of teeth on wheel C
11 TC=TH+TK+TL
12 //Speed and direction of rotation of shaft B:
13 //Calculating the values of x and y
14 //We have,  $-x-y = -400$ , or  $x+y = 400$ 
        .....( i)
15 //Also,  $x*(TH/TK)*(TL/TC)-y = 0$ 
        .....( ii)
16 A=[1 1; (TH/TK)*(TL/TC) -1]
17 B=[400; 0]
18 V=A \ B
19 x=V(1)

```

```

20 y=V(2)
21 //Calculating the speed of wheel F
22 NF=-y //rpm
23 //Calculating the speed of shaft B
24 NB=-NF*(TF/TG) //Speed of shaft B, rpm
25 //Results:
26 printf("\n\n Number of teeth on wheel C, TC = %d.\n\n",TC)
27 printf(" Speed of shaft B = %d rpm, anticlockwise.\n\n",NB)

```

---

**Scilab code Exa 13.15** To find velocity ratio

```

1 //To find velocity ratio
2 clc
3 //Given:
4 T1=80, T8=160, T4=100, T3=120, T6=20, T7=66
5 //Solution:
6 //Refer Fig. 13.19 and Table 13.14
7 //Calculating the number of teeth on wheel 2
8 T2=(T3-T1)/2
9 //Calculating the values of x and y
10 //Assuming that wheel 1 makes 1 rps anticlockwise, x
    +y = 1 ..... (i)
11 //Also,  $y-x*(T1/T3) = 0$ , or  $x*(T1/T3)-y = 0$ 
    ..... (ii)

12 A=[1 1; 1 T1/T3]
13 B=[1; 0]
14 V=A \ B
15 x=V(1)
16 y=V(2)
17 //Calculating the speed of casing C
18 NC=y //Speed of casing C, rps
19 //Calculating the speed of wheel 2
20 N2=y-x*(T1/T2) //Speed of wheel 2, rps

```

```

21 //Calculating the number of teeth on wheel 5
22 T5=(T4-T6)/2
23 //Calculating the values of x1 and y1
24 y1=-2
25 x1=(y1-0.4)*(T4/T6)
26 //Calculating the speed of wheel 6
27 N6=x1+y1 //Speed of wheel 6, rps
28 //Calculating the values of x2 and y2
29 y2=0.4
30 x2=-(14+y2)*(T7/T8)
31 //Calculating the speed of wheel 8
32 N8=x2+y2 //Speed of wheel 8, rps
33 //Calculating the velocity ratio of the output shaft
    B to the input shaft A
34 vr=N8/1 //Velocity ratio
35 //Results:
36 printf("\n\n Velocity ratio of the output shaft B to
    the input shaft A = %.2f.\n\n",vr)

```

---

**Scilab code Exa 13.16** To find speed of shaft

```

1 //To find speed of shaft
2 clc
3 //Given:
4 TA=40, TB=30, TC=50
5 NX=100, NA=NX //rpm
6 Narm=100 //Speed of arm, rpm
7 //Solution:
8 //Refer Fig. 13.22 and Table 13.18
9 //Calculating the values of x and y
10 y=+100
11 x=-100-y
12 //Calculating the speed of the driven shaft
13 NY=y-x*(TA/TB) //rpm
14 //Results:

```

```

15 printf("\n\n Speed of the driven shaft , NY = %.1 f
    rpm, anticlockwise.\n\n",NY)

```

---

**Scilab code Exa 13.17** To find speed of output shaft

```

1 //To find speed of output shaft
2 clc
3 //Given:
4 TB=20, TC=80, TD=80, TE=30, TF=32
5 NB=1000 //rpm
6 //Solution:
7 //Refer Fig. 13.23 and Table 13.19
8 //Speed of the output shaft when gear C is fixed:
9 //Calculating the values of x and y
10 //From the fourth row of the table ,  $y-x*(TB/TC) = 0$ 
    .....( i)
11 //Also ,  $x+y = +1000$ , or  $y+x = 1000$ 
    .....( ii)
12 A=[1 -TB/TC; 1 1]
13 B=[0; 1000]
14 V=A \ B
15 x=V(2)
16 y=V(1)
17 //Calculating the speed of output shaft
18 NF1=y-x*(TB/TD)*(TE/TF) //Speed of the output shaft
    when gear C is fixed , rpm
19 //Speed of the output shaft when gear C is rotated
    at 10 rpm counter clockwise:
20 //Calculating the values of x and y
21 //From the fourth row of te table ,  $y-x*(TB/TC) = +10$ 
    .....( iii)
22 //Also ,  $x+y = +1000$ , or  $y+x = 1000$ 
    .....( iv)
23 A=[1 -TB/TC; 1 1]
24 B=[10; 1000]

```



```

25 V=A \ B
26 x=V(2)
27 y=V(1)
28 //Calculating the speed of output shaft
29 NF2=y-x*(TB/TD)*(TE/TF) //Speed of the output shaft
    when gear C is rotated at 10 rpm counter
    clockwise , rpm
30 //Results:
31 printf("\n\n Speed of the output shaft when gear C
    is fixed = %.1f rpm, counter clockwise.\n\n",NF1)
32 printf(" Speed of the output shaft when gear C is
    rotated at 10 rpm counter clockwise = %.1f rpm,
    counter clockwise.\n\n",NF2)

```

---

**Scilab code Exa 13.18** To find speed of road wheel

```

1 //To find speed of road wheel
2 clc
3 //Given:
4 TA=10, TB=60
5 NA=1000, NQ=210, ND=NQ //rpm
6 //Solution:
7 //Refer Fig. 13.24 and Table 13.20
8 //Calculating the speed of crown gear B
9 NB=NA*(TA/TB) //rpm
10 //Calculating the values of x and y
11 y=200
12 x=y-210
13 //Calculating the speed of road wheel attached to
    axle P
14 NC=x+y //Speed of road wheel attached to axle P, rpm
15 //Results:
16 printf("\n\n Speed of road wheel attached to axle P
    = %d rpm.\n\n",NC)

```

---

**Scilab code Exa 13.19** To find torque exerted

```
1 //To find torque exerted
2 clc
3 //Given:
4 TA=15, TB=20, TC=15
5 NA=1000 //rpm
6 Tm=100 //Torque developed by motor, N-m
7 //Solution:
8 //Refer Fig. 13.26 and Table 13.21
9 //Calculating the number of teeth on gears E and D
10 TE=TA+2*TB
11 TD=TE-(TB-TC)
12 //Speed of the machine shaft:
13 //From the fourth row of the table,  $x+y = 1000$ , or  $y$ 
     $+x = 1000$  .....( i)
14 //Also,  $y-x*(TA/TE) = 0$ 
    .....( ii)
15 A=[1 1; 1 -TA/TE]
16 B=[1000; 0]
17 V=A \ B
18 y=V(1)
19 x=V(2)
20 //Calculating the speed of machine shaft
21 ND=y-x*(TA/TB)*(TC/TD) //rpm
22 //Calculating the torque exerted on the machine
    shaft
23 Ts=Tm*NA/ND //Torque exerted on the machine shaft, N
    -m
24 //Results:
25 printf(" \n\n Speed of machine shaft, ND = %.2 f rpm,
    anticlockwise.\n\n",ND)
26 printf(" Torque exerted on the machine shaft = %d N-
```

m.\n\n",Ts)

---

**Scilab code Exa 13.20** To find teeth and torque

```
1 //To find teeth and torque
2 clc
3 //Given:
4 Ts=100 //Torque on the sun wheel, N-m
5 r=5 //Ratio of speeds of gear S to C, NS/NC
6 //Refer Fig. 13.27 and Table 13.22
7 //Number of teeth on different wheels:
8 //Calculating the values of x and y
9 y=1
10 x=5-y
11 //Calculating the number of teeth on wheel E
12 TE=16
13 TE=4*TS
14 //Calculating the number of teeth on wheel P
15 TP=(TE-TS)/2
16 //Torque necessary to keep the internal gear
    stationary:
17 Tc=Ts*r //Torque on CN-m
18 //Calculating the torque necessary to keep the
    internal gear stationary
19 Ti=Tc-Ts //Torque necessary to keep the internal
    gear stationary, N-m
20 //Results:
21 printf(" \n\n Number of teeth on different wheels, TE
    = %d.\n\n",TE)
22 printf(" Torque necessary to keep the internal gear
    stationary = %d N-m.\n\n",Ti)
```

---

**Scilab code Exa 13.21** To find speed direction and torque

```

1 //To find speed, direction and torque
2 clc
3 //Given:
4 TA=14, TC=100
5 r=98/41 //TE/TD
6 PA=1.85*1000 //W
7 NA=1200 //rpm
8 //Solution:
9 //Refer Fig. 13.28 and Table 13.23
10 //Calculating the number of teeth on wheel B
11 TB=(TC-TA)/2
12 //Calculating the values of x and y
13 //From the fourth row of the table,  $-y+x*(TA/TC) =$ 
    0, or  $x*(TA/TC)-y = 0$  .....( i)
14 //Also,  $-x-y = 1200$ , or  $x+y = -1200$ 
    .....( ii)

15 A=[TA/TC -1; 1 1]
16 B=[0; -1200]
17 V=A \ B
18 x=V(1)
19 y=V(2)
20 //Calculating the speed of gear E
21 NE=round(-y+x*(TA/TB)*(1/r)) //rpm
22 //Fixing torque required at C:
23 //Calculating the torque on A
24 Ta=PA*60/(2*pi*NA) //Torque on A, N-m
25 //Calculating the torque on E
26 Te=PA*60/(2*pi*NE) //Torque on E
27 //Calculating the fixing torque required at C
28 Tc=Te-Ta //Fixing torque at C, N-m
29 //Results:
30 printf("\\n\\n Speed and direction of rotation of gear
    E, NE = %d rpm, anticlockwise.\\n\\n",NE)
31 printf(" Fixing torque required at C = %.1f N-m.\\n\\n
    ",Tc)

```

---

Scilab code Exa 13.22 To find holding torque

```
1 //To find holding torque
2 clc
3 //Given:
4 TB=15, TA=60, TC=20
5 omegaY=740, omegaA=omegaY //rad/s
6 P=130*1000 //W
7 //Solution:
8 //Refer Fig. 13.29 and Table 13.24
9 //Calculating the number of teeth on wheel D
10 TD=TA-(TC+TB)
11 //Calculating the values of x and y
12 //From the fourth row of the table,  $y-x*(TD/TC)*(TB/TA) = 740$  .....( i)
13 //Also,  $x+y = 0$ , or  $y+x = 0$ 
    .....( ii)
14 A=[1 -(TD/TC)*(TB/TA); 1 1]
15 B=[740; 0]
16 V=A \ B
17 x=V(2)
18 y=V(1)
19 //Calculating the speed of shaft X
20 omegaX=y //rad/s
21 //Holding torque on wheel D:
22 //Calculating the torque on A
23 Ta=P/omegaA //Torque on A, N-m
24 //Calculating the torque on X
25 Tx=P/omegaX //Torque on X, N-m
26 //Calculating the holding torque on wheel D
27 Td=Tx-Ta //Holding torque on wheel D, N-m
28 //Results:
29 printf("\n\n Speed of shaft X, omegaX = %.1f rad/s.\n")
```

```

n\n", omegaX)
30 printf(" Holding torque on wheel D = %.1f N-m.\n\n",
Td)

```

---

**Scilab code Exa 13.23** To find speed direction and torque

```

1 //To find speed, direction and torque
2 clc
3 //Given:
4 TP=144, TQ=120, TR=120, TX=36, TY=24, TZ=30
5 NI=1500 //rpm
6 P=7.5*1000 //W
7 eta=0.8
8 //Solution:
9 //Refer Fig. 13.30 and Table 13.25
10 //Calculating the values of x and y
11 //From the fourth row of the table,  $x+y = -1500$ 
12 //Also,  $y-x*(TZ/TR) = 0$ , or  $-x*(TZ/TR)+y = 0$ 
13 //.....( i)
14 //.....( ii)
15 A=[1 1; -TZ/TR 1]
16 B=[-1500; 0]
17 V=A \ B
18 x=V(1)
19 y=V(2)
20 //Calculating the values of x1 and y1
21 //We have,  $y1-x1*(TY/TQ) = y$ 
22 //.....( iii)
23 //Also,  $x1+y1 = x+y$ , or  $y1+x1 = x+y$ 
24 //.....( iv)
25 A=[1 -TY/TQ; 1 1]
26 B=[y; x+y]
27 V=A \ B
28 x1=V(2)
29 y1=V(1)

```

```

26 //Speed and direction of the driven shaft O and the
    wheel P:
27 //Calculating the speed of shaft O
28 NO=y1 //rpm
29 //Calculating the speed of wheel P
30 NP=y1+x1*(TY/TQ)*(TX/TP) //rpm
31 //Torque tending to rotate the fixed wheel R:
32 //Calculating the torque on shaft I
33 T1=P*60/(2*pi*NI) //N-m
34 //Calculating the torque on shaft O
35 T2=eta*P*60/(2*pi*(-NO)) //N-m
36 //Calculating the torque tending to rotate the fixed
    wheel R
37 T=T2-T1 //Torque tending to rotate the fixed wheel R
    , N-m
38 //Results:
39 printf("\n\n Speed of the driven shaft O, NO = %d
    rpm, clockwise.\n\n",-NO)
40 printf(" Speed of the wheel P, NP = %d rpm,
    clockwise.\n\n",-NP)
41 printf(" Torque tending to rotate the fixed wheel R
    = %.2f N-m.\n\n",T)

```

---

**Scilab code Exa 13.24** To find torque and forces

```

1 //To find torque and forces
2 clc
3 //Given:
4 TA=34, TB=120, TC=150, TD=38, TE=50
5 PX=7.5*1000 //W
6 NX=500 //rpm
7 m=3.5 //mm
8 //Solution:
9 //Refer Fig. 13.31 and Table 13.27
10 //Output torque of shaft Y:

```

```

11 //Calculating the values of x and y
12 //From the fourth row of the table , x+y = 500, or y+
    x = 500          .....( i)
13 //Also , y-x*(TA/TC) = 0
                                                    .....(
    ii)
14 A=[1 1; 1 -TA/TC]
15 B=[500; 0]
16 V=A \ B
17 y=V(1) //rpm
18 x=V(2) //rpm
19 //Calculating the speed of output shaft Y
20 NY=y-x*(TA/TB)*(TD/TE) //rpm
21 //Calculating the speed of wheel E
22 NE=NY //rpm
23 //Calculating the input power assuming 100 per cent
    efficiency
24 PY=PX //W
25 //Calculating the output torque of shaft Y
26 Ty=PY*60/(2*%pi*NY*1000) //Output torque on shaft Y,
    kN-m
27 //Tangential force between wheels D and E:
28 //Calculating the pitch circle radius of wheel E
29 rE=m*TE/(2*1000) //m
30 //Calculating the tangential force between wheels D
    and E
31 FtDE=Ty/rE //Tangential force between wheels D and E
    , kN
32 //Tangential force between wheels B and C:
33 //Calculating the input torque on shaft X
34 Tx=PX*60/(2*%pi*NX) //Input torque on shaft X, N-m
35 //Calculating the fixing torque on the fixed wheel C
36 Tf=Ty-Tx/1000 //Fixing torque on the fixed wheel C,
    kN-m
37 //Calculating the pitch circle radius of wheel C
38 rC=m*TC/(2*1000) //m
39 //Calculating the tangential forces between wheels B
    and C

```



```
40 FtBC=Tf/rC //kN
41 //Results:
42 printf("\n\n Output torque of shaft Y = %.3f kN-m.\n
      \n",Ty)
43 printf(" Tangential force between wheels D and E = %
      .1f kN.\n\n",FtDE)
44 printf(" Tangential force between wheels B and C =
      %d kN.\n\n",FtBC)
```

---

## Chapter 14

# Gyroscopic Couple and Precessional Motion

Scilab code Exa 14.1 To find speed of precession

```
1 //To find speed of precession
2 clc
3 //Given:
4 d=300/1000, r=d/2, l=600/1000 //m
5 m=5 //kg
6 N=300 //rpm
7 //Solution:
8 //Calculating the angular speed of the disc
9 omega=2*%pi*N/60 //rad/s
10 //Calculating the mass moment of inertia of the disc
    , about an axis through its centre of gravity and
    perpendicular to the plane of the disc
11 I=m*r^2/2 //kg-m^2
12 //Calculating the couple due to mass of disc
13 C=m*9.81*l //N-m
14 //Calculating the speed of precession
15 omegaP=C/(I*omega) //rad/s
16 //Results:
17 printf("\\n\\n Speed of precession , omegaP = %.1f rad/
```

```
s.\n\n", omegaP)
```

---

**Scilab code Exa 14.2** To find the resultant reaction

```
1 //To find the resultant reaction
2 clc
3 //Given:
4 d=150/1000, r=d/2, x=100/1000 //m
5 m=5 //kg
6 N=1000, NP=60 //rpm
7 //Solution:
8 //Calculating the angular speed of the disc
9 omega=2*%pi*N/60 //rad/s
10 //Calculating the speed of precession of the axle
11 omegaP=2*%pi*NP/60 //rad/s
12 //Calculating the mass moment of inertia of the disc
    , about an axis through its centre of gravity and
    perpendicular to the plane of disc
13 I=m*r^2/2 //kg-m^2
14 //Calculating the gyroscopic couple acting on the
    disc
15 C=I*omega*omegaP //N-m
16 //Calculating the force at each bearing due to the
    gyroscopic couple
17 F=C/x //N
18 //Calculating the reactions at the bearings A and B
19 RA=m/2*9.81 //N
20 RB=RA //N
21 //Resultant reaction at each bearing:
22 //Calculating the resultant reaction at the bearing
    A
23 RA1=F+RA //N
24 //Calculating the resultant reaction at the bearing
    B
25 RB1=F-RB //N
```

```

26 //Results :
27 printf("\n\n Resultant reaction at the bearing A,
      RA1 = %.1f N, upwards.\n\n",RA1)
28 printf(" Resultant reaction at the bearing B, RB1 =
      %.1f N, downwards.\n\n", RB1)

```

---

**Scilab code Exa 14.3** To find gyroscopic couple

```

1 //To find gyroscopic couple
2 clc
3 //Given:
4 R=50, k=0.3 //m
5 v=200*1000/3600 //m/s
6 m=400 //kg
7 N=2400 //rpm
8 //Solution:
9 //Calculating the angular speed of the engine
10 omega=2*%pi*N/60 //rad/s
11 //Calculating the mass moment of inertia of the
      engine and the propeller
12 I=m*k^2 //kg-m^2
13 //Calculating the angular velocity of precession
14 omegaP=v/R //rad/s
15 //Calculating the gyroscopic couple acting on the
      aircraft
16 C=I*omega*omegaP/1000 //kN-m
17 //Results:
18 printf("\n\n Gyroscopic couple acting on the
      aircraft , C = %.3f kN-m.\n\n",C)
19 printf(" The effect of the gyroscopic couple is to
      lift the nose upwards and tail downwards.\n\n")

```

---

**Scilab code Exa 14.4** To find gyroscopic couple

```

1 //To find gyroscopic couple
2 clc
3 //Given:
4 m=8*1000 //kg
5 k=0.6 , R=75 //m
6 N=1800 //rpm
7 v=100*1000/3600 //m/s
8 //Solution:
9 //Calculating the angular speed of the rotor
10 omega=2*%pi*N/60 //rad/s
11 //Calculating the mass moment of inertia of the
    rotor
12 I=m*k^2 //kg-m^2
13 //Calculating the angular velocity of precession
14 omegaP=v/R //rad/s
15 //Calculating the gyroscopic couple
16 C=I*omega*omegaP/1000 //kN-m
17 //Results:
18 printf("\\n\\n Gyroscopic couple , C = %.3f kN-m.\\n\\n",
    C)

```

---

**Scilab code Exa 14.5** To find gyroscopic couple and direction

```

1 //To find gyroscopic couple and direction
2 clc
3 //Given:
4 N=1500 //rpm
5 m=750 //kg
6 omegaP=1 //rad/s
7 k=250/1000 //m
8 //Solution:
9 //Calculating the angular speed of the rotor
10 omega=2*%pi*N/60 //rad/s
11 //Calculating the mass moment of inertia of the
    rotor

```

```

12 I=m*k^2 //kg-m^2
13 //Calculating the gyroscopic couple transmitted to
    the hull
14 C=I*omega*omegaP/1000 //kN-m
15 //Results:
16 printf("\n\n Gyroscopic couple transmitted to the
    hull , C = %.3f kN-m.\n\n",C)
17 printf(" When the pitching is upward, the relative
    gyroscopic couple acts in the clockwise direction
    .\n\n")

```

---

**Scilab code Exa 14.6** To find gyroscopic couple and effect

```

1 //To find gyroscopic couple and effect
2 clc
3 //Given:
4 m=3500 //kg
5 k=0.45 //m
6 N=3000 //rpm
7 //Solution:
8 //Calculating the angular speed of the rotor
9 omega=2*%pi*N/60 //rad/s
10 //When the ship is steering to the left:
11 R=100 //m
12 v=36*1000/3600 //m/s
13 //Calculating the mass moment of inertia of the
    rotor
14 I=m*k^2 //kg-m^2
15 //Calculating the angular velocity of precession
16 omegaP=v/R //rad/s
17 //Calculating the gyroscopic couple
18 C=I*omega*omegaP/1000 //kN-m
19 //Results:
20 printf("\n\n Gyroscopic couple when the ship is
    steering to the left , C = %.2f kN-m.\n\n",C)

```

```

21 printf(" When the rotor rotates clockwise and the
    ship takes a left turn, the effect of the
    reactive gyroscopic couple is to raise the bow
    and lower the stern.\n\n")
22 //When the ship is pitching with the bow falling:
23 tp=40 //s
24 //Calculating the amplitude of swing
25 phi=12/2*%pi/180 //rad
26 //Calculating the angular velocity of the simple
    harmonic motion
27 omega1=2*%pi/tp //rad/s
28 //Calculating the maximum angular velocity of
    precession
29 omegaP=phi*omega1 //rad/s
30 //Calculating the gyroscopic couple
31 C=I*omega*omegaP/1000 //kN-m
32 //Results:
33 printf(" Gyroscopic couple when the ship is pitching
    with the bow falling , C = %.3f kN-m.\n\n",C)
34 printf(" When the bow is falling , the effect of the
    reactive gyroscopic couple is to move the ship
    towards port side.\n\n")

```

---

**Scilab code Exa 14.7** To find gyroscopic couple and acceleration

```

1 //To find gyroscopic couple and acceleration
2 clc
3 //Given:
4 m=20*1000 //kg
5 k=0.6 //m
6 N=2000 //rpm
7 phi=6*%pi/180 //rad
8 tp=30 //s
9 //Solution:
10 //Calculating the angular speed of the rotor

```

```

11 omega=2*%pi*N/60 //rad/s
12 //Maximum gyroscopic couple:
13 //Calculating the mass moment of inertia of the
    rotor
14 I=m*k^2 //kg-m^2
15 //Calculating the angular velocity of the simple
    harmonic motion
16 omega1=2*%pi/tp //rad/s
17 //Calculating the maximum angular velocity of
    precession
18 omegaPmax=phi*omega1 //rad/s
19 //Calculating the maximum gyroscopic couple
20 Cmax=I*omega*omegaPmax/1000 //kN-m
21 //Calculating the maximum angular acceleration
    during pitching
22 alphamax=phi*omega1^2 //Maximum angular acceleration
    during pitching, rad/s^2
23 //Results:
24 printf("\n\n Maximum gyroscopic couple, Cmax = %.3f
    kN-m.\n\n",Cmax)
25 printf(" Maximum angular acceleration during
    pitching = %.4f rad/s^2.\n\n",alphamax)
26 printf(" When the rotation of the rotor is clockwise
    when looking from the left and when the bow is
    rising, then the reactive gyroscopic couple acts
    in the direction which tends to turn the bow
    towards right.\n\n")

```

---

**Scilab code Exa 14.8** To find the gyroscopic effects

```

1 //To find the gyroscopic effects
2 clc
3 //Given:
4 m=5*1000 //kg
5 N=1000 //rpm

```



```

6 k=0.5 //m
7 //Solution:
8 //Calculating the angular speed of the rotor
9 omega=2*%pi*N/60 //rad/s
10 //When the ship steers to the left:
11 v=30*1000/3600 //m/s
12 R=60 //m
13 //Calculating the angular velocity of precession
14 omegaP=v/R //rad/s
15 //Calculating the mass moment of inertia of the
    rotor
16 I=m*k^2 //kg-m^2
17 //Calculating the gyroscopic couple
18 C=I*omega*omegaP/1000 //kN-m
19 //Results:
20 printf("\n\n When the rotor rotates in a clockwise
    direction when viewed from from the stern and the
    ship steers to the left , the effect of reactive
    gyroscopic couple is to raise the bow and lower
    the stern.\n\n")
21 //When the ship pitches with the bow descending:
22 phi=6*%pi/180 //rad
23 tp=20 //s
24 //Calculating the angular velocity of simple
    harmonic motion
25 omega1=2*%pi/tp //rad/s
26 //Calculating the maximum velocity of precession
27 omegaPmax=phi*omega1 //rad/s
28 //Calculating the maximum gyroscopic couple
29 Cmax=I*omega*omegaPmax //N-m
30 //Results:
31 printf(" Since the ship is pitching with the low bow
    descending , therefore the effect of this maximum
    gyroscopic couple is to turn the ship towards
    port side.\n\n")
32 //When the ship rolls:
33 omegaP=0.03 //rad/s
34 //Calculating the gyroscopic couple

```

```

35 C=I*omega*omegaP //N-m
36 //Results:
37 printf(" In case of rolling of a ship , the axis of
    precession is always parallel to the axis of spin
    for all positions , therefore there is no effect
    of gyroscopic couple.\n\n")
38 //Calculating the maximum angular acceleration
    during pitching
39 alphamax=phi*omega1^2 //rad/s^2
40 //Results:
41 printf(" Maximum angular acceleration during
    pitching , alphamax = %.2f rad/s^2.\n\n",alphamax)

```

---

**Scilab code Exa 14.9** To find maximum acceleration during pitching

```

1 //To find maximum acceleration during pitching
2 clc
3 //Given:
4 m=2000 //kg
5 N=3000 //rpm
6 k=0.5 , R=100 //m
7 v=16.1*1855/3600 //m/s
8 //Solution:
9 //Calculating the angular speed of the rotor
10 omega=2*pi*N/60 //rad/s
11 //Gyroscopic couple:
12 //Calculating the mass moment of inertia of the
    rotor
13 I=m*k^2 //kg-m^2
14 //Calculating the angular velocity of precession
15 omegaP=v/R //rad/s
16 //Calculating the gyroscopic couple
17 C=I*omega*omegaP/1000 //kN-m
18 //Torque during pitching:
19 tp=50 //s

```

```

20 phi=12/2*%pi/180 //rad
21 //Calculating the angular velocity of simple
    harmonic motion
22 omega1=2*%pi/tp //rad/s
23 //Calculating the maximum angular velocity of
    precession
24 omegaPmax=phi*omega1 //rad/s
25 //Calculating the maximum gyroscopic couple during
    pitching
26 Cmax=I*omega*omegaPmax //N-m
27 //Calculating the maximum acceleration during
    pitching
28 alphamax=phi*omega1^2 //rad/s^2
29 //Results:
30 printf("\n\n When the rotor rotates clockwise when
    looking from a stern the ship steers to the right
    , the effect of the reactive gyroscopic couple is
    to raise the stern and lower the bow.\n\n")
31 printf(" Torque during pitching , Cmax = %d N-m.\n\n"
    ,Cmax)
32 printf(" Maximum acceleration during pitching ,
    alphamax = %.5f rad/s^2.\n\n",alphamax)

```

---

**Scilab code Exa 14.10** To find centrifugal and gyroscopic effects

```

1 //To find centrifugal and gyroscopic effects
2 clc
3 //Given:
4 m=2500 //kg
5 x=1.5, R=30, dW=0.75, rW=dW/2, h=0.9 //m
6 v=24*1000/3600 //m/s
7 G=5
8 IW=18, IE=12 //kg-m^2
9 //Solution:
10 //Calculating the road reaction on each wheel

```

```

11 r=m*9.81/4 //Road reaction on each wheel, N
12 //Calculating the angular velocity o the wheels
13 omegaW=v/rW //rad/s
14 //Calculating the angular velocity of precession
15 omegaP=v/R //rad/s
16 //Calculating the gyroscopic couple due to one pair
    of wheels and axle
17 CW=round(2*IW*omegaW*omegaP) //N-m
18 //Calculating the gyroscopic couple due to the
    rotating parts of the motor and gears
19 CE=round(2*IE*G*omegaW*omegaP) //N-m
20 //Calculating the net gyroscopic couple
21 C=CW-CE //N-m
22 //Calculating the reaction due to gyroscopic couple
    at each of the outer or inner wheels
23 P=2*(-C)/(2*x) //N
24 //Calculating the centrifugal force
25 FC=m*v^2/R //N
26 //Calculating the overturning couple
27 CO=FC*h //N-m
28 //Calculating the reaction due to overturning couple
    at each of the outer and inner wheels
29 Q=2*CO/(2*x) //N
30 //Calculating the vertical force exerted on each
    outer wheel
31 PO=m*9.81/4-P/2+Q/2 //N
32 //Calculating the vertical force exerted on each
    inner wheel
33 PI=m*9.81/4+P/2-Q/2 //N
34 //Results:
35 printf("\n\n Vertical force exerted on each outer
    wheel, PO = %.2 f N.\n\n",PO)
36 printf(" Vertical force exerted on each inner wheel,
    PI = %.2 f N.\n\n",PI)

```

---

Scilab code Exa 14.12 To find load on each wheel

```
1 //To find load on each wheel
2 clc
3 //Given:
4 m=2000, mE=75 //kg
5 b=2.5, x=1.5, h=500/1000, L=1, dW=0.8, rW=dW/2, kE
   =100/1000, R=60 //m
6 IW=0.8 //kg-m2
7 G=4
8 v=60*1000/3600 //m/s
9 //Solution:
10 //Refer Fig. 14.12
11 //Calculating the weight on the rear wheels
12 W2=(m*9.81*1)/b //N
13 //Calculating the weight on the front wheels
14 W1=m*9.81-W2 //N
15 //Calculating the weight on each of the front wheels
16 Wf=W1/2 //Weight on each of the front wheels, N
17 //Calculating the weight on each of the rear wheels
18 Wr=W2/2 //Weight on each of the rear wheels, N
19 //Calculating the angular velocity of wheels
20 omegaW=v/rW //rad/s
21 //Calculating the angular velocity of precession
22 omegaP=v/R //rad/s
23 //Calculating the gyroscopic couple due to four
   wheels
24 CW=4*IW*omegaW*omegaP //N-m
25 //Calculating the magnitude of reaction due to
   gyroscopic couple due to four wheels at each of
   the inner or outer wheel
26 P=2*(CW/(2*x)) //N
27 //Calculating the mass moment of inertia of rotating
   parts of the engine
28 IE=mE*(kE)^2 //kg-m2
29 //Calculating the gyroscopic couple due to rotating
   parts of the engine
30 CE=IE*(kE)^2*G*omegaW*omegaP //N-m
```

```

31 //Calculating the magnitude of reaction due to
    gyroscopic couple due to rotating parts of the
    engine at each of the inner or outer wheel
32 F=2*(CE/(2*b)) //N
33 //Calculating the centrifugal force
34 FC=m*v^2/R //N
35 //Calculating the centrifugal couple tending to
    overturn the car
36 CO=FC*h //N-m
37 //Calculating the magnitude of reaction due to
    overturning couple at each of the inner or outer
    wheel
38 Q=2*(CO/(2*x)) //N
39 //Calculating the load on front wheel 1
40 Fw1=W1/2-P/2-F/2-Q/2 //Load on front wheel 1, N
41 //Calculating the load on front wheel 2
42 Fw2=W1/2+P/2-F/2+Q/2 //Load on front wheel 2, N
43 //Calculating the load on rear wheel 3
44 Rw3=W2/2-P/2+F/2-Q/2 //Load on rear wheel 3, N
45 //Calculating the load on rear wheel 4
46 Rw4=W2/2+P/2+F/2+Q/2 //Load on rear wheel 4, N
47 //Results:
48 printf("\n\n Load on front wheel 1 = %.2 f N.\n\n",
    Fw1)
49 printf(" Load on front wheel 2 = %.2 f N.\n\n",Fw2)
50 printf(" Load on rear wheel 3 = %.2 f N.\n\n",Rw3)
51 printf(" Load on rear wheel 4 = %.2 f N.\n\n",Rw4)

```

---

**Scilab code Exa 14.13** To find pressure on each rail

```

1 //To find pressure on each rail
2 clc
3 //Given:
4 m=2000, mI=200 //kg
5 x=1.6, R=30, dW=0.7, rW=dW/2, k=0.3, h=1 //m

```

```

6 v=54*1000/3600 //m/s
7 theta=8 //degrees
8 //Solution:
9 //Refer Fig. 14.13
10 //Calculating the reactions at the wheels:
11 //Taking moments about B
12 RA=(m*9.81*cosd(theta)+m*v^2/R*sind(theta))*1/2+(m
    *9.81*sind(theta)-m*v^2/R*cosd(theta))*h/x //N
13 //Resolving the forces perpendicular to the track
14 RB=(m*9.81*cosd(theta)+m*v^2/R*sind(theta))-RA //N
15 //Calculating the angular velocity of wheels
16 omegaW=v/rW //rad/s
17 //Calculating the angular velocity of precession
18 omegaP=v/R //rad/s
19 //Calculating the gyroscopic couple
20 C=mI*k^2*omegaW*cosd(theta)*omegaP //N-m
21 //Calculating the force at each pair of wheels due
    to the gyroscopic couple
22 P=C/x //N
23 //Calculating the pressure on the inner rail
24 PI=RA-P //N
25 //Calculating the pressure on the outer rail
26 PO=RB+P //N
27 //Results:
28 printf("\n\n Pressure on the inner rail , PI = %.2 f N
    .\n\n",PI)
29 printf(" Pressure on the outer rail , PO = %.2 f N.\n\n\
    n",PO)

```

---

**Scilab code Exa 14.14** To find gyroscopic couple and reaction

```

1 //To find gyroscopic couple and reaction
2 clc
3 //Given:
4 I=180 //kg-m^2

```

```

5 D=1.8, R=D/2, x=1.5 //m
6 v=95*1000/3600 //m/s
7 t=0.1 //s
8 //Solution:
9 //Gyroscopic couple set up:
10 //Calculating the angular velocity of the locomotive
11 omega=v/R //rad/s
12 //Calculating the amplitude
13 A=1/2*6 //mm
14 //Calculating the maximum velocity while falling
15 vmax=2*pi/t*A/1000 //m/s
16 //Calculating the maximum angular velocity of tilt
    of the axle or angular velocity of precession
17 omegaPmax=vmax/x //rad/s
18 //Calculating the gyroscopic couple set up
19 C=I*omega*omegaPmax //N-m
20 //Calculating the reaction between the wheel and
    rail due to the gyroscopic couple
21 P==C/x //N
22 //Results:
23 printf("\n\n Gyroscopic couple set up, C = %.1f N-m
    .\n\n",C)
24 printf(" Reaction between the wheel and rail due to
    the gyroscopic couple , P = %d N.\n\n",P)

```

---

**Scilab code Exa 14.15** To find angle of inclination

```

1 //To find angle of inclination
2 clc
3 //Given:
4 m=250 //kg
5 IE=0.3, IW=1 //kg-m^2
6 G=5
7 h=0.6, rW=300/1000, R=50 //m
8 v=90*1000/3600 //m/s

```



```

9 //Solution:
10 //Calculating the angle of inclination with respect
    to the vertical of a two wheeler
11 //Equating total overturning couple to balancing
    couple ,
12 theta=atand(((1/(m*9.81*h))*((v^2/(R*rW)*(2*IW+G*IE))
    +(m*v^2/R*h)))) //degrees
13 //Results:
14 printf("\n\n Angle of inclination with respect to
    the vertical of a two wheeler, theta = %.2f
    degrees.\n\n",theta)

```

---

**Scilab code Exa 14.16** To find inclination of gyrowheel

```

1 //To find inclination of gyrowheel
2 clc
3 //Given:
4 m1=0.5, m2=0.3 //kg
5 k=20/1000, OG=10/1000, h=OG, R=50 //m
6 N=3000 //rpm
7 v=15 //m/s
8 //Solution:
9 //Refer Fig. 14.15 and Fig. 14.16
10 //Calculating the angular speed of the wheel
11 omega=2*pi*N/60 //rad/s
12 //Calculating the mass moment of inertia of the
    gyrowheel
13 I=m1*k^2 //kg-m^2
14 //Calculating the angular velocity of precession
15 omegaP=v/R //rad/s
16 //When the vehicle moves in the direction of arrow X
    taking a left turn along the curve:
17 //Calculating the angle of inclination of the
    gyrowheel from the vertical
18 //Equating the overturning couple to the balancing

```

```

    couple for equilibrium condition ,
19 theta1=atand((1/(m2*9.81*h))*(I*omega*omegaP-m2*v^2/
    R*h)) //degrees
20 //When the vehicle reverses at the same speed in the
    direction of arrow Y along the same path:
21 //Calculating the angle of inclination of the
    gyrowheel from the vertical
22 //Equating the overturning couple to the balancing
    couple for equilibrium condition ,
23 theta2=atand((1/(m2*9.81*h))*(I*omega*omegaP+m2*v^2/
    R*h)) //degrees
24 //Results:
25 printf("\n\n Angle of inclination of the gyrowheel
    from the vertical when the vehicle moves in the
    direction of arrow X taking a left turn along the
    curve, theta = %.2f degrees.\n\n",theta1)
26 printf(" Angle of inclination of the gyrowheel from
    the vertical when the vehicle reverses at the
    same speed in the direction of arrow Y along the
    same path, theta = %.2f degrees.\n\n",theta2)

```

---

**Scilab code Exa 14.17** To find the gyroscopic couple

```

1 //To find the gyroscopic couple
2 clc
3 //Given:
4 d=0.6, r=d/2 //m
5 m=30 //kg
6 theta=1 //degree
7 N=1200 //rpm
8 //Solution:
9 //Calculating the angular speed of the shaft
10 omega=2*pi*N/60 //rad/s
11 //Calculating the gyroscopic couple acting on the
    bearings

```

```
12 C=round(m/8*omega^2*r^2*sind(2*theta)) //N-m
13 //Results:
14 printf("\n\n Gyroscopic couple acting on the
    bearings , C = %d N-m.\n\n",C)
```

---

# Chapter 15

## Inertia Forces in Reciprocating Parts

Scilab code Exa 15.1 To find linear and angular velocity and acceleration

```
1 //To find linear and angular velocity and
   acceleration
2 clc
3 //Given:
4 OC=200/1000, PC=700/1000 //m
5 omega=120 //rad/s
6 //Solution:
7 //Refer Fig. 15.5
8 OM=127/1000, CM=173/1000, QN=93/1000, NO=200/1000 //
   m
9 //Velocity and acceleration of the piston:
10 //Calculating the velocity of the piston P
11 vP=omega*OM //m/s
12 //Calculating the acceleration of the piston P
13 aP=omega^2*NO //m/s^2
14 //Velocity and acceleration of the mid-point of the
   connecting rod:
15 //By measurement,
16 OD1=140/1000, OD2=193/1000 //m
```

```

17 //Calculating the velocity of D
18 vD=omega*OD1 //m/s
19 //Calculating the acceleration of D
20 aD=omega^2*OD2 //m/s^2
21 //Angular velocity and angular acceleration of the
    connecting rod:
22 //Calculating the velocity of the connecting rod PC
23 vPC=omega*CM //m/s
24 //Calculating the angular velocity of the connecting
    rod PC
25 omegaPC=vPC/PC //rad/s
26 //Calculating the tangential component of the
    acceleration of P with respect to C
27 atPC=omega^2*QN //m/s^2
28 //Calculating the angular acceleration of the
    connecting rod PC
29 alphaPC=atPC/PC //ra/s^2
30 //Results:
31 printf("\n\n Velocity of the piston P, vP = %.2 f m/s
    .\n\n",vP)
32 printf(" Acceleration of the piston P, aP = %d m/s
    ^2.\n\n", aP)
33 printf(" Velocity of D, vD = %.1 f m/s.\n\n",vD)
34 printf(" Acceleration of D, aD = %.1 f m/s^2.\n\n",aD
    )
35 printf(" Angular velocity of the connecting rod PC,
    omegaPC = %.2 f rad/s.\n\n",omegaPC)
36 printf(" Angular acceleration of the connecting rod
    PC, alphaPC = %.2 f rad/s^2.\n\n",alphaPC)

```

---

**Scilab code Exa 15.2** To find linear and angular velocity and acceleration

```

1 //To find linear and angular velocity and
    acceleration
2 clc

```

```

3 //Given:
4 OC=150/1000 , PC=600/1000 , CD=150/1000 //m
5 N=450 //rpm
6 //Solution:
7 //Refer Fig. 15.6
8 //Calculating the angular speed of the crank
9 omega=2*%pi*N/60 //rad/s
10 //By measurement,
11 OM=145/1000 , CM=78/1000 , QN=130/1000 , NO=56/1000 //m
12 //Velocity and acceleration of alider:
13 //Calculating the velocity of the slider P
14 vP=omega*OM //m/s
15 //Calculating the acceleration of the slider P
16 aP=omega^2*NO //m/s^2
17 //Velocity and acceleration of point D on the
    connecting rod:
18 //Calculating the length od CD1
19 CD1=CD/PC*CM //m
20 //By measurement,
21 OD1=145/1000 , OD2=120/1000 //m
22 //Calculating the velocity of point D
23 vD=omega*OD1 //m/s
24 //Calculating the acceleration of point D
25 aD=omega^2*OD2 //m/s^2
26 //Angular velocity and angular acceleration of the
    connecting rod:
27 //Calculating the velocity of the connecting rod PC
28 vPC=omega*CM //m/s
29 //Calculating the angular velocity of the connecting
    rod
30 omegaPC=vPC/PC //rad/s
31 //Calculating the tangential component of the
    acceleration of P with respect to C
32 atPC=omega^2*QN //m/s^2
33 //Calculating the angular acceleration of the
    connecting rod PC
34 alphaPC=atPC/PC //rad/s^2
35 //Results:

```

```

36 printf("\n\n Velocity of the slider P, vP = %.3 f m/s
    .\n\n",vP)
37 printf(" Acceleration of the slider P, aP = %.1 f m/s
    ^2.\n\n",aP)
38 printf(" Velocity of point D, vD = %.3 f m/s.\n\n",vD
    )
39 printf(" Acceleration of point D, aD = %.2 f m/s ^2.\n
    \n",aD)
40 printf(" Angular velocity of the connecting rod ,
    omegaPC = %.3 f rad/s.\n\n",omegaPC)
41 printf(" Angular acceleration of the connecting rod
    PC, alphaPC = %.2 f rad/s ^2.\n\n",alphaPC)

```

---

**Scilab code Exa 15.3** To find crank angle and velocity

```

1 //To find crank angle and velocity
2 clc
3 //Given:
4 r=300/1000, l=1 //m
5 N=200 //rpm
6 //Solution:
7 //Calculating the angular speed of the crank
8 omega=2*%pi*N/60 //rad/s
9 //Crank angle at which the maximum velocity occurs:
10 //Calculating the ratio of length of connecting rod
    to crank radius
11 n=l/r
12 //Velocity of the piston , vP = omega*r*(sind(theta)+
    sind(2*theta)/(2*n)) .....( i)
13 //For maximum velocity , d(vP)/d(theta) = 0
    .....( ii)
14 //Substituting (i) in (ii), we get , 2(cosd(theta))
    ^2+n*cosd(theta)-1 = 0
15 a=2, b=n, c=-1
16 costheta=(-b+sqrt(b^2-4*a*c))/(2*a)

```

```

17 //Calculating the crank angle from the inner dead
    centre at which the maximum velocity occurs
18 theta=round(acosd(costheta)) //degrees
19 //Calculating the maximum velocity of the piston:
20 vPmax=omega*r*(sind(theta)+sind(2*theta)/(2*n)) //m/
    s
21 //Results:
22 printf("\n\n Crank angle from the inner dead centre
    at which the maximum velocity occurs , theta = %d
    degrees.\n\n",theta)
23 printf(" Maximum velocity of the piston , vP(max) = %
    .2 f m/s.\n\n",vPmax)

```

---

**Scilab code Exa 15.4** To find velocity and acceleration

```

1 //To find velocity and acceleration
2 clc
3 //Given:
4 r=0.3, l=1.5 //m
5 N=180 //rpm
6 theta=40 //degrees
7 //Solution:
8 //Calculating the angular speed of the piston
9 omega=2*pi*N/60 //rad/s
10 //Velocity of the piston:
11 //Calculating the ratio of lengths of the connecting
    rod and crank
12 n=l/r
13 //Calculating the velocity of the piston
14 vP=omega*r*(sind(theta)+sind(2*theta)/(2*n)) //m/s
15 //Calculating the acceleration of the piston
16 aP=omega^2*r*(cosd(theta)+cosd(2*theta)/n) //m/s^2
17 //Position of the crank for zero acceleration of the
    piston:
18 ap1=0

```



```

19 //Calculating the position of the crank from the
    inner dead centre for zero acceleration of the
    piston
20 //We have,  $a_p = \omega^2 r (\cos(\theta) + \cos(2\theta))$ , or  $2(\cos(\theta))^2 + n \cos(\theta) - 1 = 0$ 
21 a=2, b=n, c=-1
22 costheta1=(-b+sqrt(b^2-4*a*c))/(2*a)
23 //Calculating the crank angle from the inner dead
    centre for zero acceleration of the piston
24 theta1=acosd(costheta1) //degrees
25 //Results:
26 printf("\n\n Velocity of the piston , vP = %.2f m/s.\n\n", vP)
27 printf(" Acceleration of the piston , aP = %.2f m/s\n\n", aP)
28 printf(" Position of the crank for zero acceleration
    of the piston , theta1 = %.2f degrees or %.2f
    degrees.\n\n", theta1, 360-theta1)

```

---

**Scilab code Exa 15.5** To find linear and angular velocity and acceleration

```

1 //To find linear and angular velocity and
    acceleration
2 clc
3 //Given:
4 r=150/1000, l=600/1000 //m
5 theta=60 //degrees
6 N=450 //rpm
7 //Solution:
8 //Calculating the angular speed of the crank
9 omega=2*pi*N/60 //rad/s
10 //Velocity and acceleration of the slider:
11 //Calculating the ratio of length of connecting rod
    and crank

```

```

12 n=1/r
13 //Calculating the velocity of the slider
14 vP=omega*r*(sind(theta)+sind(2*theta)/(2*n)) //m/s
15 //Calculating the acceleration of the slider
16 aP=omega^2*r*(cosd(theta)+cosd(2*theta)/n) //m/s^2
17 //Angular velocity and angular acceleration of the
    connecting rod:
18 //Calculating the angular velocity of the connecting
    rod
19 omegaPC=omega*cosd(theta)/n //rad/s
20 //Calculating the angular acceleration of the
    connecting rod
21 alphaPC=round(omega^2*sind(theta)/n) //rad/s^2
22 //Results:
23 printf("\n\n Velocity of the slider , vP = %.1f m/s.\n\n",vP)
24 printf(" Acceleration of the slider , aP = %.2f m/s\n\n",aP)
25 printf(" Angular velocity of the connecting rod ,\n\n",omegaPC)
26 printf(" Angular acceleration of the connecting rod ,\n\n",alphaPC)

```

---

**Scilab code Exa 15.6** To find inertia force

```

1 //To find inertia force
2 clc
3 //Given:
4 D=175/1000, L=200/1000, r=L/2, l=400/1000 //m
5 N=500 //rpm
6 mR=180 //kg
7 //Solution:
8 //Calculating the angular speed of the crank
9 omega=2*pi*N/60 //rad/s
10 //Analytical method:

```

```

11 //Calculating the ratio of lengths of connecting rod
    and crank
12 n=l/r
13 //Calculating the inertia force
14 FI=mR*omega^2*r*(cosd(theta)+cosd(2*theta)/n)/1000
    //kN
15 //Results:
16 printf("\n\n Inertia force , FI = %.2f kN.\n\n",FI)

```

---

**Scilab code Exa 15.7** To find pressure thrust force and moment

```

1 //To find pressure , thrust , force and moment
2 clc
3 //Given:
4 r=300/1000 , l=1.2 , D=0.5 //m
5 mR=250 //kg
6 theta=60 //degrees
7 dp=0.35 //p1-p2 , N/mm^2
8 N=250 //rpm
9 //Solution:
10 //Calculating the angular speed of the crank
11 omega=2*%pi*N/60 //rad/s
12 //Calculating the net load on the piston
13 FL=(dp)*%pi/4*(D*1000)^2 //N
14 //Calculating the ratio of length of connecting rod
    and crank
15 n=l/r
16 //Calculating the accelerating or inertia force on
    reciprocating parts
17 FI=mR*omega^2*r*(cosd(theta)+cosd(2*theta)/n) //N
18 //Calculating the piston effort
19 FP=(FL-FI)/1000 //kN
20 //Pressure on slide bars:
21 //Calculating the angle of inclination of the
    connecting rod to the line of stroke

```

```

22 phi=asind(sind(theta)/n) //degrees
23 //Calculating the pressure on the slide bars
24 FN=FP*tand(phi) //kN
25 //Calculating the thrust in the connecting rod
26 FQ=FP/cosd(phi) //kN
27 //Calculating the tangential force on the crank pin
28 FT=FQ*sind(theta+phi) //kN
29 //Calculating the turning moment on the crank shaft
30 T=FT*r //kN-m
31 //Results:
32 printf("\n\n Pressure on the slide bars , FN = %.2f
      kN.\n\n",FN)
33 printf(" Thrust in the connecting rod , FQ = %.2f kN
      .\n\n",FQ)
34 printf(" Tangential force on the crank-pin , FT = %.2
      f kN.\n\n",FT)
35 printf(" Turning moment on the crank shaft , T = %.3f
      kN-m.\n\n",T)

```

---

**Scilab code Exa 15.8** To find turning moment

```

1 //To find turning moment
2 clc
3 //Given:
4 D=300/1000 , L=450/1000 , r=L/2 , d=50/1000 , l=1.2 //m
5 N=200 //rpm
6 mR=225 //kg
7 theta=125 //degrees
8 p1=30*1000 , p2=1.5*1000 //N/m^2
9 //Solution:
10 //Calculating the angular speed of the crank
11 omega=2*%pi*N/60 //rad/s
12 //Calculating the area of the piston
13 A1=%pi/4*D^2 //m^2
14 //Calculating the area of the piston rod

```

```

15 a=%pi/4*d^2 //m^2
16 //Calculating the force on the piston due to steam
    pressure
17 FL=round(p1*A1-p2*(A1-a)) //N
18 //Calculating the ratio of lengths of connecting rod
    and crank
19 n=l/r
20 //Calculating the inertia force on the reciprocating
    parts
21 FI=mR*omega^2*r*(cosd(theta)+cosd(2*theta)/n) //N
22 //Calculating the net force on the piston or piston
    effort
23 FP=FL-FI+mR*9.81 //N
24 //Calculating the angle of inclination of the
    connecting rod to the line of stroke
25 phi=asind(sind(theta)/n) //degrees
26 //Calculating the effective turning moment on the
    crank shaft
27 T=FP*sind(theta+phi)/cosd(phi)*r //N-m
28 //Results:
29 printf("\n\n Effective turning moment of the crank
    shaft , T = %.1f N-m.\n\n",T)

```

---

**Scilab code Exa 15.9** To find load thrust reaction and speed

```

1 //To find load , thrust , reaction and speed
2 clc
3 //Given:
4 N=1800 //rpm
5 r=50/1000, l=200/1000, D=80/1000, x=10/1000 //m
6 mR=1 //kg
7 p=0.7 //N/mm^2
8 //Solution:
9 //Calculating the angular speed of the crank
10 omega=2*%pi*N/60 //rad/s

```

```

11 //Net load on the gudgeon pin:
12 //Calculating the load on the piston
13 FL=round(%pi/4*(D*1000)^2*p) //N
14 //Refer Fig. 15.10
15 //By measurement,
16 theta=33 //degrees
17 //Calculating the ratio of lengths of connecting rod
    and crank
18 n=1/r
19 //Calculating the inertia force on the reciprocating
    parts
20 FI=mR*omega^2*r*(cosd(theta)+cosd(2*theta)/n) //N
21 //Calculating the net load on the gudgeon pin
22 FP=FL-FI //N
23 //Thrust in the connecting rod:
24 //Calculating the angle of inclination of the
    connecting rod to the line of stroke
25 phi=asind(sind(theta)/n) //degrees
26 //Calculating the thrust in the connecting rod
27 FQ=FP/cosd(phi) //N
28 //Calculating the reaction between the piston and
    cylinder
29 FN=FP*tand(phi) //N
30 //Engine speed at which the above values will become
    zero:
31 //Calculating the speed at which FI=FL
32 omega1=sqrt((%pi/4*(D*1000)^2*p)/(mR*r*(cosd(theta)+
    cosd(2*theta)/n))) //rad/s
33 //Calculating the corresponding speed in rpm
34 N1=omega1*60/(2*pi) //rpm
35 //Results:
36 printf("\n\n Net load on the gudgeon pin , FP = %d N
    .\n\n",FP)
37 printf(" Thrust in the connecting rod , FQ = %.1f N.\n
    \n",FQ)
38 printf(" Reaction between the piston and cylinder ,
    FN = %d N.\n\n",FN)
39 printf(" Engine speed at which the above values will

```

become zero ,  $N1 = \%d \text{ rpm}.\backslash n\backslash n$  ,N1)

---

**Scilab code Exa 15.10** To find reaction thrust and turning moment

```
1 //To find reaction , thrust and turning moment
2 clc
3 //Given:
4 aP=36 //m/s^2
5 theta=30 //degrees
6 p=0.5 //N/mm^2
7 RF=600 //N
8 D=300/1000 , r=300/1000 //m
9 mR=180 //kg
10 n=4.5
11 //Solution:
12 //Reaction on the guide bars:
13 //Calculating the load on the piston
14 FL=round(p*pi/4*(D*1000)^2) //N
15 //Calculating the inertia force due to reciprocating
    parts
16 FI=mR*aP //N
17 //Calculating the piston effort
18 FP=(FL-FI-RF)/1000 //kN
19 //Calculating the angle of inclination of the
    connecting rod to the line of stroke
20 phi=asind(sind(theta)/n) //degrees
21 //Calculating the reaction on the guide bars
22 FN=FP*tand(phi) //kN
23 //Calculating the thrust on the crank shaft bearing
24 FB=(FP*cosd(phi+theta))/cosd(phi) //kN
25 //Calculating the turning moment on the crank shaft
26 T=(FP*sind(theta+phi))/cosd(phi)*r //kN-m
27 //Results:
28 printf("\n\n Reaction on the guide bars , FN = %.2 f
    kN.\n\n" ,FN)
```

```

29 printf(" Thrust on the crank shaft bearing , FB = %.1
    f kN.\n\n",FB)
30 printf(" Turning moment on the crank shaft , T = %.2 f
    kN-m.\n\n",T)

```

---

**Scilab code Exa 15.11** To find force load thrust and speed

```

1 //To find force , load , thrust and speed
2 clc
3 //Given:
4 D=100/1000 , L=120/1000 , r=L/2 , l=250/1000 //m
5 mR=1.1 //kg
6 N=2000 //rpm
7 theta=20 //degrees
8 p=700 //kN/m^2
9 //Solution:
10 //Calculating the angular speed of the crank
11 omega=2*%pi*N/60 //rad/s
12 //Net force on the piston:
13 //Calculating the force due to gas pressure
14 FL=p*%pi/4*D^2 //kN
15 //Calculating the ratio of lengths of the connecting
    rod and crank
16 n=l/r
17 //Calculating the inertia force on the piston
18 FI=round(mR*omega^2*r*(cosd(theta)+cosd(2*theta)/n))
    //N
19 //Calculating the net force on the piston
20 FP=(FL*1000)-FI+mR*9.81 //N
21 //Resultant force on the gudgeon pin:
22 //Calculating the angle of inclination of the
    connecting rod to the line of stroke
23 phi=asind(sind(theta)/n) //degrees
24 //Calculating the resultant load on the gudgeon pin
25 FQ=round(FP/cosd(phi)) //N

```



```

26 //Calculating the thrust on the cylinder walls
27 FN=FP*tand(phi) //N
28 //Speed, above which, the gudgeon pin load would be
    reversed in direction:
29 //Calculating the minimum speed for FP to be
    negative
30 omega1=sqrt((FL*1000+mR*9.81)/(mR*r*(cosd(theta)+
    cosd(2*theta)/n))) //rad/s
31 //Calculating the corresponding speed in rpm
32 N1=omega1*60/(2*pi) //rpm
33 //Results:
34 printf("\n\n Net force on the piston , FP = %.1f N.\n
    \n",FP)
35 printf(" Resultant load on the gudgeon pin , FQ = %d
    N.\n\n",FQ)
36 printf(" Thrust on the cylinder walls , FN = %.1f N.\n
    \n",FN)
37 printf(" Speed, above which, the gudgeon pin load
    would be reversed in direction , N1 > %d rpm.\n\n"
    ,N1)

```

---

**Scilab code Exa 15.12** To find turning moment thrust and acceleration

```

1 //To find turning moment, thrust and acceleration
2 clc
3 //Given:
4 N=120 //rpm
5 D=250/1000, L=400/1000, r=L/2, l=0.6, d=50/1000 //m
6 mR=60 //kg
7 theta=45 //degrees
8 p1=550*1000, p2=70*1000 //N/m^2
9 //Solution:
10 //Calculating the angular speed of the crank
11 omega=2*pi*N/60 //rad/s
12 //Turning moment on the crankshaft:

```

```

13 //Calculating the area of the piston on the cover
    end side
14 A1=%pi/4*D^2 //m^2
15 //Calculating the area of the piston rod
16 a=%pi/4*d^2 //m^2
17 //Calculating the net load on the piston
18 FL=p1*A1-p2*(A1-a) //N
19 //Calculating the ratio of lengths of the connecting
    rod and crank
20 n=1/r
21 //Calculating the inertia force on the reciprocating
    parts
22 FI=mR*omega^2*r*(cosd(theta)+cosd(2*theta)/n) //N
23 //Calculating the net force on the piston or piston
    effort
24 FP=(FL-FI)/1000 //kN
25 //Calculating the angle of inclination of the
    connecting rod to the line of stroke
26 phi=asind(sind(theta)/n) //degrees
27 //Calculating the turning moment on the crank shaft
28 T=(FP*sind(theta+phi))/cosd(phi)*r*1000 //N-m
29 //Calculating the thrust on the bearings
30 FB=(FP*cosd(theta+phi))/cosd(phi) //kN
31 //Acceleration of the flywheel:
32 P=20*1000 //W
33 m=60 //kg
34 k=0.6 //m
35 //Calculating the mass moment of inertia of the
    flywheel
36 I=m*k^2 //kg-m^2
37 //Calculating the resisting torque
38 TR=P*60/(2*pi*N) //N-m
39 //Calculating the acceleration of the flywheel
40 alpha=(T-TR)/I //rad/s^2
41 //Results:
42 printf("\n\n Turning moment on the crank shaft , T =
    %d N-m.\n\n",T)
43 printf(" Thrust on the bearings , FB = %.2f kN.\n\n",

```

```

    FB)
44 printf(" Acceleration of the flywheel , alpha = %.1f
    rad/s ^2.\n\n",alpha)

```

---

**Scilab code Exa 15.13** To find effort thrust and turning moment

```

1 //To find effort , thrust and turning moment
2 clc
3 //Given:
4 D=300/1000 , L=500/1000 , r=L/2 //m
5 n=4.5
6 N=180 //rpm
7 mR=280 //kg
8 theta=45 //degrees
9 p1=0.1 //N/mm^2
10 CR=14 //Compression ratio , V1/V2
11 //Solution:
12 //Refer Fig. 15.12
13 //Calculating the angular speed of the crank
14 omega=2*pi*N/60 //rad/s
15 //Calculating the pressure corresponding to point 2
16 p2=p1*(CR)^1.35 //N/mm^2
17 //Calculating the swept volume
18 VS=%pi/4*D^2*L //m^3
19 //Calculating the clearance volume
20 VC=VS/(CR-1) //m^3
21 //Calculating the volume corresponding to point 3
22 V3=VC+(1/10*VS) //m^3
23 //Calculating the displacement of the piston
    corresponding to crank displacement of 45 degrees
24 x=r*((1-cosd(theta))+(sind(theta))^2/(2*n)) //m
25 //Calculating the volume corresponding to point 4'
26 V4dash=VC+(%pi/4*D^2*x) //m^2
27 //Calculating the pressure corresponding to point 4'
28 p3=p2

```

```

29 p4dash=p3*(V3/V4dash)^1.35 //N/mm^2
30 //Calculating the difference of pressures on two
    sides of the piston
31 p=(p4dash-p1)*10^6 //N/m^2
32 //Calculating the net load on the piston
33 FL=p*pi/4*D^2 //N
34 //Calculating the inertia force on the reciprocating
    parts
35 FI=mR*omega^2*r*(cosd(theta)+cosd(2*theta)/n) //N
36 //Calculating the net force on the piston or piston
    effort
37 FP=FL-FI+mR*9.81 //N
38 //Crank-pin effort:
39 //Calculating the angle of inclination of the
    connecting rod to the line of stroke
40 phi=asind(sind(theta)/n) //degrees
41 //Calculating the crank-pin effort
42 FT=(FP*sind(theta+phi))/(cosd(phi)*1000) //kN
43 //Calculating the thrust on the bearings
44 FB=(FP*cosd(theta+phi))/(cosd(phi)*1000) //kN
45 //Calculating the turning moment on the crankshaft
46 T=FT*r //kN-m
47 //Results:
48 printf("\n\n Crank-pin effort , FT = %.3f kN.\n\n",FT
    )
49 printf(" Thrust on the bearings , FB = %.3f kN.\n\n",
    FB)
50 printf(" Turning moment on the crankshaft , T = %.2f
    kN-m.\n\n",T)

```

---

**Scilab code Exa 15.14** To find turning moment

```

1 //To find turning moment
2 clc
3 //Given:

```

```

4 D=240/1000, L=360/1000, r=L/2, l=0.6 //m
5 N=300 //rpm
6 mR=160 //kg
7 pA=(8+1.03)*10^5, pE=(-0.75+1.03)*10^5 //N/m^2
8 FR=500 //N
9 theta=75 //degrees
10 //Solution:
11 //Refer Fig. 15.13
12 //Calculating the angular speed of the crank
13 omega=2*pi*N/60 //rad/s
14 //Calculating the stroke volume
15 VS=%pi/4*D^2*L //m^3
16 //Calculating the volume of steam at cut-off
17 VB=VS/3 //m^3
18 //Calculating the ratio of lengths of the connecting
    rod and crank
19 n=l/r
20 //Calculating the displacement of the piston when
    the crank position is 75 degrees from the top
    dead centre
21 x=r*((1-cosd(theta))+(sind(theta))^2/(2*n)) //m^3
22 //Calculating the volume corresponding to point C'
23 VCdash=VS*x/L //m^3
24 //Calculating the pressure corresponding to point C'
25 pB=pA
26 pCdash=round((pB*VB)/VCdash) //N/m^2
27 //Calculating the difference of pressures on the two
    sides of the piston
28 p=round(pCdash-pE) //N/m^2
29 //Calculating the net load on the piston
30 FL=round(%pi/4*D^2*p) //N
31 //Calculating the inertia force on the reciprocating
    parts
32 FI=round(mR*omega^2*r*(cosd(theta)+(cosd(2*theta)/n)
    )) //N
33 //Calculating the piston effort
34 FP=FL-FI+mR*9.81-FR //N
35 //Turning moment on the crankshaft:

```

```

36 //Calculating the angle of inclination of the
    connecting rod to the line of stroke
37 phi=asind(sind(theta)/n) //degrees
38 //Calculating the turning moment on the crankshaft
39 T=(FP*sind(theta+phi))/cosd(phi)*r //N-m
40 //Results:
41 printf("\n\n Turning moment on the crankshaft , T =
    %d N-m.\n\n",T)

```

---

**Scilab code Exa 15.15** To find equivalent system

```

1 //To find equivalent system
2 clc
3 //Given:
4 l=300, l1=200 //mm
5 m=15 //kg
6 I=7000 //kg-mm^2
7 //Solution:
8 //Refer Fig. 15.16 and Fig. 15.17
9 //Calculating the radius of gyration of the
    connecting rod about an axis passing through its
    centre of gravity
10 kG=sqrt(I/m) //mm
11 //Calculating the distance of other mass from the
    centre of gravity
12 l2=(kG)^2/l1 //mm
13 //Calculating the magnitude of mass placed at the
    small end centre
14 m1=(l2*m)/(l1+l2) //kg
15 //Calculating the magnitude of the mass placed at a
    distance l2 from the centre of gravity
16 m2=(l1*m)/(l1+l2) //kg
17 //Results:
18 printf("\n\n Mass placed at the small end centre , m1
    = %.2f kg.\n\n",m1)

```

```
19 printf(" Mass placed at a distance %.2 f mm from the
    centre of gravity , m2 = %.2 f kg.\n\n",l2,m2)
```

---

**Scilab code Exa 15.16** To find equivalent system

```
1 //To find equivalent system
2 clc
3 //Given:
4 h=650/1000 , l1=(650-25)/1000 //m
5 m=37.5 //kg
6 tp=1.87 //seconds
7 //Solution:
8 //Refer Fig. 15.18 and Fig. 15.19
9 //Calculating the radius of gyration of the
    connecting rod about an axis passing through its
    centre of gravity
10 kG=sqrt((tp/(2*pi))^2*(9.81*h)-h^2) //m
11 //Calculating the distance of mass m2 from the
    centre of gravity
12 l2=(kG)^2/l1 //m
13 //Calculating the magnitude of mass placed at the
    small end centre
14 m1=(l2*m)/(l1+l2) //kg
15 //Calculating the magnitude of mass placed at a
    distance l2 from centre of gravity
16 m2=(l1*m)/(l1+l2) //kg
17 //Results:
18 printf("\n\n Mass placed at the small end centre A,
    m1 = %d kg.\n\n",m1)
19 printf(" Mass placed at a distance %.3 f m from G, m2
    = %.1 f kg.\n\n",l2,m2)
```

---

**Scilab code Exa 15.17** To find radius and MI

```

1 //To find radius and MI
2 clc
3 //Given:
4 m=55 //kg
5 l=850/1000, d1=75/1000, d2=100/1000 //m
6 tp1=1.83, tp2=1.68 //seconds
7 //Solution:
8 //Refer Fig. 15.20
9 //Calculating the length of equivalent simple
    pendulum when suspended from the top of small end
    bearing
10 L1=9.81*(tp1/(2*%pi))^2 //m
11 //Calculating the length of equivalent simple
    pendulum when suspended from the top of big end
    bearing
12 L2=9.81*(tp2/(2*%pi))^2 //m
13 //Radius of gyration of the rod about an axis
    passing through the centre of gravity and
    perpendicular to the plane of oscillation:
14 //Calculating the distances of centre of gravity
    from the top of small end and big end bearings
15 //We have,  $h_1*(L_1-h_1) = h_2*(L_2-h_2)$ , or  $h_1^2-h_2^2+h_2*
    L_2-h_1*L_1 = 0$ 

    ..... ( i )
16 //Also,  $h_1+h_2 = d_1/2+l+d_2/2$ , or  $h_1+h_2-d_1/2-l-d_2/2 =
    0$ 

    ..... ( ii )
17 function y=f(x)
18     h1=x(1)
19     h2=x(2)
20     y(1)=h1^2-h2^2+h2*L2-h1*L1
21     y(2)=h1+h2-d1/2-l-d2/2
22 endfunction
23 z=fsolve([1,1],f)
24 h1=z(1), h2=z(2) //m
25 //Calculating the required radius of gyration of the

```



```

        rod
26 kG=sqrt(h1*(L1-h1)) //m
27 //Calculating the moment of inertia of the rod
28 I=m*(kG)^2 //kg-m^2
29 //Dynamically equivalent system for the rod:
30 //Calculating the distance of the mass situated at
    the centre of small end bearing from the centre
    of gravity
31 l1=h1-d1/2 //m
32 //Calculating the distance of the second mass from
    the centre of gravity towards big end bearing
33 l2=(kG)^2/l1 //m
34 //Calculating the magnitude of the mass situated at
    the centre of small end bearing
35 m1=(l2*m)/(l1+l2) //kg
36 //Calculating the magnitude of the second mass
37 m2=(l1*m)/(l1+l2) //kg
38 //Results:
39 printf("\n\n Radius of gyration of the rod about an
    axis passing through the centre of gravity and
    perpendicular to the plane of oscillation , kG = %
    .3f m.\n\n",kG)
40 printf(" Moment of inertia of the rod , I = %.2f kg-m
    ^2.\n\n",I)
41 printf(" Magnitude of the mass situated at the
    centre of small end bearing , m1 = %.2f kg.\n\n",
    m1)
42 printf(" Magnitude of the second mass , m2 = %.2f kg
    .\n\n",m2)

```

---

**Scilab code Exa 15.18** To find correcting couple

```

1 //To find correcting couple
2 clc
3 //Given:

```

```

4 m=2 //kg
5 l=250/1000, l1=100/1000, kG=110/1000 //m
6 alpha=23000 //rad/s^2
7 //Solution:
8 //Equivalent dynamical system:
9 //Calculating the distance of the second mass from
  the centre of gravity
10 l2=(kG)^2/l1 //m
11 //Calculating the magnitude of the mass placed at
  the gudgeon pin
12 m1=(l2*m)/(l1+l2) //kg
13 //Calculating the magnitude of the mass placed at a
  distance l2 from centre of gravity
14 m2=(l1*m)/(l1+l2) //kg
15 //Correction couple:
16 //Calculating the magnitude of l3
17 l3=l-l1 //m
18 //Calculating the new radius of gyration
19 k1=sqrt(l1*l3) //m^2
20 //Calculating the correction couple
21 Tdash=m*(k1^2-kG^2)*alpha //N-m
22 //Results:
23 printf("\n\n Mass placed at the gudgeon pin , m1 = %
  .1 f kg.\n\n",m1)
24 printf(" Mass placed at a distance %.3 f m from the
  centre of gravity , m2 = %.1 f kg.\n\n",l2,m2)
25 printf(" Correction couple , Tdash = %.1 f N-m.\n\n",
  Tdash)

```

---

**Scilab code Exa 15.19** To find acceleration and inertia force

```

1 //To find acceleration and inertia force
2 clc
3 //Given:
4 r=125, OC=r, l=500, PC=1, PG=275, kG=150 //mm

```

```

5 mC=60 //kg
6 N=600 //rpm
7 theta=45 //degrees
8 //Solution:
9 //Refer Fig. 15.24
10 //Calculating the angular speed of the crank
11 omega=2*pi*N/60 //rad/s
12 //Acceleration of the piston:
13 //By measurement,
14 NO=90/1000 //m
15 //Calculating the acceleration of the piston
16 aP=omega^2*NO //m/s^2
17 //The magnitude, position and direction of inertia
    force due to the mass of the connecting rod:
18 //By measurement,
19 g0=103/1000 //m
20 //Calculating the magnitude of the inertia force of
    the connecting rod
21 FC=mC*omega^2*g0/1000 //kN
22 //Results:
23 printf("\n\n Acceleration of the piston, aP = %.1f m
    /s ^2.\n\n",aP)
24 printf(" The magnitude of inertia force due to the
    mass of the connecting rod, FC = %.1f kN.\n\n",FC
    )

```

---

**Scilab code Exa 15.20** To find torque exerted

```

1 //To find torque exerted
2 clc
3 //Given:
4 D=240/1000, L=600/1000, r=L/2, l=1.5, GC=500/1000,
    kG=650/1000 //m
5 mR=300, mC=250 //kg
6 N=125 //rpm

```

```

7 theta=30 //degrees
8 //Solution:
9 //Refer Fig. 15.25
10 //Calculating the angular speed of the crank
11 omega=2*pi*N/60 //rad/s
12 //Analytical method:
13 //Calculating the distance of centre of gravity of
    the connecting rod from P
14 l1=1-GC //m
15 //Calculating the ratio of lengths of the connecting
    rod and crank
16 n=1/r
17 //Calculating the inertia force due to total mass of
    the reciprocating parts at P
18 FI=(mR+(1-l1)/l*mC)*omega^2*r*(cosd(theta)+cosd(2*
    theta)/n) //N
19 //Calculating the corresponding torque due to FI
20 TI=FI*r*(sind(theta)+sind(2*theta)/(2*sqrt(n^2-(sind
    (theta))^2))) //N-m
21 //Calculating the equivalent length of a simple
    pendulum when swung about an axis through P
22 L=((kG)^2+(l1)^2)/l1 //m
23 //Calculating the correcting torque
24 TC=mC*l1*(1-L) //N-m
25 //Calculating the torque due to the weight of the
    connecting rod at C
26 TW=mC*9.81*(l1/n)*cosd(theta) //N-m
27 //Calculating the total torque exerted on the
    crankshaft
28 Tt=TI+TC+TW //Total torque exerted on the crankshaft
    , N-m
29 //Results:
30 printf("\n\n Total torque exerted on the crankshaft
    = %.1f N-m.\n\n",Tt)

```

---

**Scilab code Exa 15.21** To find acceleration and inertia torque

```
1 //To find acceleration and inertia torque
2 clc
3 //Given:
4 N=1200 //rpm
5 L=110/1000, r=L/2, l=250/1000, PC=1, CG=75/1000 //m
6 mC=1.25 //kg
7 theta=40 //degrees
8 //Solution:
9 //Refer Fig. 15.26
10 //Calculating the angular speed of the crank
11 omega=2*pi*N/60 //rad/s
12 //Radius of gyration of the connecting rod about an
    axis through its mass centre:
13 //Calculating the distance of the centre of gravity
    from the point of suspension
14 l1=1-CG //m
15 PG=l1
16 //Calculating the frequency of oscillation
17 n=21/20 //Hz
18 //Calculating the radius of gyration of the
    connecting rod about an axis through its mass
    centre
19 kG=round(sqrt((9.81*l1/(2*pi*n)^2)-l1^2)*1000) //mm
20 //Acceleration of the piston:
21 //Calculating the ratio of lengths of the connecting
    rod and crank
22 n=1/r
23 //Calculating the acceleration of the piston
24 aP=omega^2*r*(cosd(theta)+cosd(2*theta)/n) //m/s^2
25 //Calculating the angular acceleration of the
    connecting rod
26 alphaPC=(-omega^2*sind(theta))/n //rad/s^2
27 //Inertia torque exerted on the crankshaft:
28 //Calculating the mass of the connecting rod at P
29 m1=(1-l1)/l*mC //kg
30 //Calculating the vertical inertia force
```

```

31 FI=round(m1*aP) //N
32 //By measurement ,
33 OM=0.0425, NC=0.035 //m
34 //Calculating the corresponding torque due to FI
35 TI=FI*OM //N-m
36 //Calculating the equivalent length of a simple
    pendulum when swung about an axis passing through
    P
37 L=((kG/1000)^2+(l1)^2)/l1 //m
38 //Calculating the correction couple
39 Tdash=mC*l1*(1-L)*alphaPC //N-m
40 //Calculating the corresponding torque on the
    crankshaft
41 TC=-Tdash*cosd(theta)/n //N-m
42 //Calculating the torque due to mass at P
43 TP=m1*9.81*OM //N-m
44 //Calculating the equivalent mass of the connecting
    rod at C
45 m2=mC*(l1/l) //kg
46 //Calculating the torque due to mass at C
47 TW=m2*9.81*NC //N-m
48 //Calculating the inertia force exerted on the
    crankshaft
49 Ti=TI+TC-TP-TW //Inertia torque exerted on the
    crankshaft , N-m
50 //Results :
51 printf("\n\n Radius of gyration of the connecting
    rod about an axis through its mass centre , kG =
    %d mm.\n\n",kG)
52 printf(" Acceleration of the piston , aP = %.1f m/s
    ^2.\n\n",aP)
53 printf(" Angular acceleration of the connecting rod ,
    alphaPC = %.1f rad/s^2.\n\n",alphaPC)
54 printf(" Inertia torque exerted on the crankshaft =
    %.3f N-m.\n\n",Ti)

```

---

**Scilab code Exa 15.22** To find resultant force

```
1 //To find resultant force
2 clc
3 //Given:
4 l=225/1000, PC=1, L=150/1000, r=L/2, D=112.5/1000,
   PG=150/1000, kG=87.5/1000 //m
5 mC=1.6, mR=2.4 //kg
6 theta=40 //degrees
7 p=1.8*10^6 //N/m^2
8 N=2000 //rpm
9 //Solution:
10 //Refer Fig. 15.27
11 //Calculating the angular speed of the crank
12 omega=2*%pi*N/60 //rad/s
13 //By measurement,
14 NO=0.0625, g0=0.0685, IC=0.29, IP=0.24, IY=0.148, IX
   =0.08 //m
15 //Calculating the force due to gas pressure
16 FL=%pi/4*D^2*p //N
17 //Calculating the inertia force due to mass of the
   reciprocating parts
18 FI=mR*omega^2*NO //N
19 //Calculating the net force on the piston
20 FP=FL-FI //N
21 //Calculating the inertia force due to mass of the
   connecting rod
22 FC=mC*omega^2*g0 //N
23 //Calculating the force acting perpendicular to the
   crank OC
24 FT=((FP*IP)-((mC*9.81*IY)+(FC*IX)))/IC //N
25 //By measurement,
26 FN=3550, FR=7550, FQ=13750 //N
27 //Results:
```

```
28 printf("\n\n Resultant force on the crank pin , FQ =  
    %d N.\n\n",FQ)
```

---



# Chapter 16

## Turning Moment Diagrams and Flywheel

Scilab code Exa 16.1 To find maximum and minimum speeds

```
1 //To find maximum and minimum speeds
2 clc
3 //Given:
4 m=6.5*1000 //kg
5 k=1.8 //m
6 deltaE=56*1000 //N-m
7 N=120 //rpm
8 //Solution:
9 //Calculating the maximum and minimum speeds
10 //We know that fluctuation of energy,  $\Delta E = \frac{\pi^2}{900} m k^2 N (N_1 - N_2)$ , or  $N_1 - N_2 = \frac{\Delta E}{(\frac{\pi^2}{900} m k^2 N)}$  ..... ( i )
11 //Also mean speed,  $N = \frac{(N_1 + N_2)}{2}$ , or  $N_1 + N_2 = 2 * N$ 
    ..... ( ii )
12 A=[1 -1; 1 1]
13 B=[deltaE/(%pi^2/900*m*k^2*N); 2*N]
14 V=A \ B
15 N1=round(V(1)) //rpm
```

```

16 N2=round(V(2)) //rpm
17 //Results:
18 printf("\n\n Maximum speed , N1 = %d rpm.\n\n",N1)
19 printf(" Minimum speed , N2 = %d rpm.\n\n",N2)

```

---

**Scilab code Exa 16.2** To find angular acceleration and KE

```

1 //To find angular acceleration and KE
2 clc
3 //Given:
4 k=1 //m
5 m=2500 //kg
6 T=1500 //N-m
7 //Solution:
8 //Angular acceleration of the flywheel:
9 //Calculating the mass moment of inertia of the
  flywheel
10 I=m*k^2 //kg-m^2
11 //Calculating the angular acceleration of the
  flywheel
12 alpha=T/I //rad/s^2
13 //Kinetic energy of the flywheel:
14 omega1=0 //Angular speed at rest
15 //Calculating the angular speed after 10 seconds
16 omega2=omega1+alpha*10 //rad/s
17 //Calculating the kinetic energy of the flywheel
18 KE=1/2*I*(omega2)^2/1000 //Kinetic energy of the
  flywheel , kN-m
19 //Results:
20 printf("\n\n Angular acceleration of the flywheel ,
  alpha = %.1f rad/s^2.\n\n",alpha)
21 printf(" Kinetic energy of the flywheel = %d kN-m.\n
  \n",KE)

```

---

**Scilab code Exa 16.3** To find weight of flywheel

```
1 //To find weight of flywheel
2 clc
3 //Given:
4 P=300*1000 //W
5 N=90 //rpm
6 CE=0.1
7 k=2 //m
8 //Solution:
9 //Calculating the mean angular speed
10 omega=2*%pi*N/60 //rad/s
11 //Calculating the coefficient of fluctuation of
    speed
12 CS=1/100
13 //Calculating the work done per cycle
14 WD=P*60/N //Work done per cycle, N-m
15 //Calculating the maximum fluctuation of energy
16 deltaE=WD*CE //N-m
17 //Calculating the mass of the flywheel
18 m=deltaE/(k^2*omega^2*CS) //kg
19 //Results:
20 printf("\\n\\n Mass of the flywheel , m = %d kg.\\n\\n",m
    )
```

---

**Scilab code Exa 16.4** To find coefficient of fluctuation of speed

```
1 //To find coefficient of fluctuation of speed
2 clc
3 //Given:
4 m=36 //kg
5 k=150/1000 //m
```

```

6 N=1800 //rpm
7 //Solution:
8 //Refer Fig. 16.6
9 //Calculating the angular speed of the crank
10 omega=2*pi*N/60 //rad/s
11 //Calculating the value of 1 mm^2 on the turning
    moment diagram
12 c=5*pi/180 //Value of 1 mm^2 on turning miment
    diagram, N-m
13 //Calculating the maximum fluctuation of energy
14 //From the turning moment diagram, maximum energy =
    E+295, and minimum energy = E-690
15 deltaE=(285-(-690))*c //N-m
16 //Calculating the coefficient of fluctuation of
    energy
17 CS=deltaE/(m*k^2*omega^2)*100 //%
18 //Results:
19 printf("\n\n Coefficient of fluctuation of speed, CS
    = %.1 f %c.\n\n",CS,"%")

```

---

**Scilab code Exa 16.5** To find mass of the flywheel

```

1 //To find mass of the flywheel
2 clc
3 //Given:
4 N=600 //rpm
5 R=0.5 //m
6 //Solution:
7 //Refer Fig. 16.7
8 //Calculating the angular speed of the crank
9 omega=2*pi*N/60 //rad/s
10 //Calculating the coefficient of fluctuation of
    speed
11 CS=3/100
12 //Calculating the value of 1 mm^2 on turning moment

```

```

    diagram
13 c=600*%pi/60 //Value of 1 mm^2 on turning moment
    diagram, N-m
14 //Calculating the maximum fluctuation of energy
15 //From the turning moment diagram, maximum
    fluctuation = E+52, and minimum fluctuation = E
    -120
16 deltaE=(52-(-120))*c //N-m
17 //Calculating the mass of the flywheel
18 m=deltaE/(R^2*omega^2*CS) //kg
19 //Results:
20 printf("\n\n Mass of the flywheel , m = %d kg.\n\n",m
    )

```

---

**Scilab code Exa 16.6** To find power and speed fluctuation

```

1 //To find power and speed fluctuation
2 clc
3 //Given:
4 N=250 //rpm
5 m=500 //kg
6 k=600/1000 //m
7 //Solution:
8 //Refer Fig. 16.8
9 //Calculating the angular speed of the crank
10 omega=2*%pi*N/60 //rad/s
11 //Calculating the torque required for one complete
    cycle
12 T=(6*%pi*750)+(1/2*%pi*(3000-750))+(2*%pi*(3000-750)
    )+(1/2*%pi*(3000-750)) //N-m
13 //Calculating the mean torque
14 Tmean=T/(6*%pi) //N-m
15 //Calculating the power required to drive the
    machine
16 P=Tmean*omega/1000 //kW

```

```

17 // Coefficient of fluctuation of speed:
18 // Calculating the value of LM
19 LM=%pi*((3000-1875)/(3000-750))
20 // Calculating the value of NP
21 NP=%pi*((3000-1875)/(3000-750))
22 // Calculating the value of BM
23 BM=3000-1875 //N-m
24 CN=BM
25 // Calculating the value of MN
26 MN=2*%pi
27 // Calculating the maximum fluctuation of energy
28 deltaE=(1/2*LM*BM)+(MN*BM)+(1/2*NP*CN) //N-m
29 // Calculating the coefficient of fluctuation of
    speed
30 CS=deltaE/(m*k^2*omega^2)
31 // Results:
32 printf("\n\n Power required to drive the machine, P
    = %.3f kW.\n\n",P)
33 printf(" Coefficient of speed, CS = %.3f.\n\n",CS)

```

---

**Scilab code Exa 16.7** To find coefficient of fluctuation

```

1 //To find coefficient of fluctuation
2 clc
3 //Given:
4 N=100 //rpm
5 k=1.75 //m
6 //Solution:
7 //Refer Fig. 16.9
8 //Calculating the angular speed of the crank
9 omega=2*%pi*N/60 //rad/s
10 //Calculating the coefficient of fluctuation of
    speed
11 CS=1.5/100
12 // Coefficient of fluctuation of energy:

```

```

13 AB=2000, LM=1500 //N-m
14 //Calculating the work done per cycle
15 WD=(1/2*%pi*AB)+(1/2*%pi*LM) //Work done per cycle,
    N-m
16 //Calculating the mean resisting torque
17 Tmean=WD/(2*%pi) //N-m
18 //Calculating the value of CD
19 CD=%pi/2000*(2000-875) //rad
20 //Calculating the maximum fluctuation of energy
21 deltaE=1/2*CD*(2000-875) //N-m
22 //Calculating the coefficient of fluctuation of
    energy
23 Ce=deltaE/WD*100 //%
24 //Calculating the mass of the flywheel
25 m=deltaE/(k^2*omega^2*CS) //kg
26 //Crank angles for minimum and maximum speeds:
27 //Calculating the value of CE
28 CE=(2000-875)/2000*(4*%pi/9) //rad
29 //Calculating the crank angle for minimum speed
30 thetaC=((4*%pi/9)-CE)*180/%pi //degrees
31 //Calculating the value of ED
32 ED=(2000-875)/2000*(%pi-(4*%pi/9)) //rad
33 //Calculating the crank angle for maximum speed
34 thetaD=((4*%pi/9)+ED)*180/%pi //degrees
35 //Results:
36 printf("\n\n Coefficient of fluctuation of energy ,
    CE = %d %c.\n\n",Ce,"%")
37 printf(" Mass of the flywheel , m = %.1f kg.\n\n",m)
38 printf(" Crank angle from IDC for the minimum speed ,
    thetaC = %d degrees.\n\n",thetaC)
39 printf(" Crank angle from IDC for the maximum speed ,
    thetaD = %d degrees.\n\n",thetaD)

```

---

**Scilab code Exa 16.8** To find power and coefficients

```

1 //To find power and coefficients
2 clc
3 //Given:
4 N=600 //rpm
5 Tmax=90 //N-m
6 m=12 //kg
7 k=80/1000 //m
8 //Solution:
9 //Refer Fig. 16.10
10 //Calculating the angular speed of the crank
11 omega=2*%pi*N/60 //rad/s
12 //Power developed:
13 //Calculating the work done per cycle
14 WD=3*1/2*%pi*90 //Work done per cycle, N-m
15 //Calculating the mean torque
16 Tmean=WD/(2*%pi) //N-m
17 //Calculating the power developed
18 P=Tmean*omega/1000 //Power developed, kW
19 //Coefficient of fluctuation of speed:
20 //Calculating the maximum fluctuation of energy
21 //From the torque-crank angle diagram, maximum
    energy=E+5.89, and minimum energy=E-5.89
22 deltaE=5.89-(-5.89) //N-m
23 //Calculating the coefficient of fluctuation of
    speed
24 CS=round(deltaE/(m*k^2*omega^2)*100) // %
25 //Calculating the coefficient of fluctuation of
    energy
26 CE=deltaE/WD*100 // %
27 //Calculating the maximum angular acceleration of
    the flywheel
28 alpha=(Tmax-Tmean)/(m*k^2) //rad/s^2
29 //Results:
30 printf(" \n\n Power developed = %.2f kW.\n\n",P)
31 printf(" Coefficient of fluctuation of speed, CS =
    %d %c.\n\n",CS,"%")
32 printf(" Coefficient of fluctuation of energy, CE =
    %.2f %c.\n\n",CE,"%")

```



```

33 printf(" Maximum angular acceleration of the
    flywheel , alpha = %d rad/s^2.\n\n",alpha)

```

---

**Scilab code Exa 16.9** To find moment of inertia

```

1 //To find moment of inertia
2 clc
3 //Given:
4 P=20*1000 //W
5 N=300 //rpm
6 //Solution:
7 //Refer Fig. 16.11
8 //Calculating the angular speed of the crank
9 omega=2*%pi*N/60 //ra/s
10 //Calculating the coefficient of fluctuation of
    speed
11 CS=4/100
12 //Calculating the number of working strokes per
    cycle for a four stroke engine
13 n=N/2
14 //Calculating the work done per cycle
15 WD=P*60/n //Work done per cycle , N-m
16 //Calculating the work done during expansion cycle
17 WE=WD*3/2 //N-m
18 //Calculating the maximum turning moment
19 Tmax=WE*2/%pi //N-m
20 //Calculating the mean turning moment
21 Tmean=WD/(4*%pi) //N-m
22 //Calculating the excess turning moment
23 Texcess=Tmax-Tmean //N-m
24 //Calculating the value of DE
25 DE=Texcess/Tmax*%pi //rad
26 //Calculating the maximum fluctuation of energy
27 deltaE=(1/2*DE*Texcess) //N-m
28 //Calculating the moment of inertia of the flywheel

```

```

29 I=deltaE/(omega^2*CS) //kg-m^2
30 //Results:
31 printf("\n\n Moment of inertia of the flywheel, I =
    %.1f kg-m^2.\n\n",I)

```

---

**Scilab code Exa 16.10** To find mass of the rim

```

1 //To find mass of the rim
2 clc
3 //Given:
4 a1=0.45*10^-3, a2=1.7*10^-3, a3=6.8*10^-3, a4
    =0.65*10^-3 //m^2
5 N1=202, N2=198 //rpm
6 R=1.2 //m
7 //Solution:
8 //Refer Fig. 16.12
9 //Calculating the net area
10 a=a3-(a1+a2+a4) //Net area, m^2
11 //Calculating the energy scale constant
12 c=3*10^6 //Energy scale constant, N-m
13 //Calculating the net work done per cycle
14 WD=a*c //Net work done per cycle, N-m
15 //Calculating the mean torque
16 Tmean=WD/(4*%pi) //N-m
17 //Calculating the value of FG
18 FG=Tmean //N-m
19 //Calculating the work done during expansion stroke
20 WDe=a3*c //Work done during expansion stroke, N-m
21 //Calculating the value of AG
22 AG=WDe/(1/2*%pi) //N-m
23 //Calculating the excess torque
24 Texcess=AG-FG //N-m
25 //Calculating the value of AF
26 AF=Texcess //N-m
27 //Calculating the value of DE

```

```

28 DE=AF/AG*%pi //rad
29 //Calculating the maximum fluctuation of energy
30 deltaE=1/2*DE*AF //N-m
31 //Mass of the rim of a flywheel:
32 //Calculating the mean speed of the flywheel
33 N=(N1+N2)/2 //rpm
34 //Calculating the mass of the rim of a flywheel
35 m=deltaE/(%pi^2/900*R^2*N*(N1-N2)) //kg
36 //Results:
37 printf("\n\n Mass of the rim of the flywheel , m = %d
      kg.\n\n",m)

```

---

**Scilab code Exa 16.12** To find fluctuation of energy and speed

```

1 //To find fluctuation of energy and speed
2 clc
3 //Given:
4 m=500 //kg
5 k=0.4 //m
6 N=150 //rpm
7 //Solution:
8 //Refer Fig. 16.14
9 //Calculating the angular speed of the crank
10 omega=2*%pi*N/60 //rad/s
11 //Fluctuation of energy:
12 //Equating the change in torque to zero and
    calculating the value of theta
13 thetaA=asind(0), thetaC=asind(0)+180, thetaE=asind
    (0)+360 //degrees
14 thetaB=acosd(1/(2*(600/500))), thetaD=360-acosd
    (1/(2*(600/500))) //degrees
15 //Calculating the maximum fluctuation of energy
16 deltaE=round(integrate('(5000+600*sin(2*theta))
    -(5000+500*sin(theta))', 'theta', thetaC*%pi/180,
    thetaD*%pi/180)) //N-m

```

```

17 //Calculating the total percentage fluctuation of
    speed
18 CS=deltaE/(m*k^2*omega^2)*100 //%
19 //Maximum and minimum angular acceleration of the
    flywheel and the corresponding shaft positions:
20 //Calculating the maximum or minimum values of theta
21 //Differentiating (600*sin(2*theta))-500*sin(theta)
    = 0 with respect to theta and equating to zero,
    we get 12*2*(cosd(theta))^2-5*cosd(theta)-12 = 0
22 a=12*2, b=-5, c=-12
23 cosdtheta1=(-b+sqrt(b^2-4*a*c))/(2*a)
24 cosdtheta2=(-b-sqrt(b^2-4*a*c))/(2*a)
25 theta1=round(acosd(cosdtheta1)), theta2=acosd(
    cosdtheta2) //degrees
26 //Calculating the maximum torque
27 Tmax=600*sind(2*theta1)-500*sind(theta1) //N-m
28 //Calculating the minimum torque
29 Tmin=600*sind(2*theta2)-500*sind(theta2) //N-m
30 //Calculating the maximum acceleration
31 alphamax=Tmax/(m*k^2) //rad/s^2
32 //Calculating the minimum acceleration
33 alphamin=abs(Tmin)/(m*k^2) //rad/s^2
34 //Results:
35 printf("\n\n Fluctuation of energy, deltaE = %d N-m
    .\n\n",deltaE)
36 printf(" Total percentage fluctuation of speed, CS =
    %.1f %c.\n\n",CS,"%")
37 printf(" Shaft position corresponding to maximum and
    minimum accelerations, theta = %d degrees and %
    .1f degrees.\n\n",theta1,theta2)
38 printf(" Maximum acceleration, alphamax = %.2f rad/s
    ^2.\n\n",alphamax)
39 printf(" Minimum acceleration, alphamin = %.1f rad/s
    ^2.\n\n",alphamin)

```

---

**Scilab code Exa 16.13** To find power fluctuation and torque

```

1 //To find power, fluctuation and torque
2 clc
3 //Given:
4 I=1000 //kg-m2
5 N=300 //rpm
6 //Solution:
7 //Refer Fig. 16.15 and Fig. 16.16
8 //Calculating the angular speed of the crank
9 omega=2*%pi*N/60 //rad/s
10 //Power of the engine:
11 //Calculating the work done per revolution
12 WD=integrate('5000+1500*sin(3*theta)', 'theta', 0, 2*
    %pi) //Work done per cycle, N-m
13 //Calculating the mean resisting torque
14 Tmean=WD/(2*%pi) //N-m
15 //Calculating the power of the engine
16 P=Tmean*omega/1000 //kW
17 //Maximum fluctuation of the speed of the flywheel
    when resisting torque is constant:
18 //Calculating the value of theta
19 sind3theta=(5000-5000)/1500
20 theta=1/3*(asind(sind3theta)+180) //degrees
21 //Calculating the maximum fluctuation of energy
22 deltaE=integrate('5000+1500*sin(3*theta)-5000', '
    theta', 0, 60*%pi/180) //N-m
23 //Calculating the maximum fluctuation of speed of
    the flywheel
24 CS1=deltaE/(I*omega2)*100 //%
25 //Maximum fluctuation of speed of the flywheel when
    resisting torque (5000+600*sin(theta)) N-m:
26 //Calculating the values of theta, thetaB and thetaC
27 thetaB=asind(sqrt((1/4*(3-600/1500)))) //degrees
28 thetaC=180-thetaB //degrees
29 //Calculating the maximum fluctuation of energy
30 deltaE=round(integrate('(5000+1500*sin(3*theta))
    -(5000+600*sin(theta))', 'theta', thetaB*%pi/180,

```

```

    thetaC*%pi/180)) //N-m
31 //Calculating the maximum fluctuation of speed of
    the flywheel
32 CS2=abs(deltaE)/(I*omega^2)*100 //%
33 //Results:
34 printf("\n\n Power of the engine , P = %.1f kW.\n\n",
    P)
35 printf(" Maximum fluctuation of the speed of the
    flywheel when resisting torque is constant, CS =
    %.1f %c.\n\n",CS1,"%")
36 printf(" Maximum fluctuation of speed of the
    flywheel when resisting torque (5000+600*sin(
    theta)) N-m, CS = %.3f %c.\n\n",CS2,"%")

```

---

**Scilab code Exa 16.14** To find diameter and cross section

```

1 //To find diameter and cross section
2 clc
3 //Given:
4 N=800 //rpm
5 stroke=300 //mm
6 sigma=7*10^6 //N/m^2
7 rho=7200 //kg/m^3
8 //Solution:
9 //Refer Fig. 16.18
10 //Calculating the angular speed of the engine
11 omega=2*%pi*N/60 //rad/s
12 //Calculating the coefficient of fluctuation of
    speed
13 CS=4/100
14 //Diameter of the flywheel rim:
15 //Calculating the peripheral velocity of the
    flywheel rim
16 v=sqrt(sigma/rho) //m/s
17 //Calculating the diameter of the flywheel rim

```

```

18 D=v*60/(%pi*N) //m
19 //Cross-section of the flywheel rim:
20 //Calculating the value of 1 mm^2 on the turning
    moment diagram
21 c=500*%pi/30 //Value of 1 mm^2 on the turning moment
    diagram, N-m
22 //Calculating the maximum fluctuation of energy
23 deltaE=round((420-(-30))*c) //N-m
24 //Calculating the mass of the flywheel rim
25 m=deltaE/(v^2*CS) //kg
26 //Calculating the thickness of the flywheel rim
27 t=sqrt(m/(%pi*D*5*rho))*1000 //mm
28 //Calculating the width of the flywheel rim
29 b=5*t //mm
30 //Results:
31 printf("\n\n Diameter of the flywheel rim, D = %.3 f
    m.\n\n",D)
32 printf(" Thickness of the flywheel rim, t = %d mm.\n
    \n",t)
33 printf(" Width of the flywheel rim, b = %d mm.\n\n",
    b)

```

---

**Scilab code Exa 16.15** To find mass and cross section

```

1 //To find mass and cross section
2 clc
3 //Given:
4 P=150*1000 //W
5 N=80 //rpm
6 CE=0.1
7 D=2, R=D/2 //m
8 rho=7200 //kg/m^3
9 //Solution:
10 //Calculating the angular speed of the engine
11 omega=2*%pi*N/60 //rad/s

```

```

12 //Calculating the coefficient of fluctuation of
    speed
13 CS=4/100
14 //Mass of the flywheel rim:
15 //Calculating the work done per cycle
16 WD=P*60/N //Work done per cycle, N-m
17 //Calculating the maximum fluctuation of energy
18 deltaE=WD*CE //N-m
19 //Calculating the mass moment of inertia of the
    flywheel
20 I=deltaE/(omega^2*CS) //kg-m^2
21 //Calculating the mass moment of inertia of the
    flywheel rim
22 Irim=0.95*I //kg-m^2
23 //Calculating the mass of the flywheel rim
24 k=R //Radius of gyration, m
25 m=Irim/k^2 //kg
26 //Calculating the cross-sectional area of the
    flywheel rim
27 A=m/(2*pi*R*rho) //m^2
28 //Results:
29 printf("\n\n Mass of the flywheel rim, m = %d kg.\n\n",m)
30 printf(" Cross-sectional area of the flywheel rim, A
    = %.3 f m^2.\n\n",A)

```

---

**Scilab code Exa 16.16** To find MI and dimensions

```

1 //To find MI and dimensions
2 clc
3 //Given:
4 N=600 //rpm
5 rho=7250 //kg/m^3
6 sigma=6*10^6 //N/m^2
7 //Solution:

```



```

8 //Refer Fig. 16.19
9 //Calculating the angular speed of the engine
10 omega=2*%pi*N/60 //rad/s
11 //Calculating the total fluctuation of speed
12 CS=2/100
13 //Moment of inertia of the flywheel:
14 //Calculating the value of 1 mm^2 of turning moment
    diagram
15 c=250*%pi/60 //Value of 1 mm^2 of turning moment
    diagram , N-m
16 //Calculating the maximum fluctuation of energy
17 deltaE=round((162-(-35))*c) //N-m
18 //Calculating the moment of inertia of the flywheel
19 I=deltaE/(omega^2*CS) //kg-m^2
20 //Dimensions of the flywheel rim:
21 //Calculating the peripheral velocity of the
    flywheel
22 v=sqrt(sigma/rho) //m/s
23 //Calculating the mean diameter of the flywheel
24 D=v*60/(%pi*N) //m
25 //Calculating the maximum fluctuation of energy of
    the flywheel rim
26 deltaErim=0.92*deltaE //N-m
27 //Calculating the mass of the flywheel rim
28 m=deltaErim/(v^2*CS) //kg
29 //Calculating the thickness of the flywheel rim
30 t=sqrt(m/(%pi*D*2*rho))*1000 //mm
31 //Calculating the breadth of the flywheel rim
32 b=2*t //mm
33 //Results:
34 printf("\n\n Moment of inertia of the flywheel , I =
    %.1f kg-m^2.\n\n",I)
35 printf(" Thickness of the flywheel rim , t = %.1f mm
    .\n\n",t)
36 printf(" Breadth of the flywheel rim , b = %.1f mm.\n
    \n",b)

```

---

Scilab code Exa 16.17 To find MI and size

```
1 //To find MI and size
2 clc
3 //Given:
4 a1=5*10^-5, a2=21*10^-5, a3=85*10^-5, a4=8*10^-5 //m
   ^2
5 N2=98, N1=102 //rpm
6 rho=8150 //kg/m^3
7 sigma=7.5*10^6 //N/m^2
8 //Solution:
9 //Refer Fig. 16.20
10 //Calculating the net area
11 a=a3-(a1+a2+a4) //Net area, m^2
12 //Calculating the value of 1 m^2 on the turning
   moment diagram in terms of work
13 c=14*10^6 //Value of 1 m^2 on the turning moment
   diagram, N-m
14 //Calculating the net work done per cycle
15 WD=a*c //Net work done per cycle, N-m
16 //Calculating the mean torque on the flywheel
17 Tmean=WD/(4*%pi) //N-m
18 FG=Tmean //N-m
19 //Calculating the work done during expansion stroke
20 WDe=a3*c //Work done during expansion stroke, N-m
21 //Calculating the value of AG
22 AG=WDe/(1/2*%pi) //N-m
23 //Calculating the excess torque
24 Texcess=AG-FG //Excess torque, N-m
25 AF=Texcess //N-m
26 //Calculating the value of DE
27 DE=AF/AG*%pi //rad
28 //Calculating the maximum fluctuation of energy
29 deltaE=1/2*DE*AF //N-m
```

```

30 //Moment of inertia of the flywheel:
31 //Calculating the mean speed during the cycle
32 N=(N1+N2)/2 //rpm
33 //Calculating the corresponding angular mean speed
34 omega=2*%pi*N/60 //rad/s
35 //Calculating the coefficient of fluctuation of
    speed
36 CS=(N1-N2)/N
37 //Calculating the moment of inertia of the flywheel
38 I=deltaE/(omega^2*CS) //kg-m^2
39 //Size of flywheel:
40 //Calculating the peripheral velocity of the
    flywheel
41 v=sqrt(sigma/rho) //m/s
42 //Calculating the mean diameter of the flywheel
43 D=v*60/(%pi*N) //m
44 //Calculating the mass of the flywheel rim
45 m=deltaE/(v^2*CS) //kg
46 //Calculating the thickness of the flywheel rim
47 t=sqrt(m/(%pi*D*4*rho))*1000 //mm
48 //Calculating the width of the flywheel rim
49 b=4*t //mm
50 //Results:
51 printf("\n\n Moment of inertia of the flywheel , I =
    %d kg-m^2.\n\n",I)
52 printf(" Thickness of the flywheel rim , t = %.1f mm
    .\n\n",t)
53 printf(" Width of the flywheel rim , b = %.1f mm.\n\n
    ",b)

```

---

**Scilab code Exa 16.18** To find diameter and cross section

```

1 //To find diameter and cross section
2 clc
3 //Given:

```

```

4 P=50*1000 //W
5 N=150 //rpm
6 n=75
7 sigma=4*10^6 //N/m^2
8 rho=7200 //kg/m^3
9 //Solution:
10 //Refer Fig. 16.21
11 //Calculating the angular speed of the engine
12 omega=2*%pi*N/60 //rad/s
13 //Calculating the mean torque transmitted by the
    flywheel
14 Tmean=P/omega //N-m
15 FG=Tmean //N-m
16 //Calculating the work done per cycle
17 WD=Tmean*4*%pi //Work done per cycle, N-m
18 //Calculating the work done during power stroke
19 Wdp=1.4*WD //Work done during power stroke, N-m
20 //Calculating the maximum torque transmitted by the
    flywheel
21 Tmax=Wdp/(1/2*%pi) //N-m
22 BF=Tmax //N-m
23 //Calculating the excess torque
24 Texcess=Tmax-Tmean //N-m
25 BG=Texcess //N-m
26 //Calculating the value of DE
27 DE=BG/BF*%pi //N-m
28 //Calculating the maximum fluctuation of energy
29 deltaE=1/2*DE*BG //N-m
30 //Mean diameter of the flywheel:
31 //Calculating the peripheral velocity of the
    flywheel
32 v=sqrt(sigma/rho) //m/s
33 //Calculating the mean diameter of the flywheel
34 D=v*60/(%pi*N) //m
35 //Cross-sectional dimensions of the rim:
36 //Calculating the coefficient of fluctuation of
    speed
37 CS=1/100

```

```

38 //Calculating the total energy of the flywheel
39 E=deltaE/(2*CS) //N-m
40 //Calculating the energy of the rim
41 Erim=15/16*E //N-m
42 //Calculating the mass of the flywheel rim
43 m=Erim/(1/2*v^2) //kg
44 //Calculating the thickness of the rim
45 t=round(sqrt(m/(%pi*D*4*rho))*1000) //mm
46 //Calculating the width of the rim
47 b=4*t //mm
48 //Results:
49 printf("\n\n Mean diameter of the flywheel , D = %d m
      .\n\n",D)
50 printf(" Thickness of the flywheel rim , t = %d mm.\n
      \n",t)
51 printf(" Width of the flywheel rim , b = %d mm.\n\n",
      b)

```

---

**Scilab code Exa 16.19** To find power and mass

```

1 //To find power and mass
2 clc
3 //Given:
4 N1=225, N2=200 //rpm
5 k=0.5 //m
6 E1=15*1000 //N-m
7 HolePunched=720 //per hour
8 //Solution:
9 //Power of the motor:
10 //Calculating the total energy required per second
11 E=E1*HolePunched/3600 //N-m/s
12 //Calculating the power of the motor
13 P=E/1000 //kW
14 //Minimum mass of the flywheel:
15 //Calculating the energy supplied by the motor in 2

```

```

        seconds
16 E2=E*2 //N-m
17 //Calculating the energy supplied by the flywheel
    during punching
18 deltaE=E1-E2 //N-m
19 //Calculating the mean speed of the flywheel
20 N=(N1+N2)/2 //rpm
21 //Calculating the minimum mass of the flywheel
22 m=round(deltaE*900/(%pi^2*k^2*N*(N1-N2))) //kg
23 //Results:
24 printf("\n\n Power of the motor , P = %d kW.\n\n",P)
25 printf(" Minimum mass of the flywheel , m = %d kg.\n\n\
    n" ,m)

```

---

**Scilab code Exa 16.20** To find power and mass

```

1 //To find power and mass
2 clc
3 //Given:
4 d=38, t=32, s=100 //mm
5 E1=7 //N-m/mm^2 of sheared area
6 v=25 //m/s
7 //Solution:
8 //Power of the motor required:
9 //Calculating the sheared area
10 A=round(%pi*d*t) //mm^2
11 //Calculating the total energy required per hole
12 E1=E1*A //N-m
13 //Calculating the energy required for punching work
    per second
14 E=E1/10 //Energy required for punching work per
    second, N-m/s
15 //Calculating the power of the motor required
16 P=E/1000 //Power of the motor required , kW
17 //Mass of the flywheel required:

```

```

18 //Calculating the time required to punch a hole in a
    32 mm thick plate
19 t32=10/(2*s)*t //Time required to punch a hole in 32
    mm thick plate , seconds
20 //Calculating the energy supplied by the motor in
    t32 seconds
21 E2=E*t32 //N-m
22 //Calculating the energy to be supplied by the
    flywheel during punching
23 deltaE=E1-E2 //N-m
24 //Calculating the coefficient of fluctuation of
    speed
25 CS=3/100
26 //Calculating the mass of the flywheel required
27 m=round(deltaE/(v^2*CS)) //kg
28 //Results:
29 printf("\n\n Power of the motor required , P = %.3f
    kW.\n\n",P)
30 printf(" Mass of the flywheel required , m = %d kg.\n
    \n",m)

```

---

**Scilab code Exa 16.21** To find speed and number of rivets

```

1 //To find speed and number of rivets
2 clc
3 //Given:
4 P=3 //kW
5 m=150 //kg
6 k=0.6 //m
7 N1=300 //rpm
8 //Solution:
9 //Calculating the angular speed of the flywheel
    before riveting
10 omega1=2*pi*N1/60 //rad/s
11 //Speed of the flywheel immediately after riveting:

```

```

12 //Calculating the energy supplied by the motor
13 E2=P*1000 //N-m/s
14 //Calculating the energy absorbed during one
    riveting operation which takes 1 second
15 E1=10000 //N-m
16 //Calculating the energy to be supplied by the
    flywheel for each riveting operation per second
17 deltaE=E1-E2 //N-m
18 //Calculating the angular speed of the flywheel
    immediately after riveting
19 omega2=sqrt(omega1^2-(2*deltaE/(m*k^2))) //rad/s
20 //Calculating the corresponding speed in rpm
21 N2=omega2*60/(2*pi) //rpm
22 //Calculating the number of rivets that can be
    closed per minute
23 n=E2/E1*60 //Number of rivets that can be closed per
    minute
24 //Results:
25 printf("\n\n Speed of the flywheel immediately after
    riveting , N2 = %.1f rpm.\n\n",N2)
26 printf(" Number of rivets that can be closed per
    minute = %d rivets.\n\n",n)

```

---

**Scilab code Exa 16.22** To find mass of the flywheel

```

1 //To find mass of the flywheel
2 clc
3 //Given:
4 d=40, t=15 //mm
5 NoofHoles=30 //per minute
6 EnergyRequired=6 //N-m/mm^2
7 Time=1/10 //seconds
8 N1=160, N2=140 //rpm
9 k=1 //m
10 //Solution:

```



```

11 //Calculating the sheared area per hole
12 A=round(%pi*d*t) //Sheared area per hole , mm^2
13 //Calculating the energy required to punch a hole
14 E1=EnergyRequired*A //N-m
15 //Calculating the energy required for punching work
    per second
16 E=E1*NoofHoles/60 //Energy required for punching
    work per second , N-m/s
17 //Calculating the energy supplied by the motor
    during the time of punching
18 E2=E*Time //N-m
19 //Calculating the energy to be supplied by the
    flywheel during punching a hole
20 deltaE=E1-E2 //N-m
21 //Calculating the mean speed of the flywheel
22 N=(N1+N2)/2 //rpm
23 //Calculating the mass of the flywheel required
24 m=round(deltaE*900/(%pi^2*k^2*N*(N1-N2))) //kg
25 //Results:
26 printf("\n\n Mass of the flywheel required , m = %d
    kg.\n\n",m)

```

---

**Scilab code Exa 16.23** To find power and cross section

```

1 //To find power and cross section
2 clc
3 //Given:
4 n=25
5 d1=25/1000 , t1=18/1000 , D=1.4 , R=D/2 //m
6 touu=300*10^6 //N/m^2
7 etam=95/100 , CS=0.1
8 sigma=6*10^6 //N/m^2
9 rho=7250 //kg/m^3
10 //Solution:
11 //Power needed for the driving motor:

```

```

12 //Calculating the area of the plate sheared
13 AS=%pi*d1*t1 //m^2
14 //Calculating the maximum shearing force required
    for punching
15 FS=AS*touu //N
16 //Calculating the energy required per stroke
17 E=1/2*FS*t1 //Energy required per stroke , N-m
18 //Calculating the energy required per minute
19 E1=E*n //Energy required per minute, N-m
20 //Calculating the power required for the driving
    motor
21 P=E1/(60*etam)/1000 //Energy required for the
    driving motor , kW
22 //Dimensions for the rim cross-section:
23 //Calculating the maximum fluctuation of energy
24 deltaE=9/10*E //N-m
25 //Calculating the maximum fluctuation of energy
    provided by the rim
26 deltaErim=0.95*deltaE //N-m
27 //Calculating the mean speed of the flywheel
28 N=9*25 //rpm
29 //Calculating the mean angular speed
30 omega=2*pi*N/60 //rad/s
31 //Calculating the mass of the flywheel
32 m=round(deltaErim/(R^2*omega^2*CS)) //kg
33 //Calculating the thickness of rim
34 t=sqrt(m/(%pi*D*2*rho))*1000 //mm
35 //Calculating the width of rim
36 b=2*t //mm
37 //Results:
38 printf("\n\n Power needed for the driving motor = %
    .3 f kW.\n\n",P)
39 printf(" Thickness of the flywheel rim , t = %d mm.\n
    \n",t)
40 printf(" Width of the flywheel rim , b = %d mm.\n\n",
    b)

```

---