Scilab Textbook Companion for Engineering Physics by M.Arumugam[1]

Created by
Mohd Arif
B. Tech
Computer Engineering
Uttrakhand Technical Univeristy
College Teacher
None
Cross-Checked by
None

July 31, 2019

[1]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: Engineering Physics

Author: M.Arumugam

Publisher: Anuradha Publications

Edition: 2

Year: 2015

ISBN: 81-89638-02-5

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

Chapter 1

INTERFERENCE AND DIFFRACTION OF LIGHT

Scilab code Exa 1.1 `Fringe width`

```
//Example number 1.1, Page number 1.35

clc;clear;close


//Variable declaration
D=1                         //Distance in metre
lamda=589*10**-9            //nm to metres
d=2*10**-3                  //mm to metre

//Calculation
Beta=(D*lamda)/d   // in mm

//Result
printf("The fringe width beta=%0.4f mm",(Beta*10**3))
```

Scilab code Exa 1.2 Thickness of glass

```
//Example number 1.2, Page number 1.35
clc;clear;close

//Variable declaration
N=3                    //position
lamda=5450*10**-10         //Wawelength in Armstrong to
    metre
mu=1.5   // unitless

//Calculation
t=(N*lamda)/(mu-1)   // micron

//Result
printf("Thickness of glass plate=%0.2f micron",(t
   *10**6))
```

Scilab code Exa 1.3 No of lines

```
//Example number 1.3, Page number 1.36

clc;clear;close


//Variable declaration
w=0.02    // in m
n=1
lamda=6.56*10**-7 // in m
theta=(18+(14/60))*%pi/180 // in radian

//Calculation
N=(w*sin(theta))/(n*lamda) // no. of lines

//Result
```

```
printf("Total number of lines n the grating=%d",
   round(N))
//Answer varies due to rounding of number"
```

Scilab code Exa 1.4 Thickness of glass

```
//Example number 1.4, Page number 1.36

clc;clear;close


//Variable declaration
lamda=5893*10**-10              //Angstroms to mts
x=4*10**-2 // unitless
Beta=1*10**-3 // unitless

//Calculation
t=(lamda*x)/(2*Beta)

//Result
printf("t=%0.3f micron",(t*10**6))
```

Scilab code Exa 1.6 Thickness of coating

```
//Example number 1.6, Page number 1.36

clc;clear;close


//Variable declaration
lamda=5500  // Angstrom
nf=1.38 //unitless

```

```
//Calculation
t=lamda/(4*nf) // Angstrom

//Result
printf("The minimum thickness of coating,t=%0.1f
    Angstrom",t)
```

Scilab code Exa 1.7 Wavelength

```
//Example number 1.7, Page number 1.37

clc;clear;close


//Variable declaration
Beta=0.00227      //distance between adjascent green
    lines
D=2.5             // in m
d=0.0006          //distance between narrow slits

//Calculation
lamda=(Beta*d)/D // in m

//Result
printf("Wavelength,lamda=%.4e m",(lamda))
//Answer varies due to rounding of number"
```

Scilab code Exa 1.8 Thickness of plate

```
//Example number 1.8, Page number 1.37

clc;clear;close

```

```
//Variable declaration
lamda=5890*10**-10 // in m
mu=1.5 // unitless
theta=60*%pi/180      //Converting in to degrees

//Calculation

t=(lamda)/(2*mu*(cos(theta))) // in m

//Result
printf("Smallest thickness of plate,t=%0.4e m",t)
```

Scilab code Exa 1.9 Refractive index

```
//Example number 1.9, Page number 1.37

clc;clear;close


//Variable declaration
R=1// unitless
n=5// unitless
lamda=5.895*10**-7 // in m
dn=0.003 // in m

//Calculation
mu=(4*R*n*lamda)/(dn**2)

//Result
printf("Refractive index,mu = %0.2f",mu )
```

Scilab code Exa 1.10 Number of rulings

```
//Example number 1.10, Page number 1.38

clc;clear;close


//Variable declaration
lamda=5893; // in micron
n=3 // unitless
d_lamda=6 // in micron

//Calculation
N=(lamda)/(n*d_lamda)  // number of rulings

//Result
printf("N = %0.1f",N)
printf("\nThe number of rulings needed is 328. This
   is the minimum requirement.")
```

Scilab code Exa 1.11 Angular separation

```
//Example number 1.11, Page number 1.38

clc;clear;close


//Variable declaration
lamda=5.5*10**-7  // in m
d=2.54 // in m
x=1.22// unitless
//Calculation
dtheta=(x*lamda)/d // radian

//Result
printf("Smallest angular separation of two stars =
   %0.3e radian",dtheta)
```

Scilab code Exa 1.12 `Slit width`

```
//Example number 1.12, Page number 1.38

clc;clear;close


//Variable declaration
lamda=6500 // in Angstrom
theta=30*%pi/180 // radian

//Calculation
a=lamda/sin(theta) // Angstrom

//Result
printf("Slit width value, a= %0.f Angstroms",a)
printf("\nor a = %0.1f micron",(a*10^-4))
```

Scilab code Exa 1.13 `Ratio of the amplitudes`

```
//Example number 1.13, Page number 1.38

clc;clear;close


//Variable declaration
a2=1 // amplitude
a1=2*a2 // amplitude
//Calculation
r=a1/a2 // ratio

```

```
//Result
printf("r=%.f/1",r) //r = r/1 = r:1
printf("\nHence the ratio of the amplitudes= 2:1")
```

Scilab code Exa 1.14 Thickness of plate

```
//Example number 1.14, Page number 1.39

clc;clear;close


//Variable declaration
theta=5*10**-3/2// unitless
lamda=5*10**-7 // in m

//Calculation
a=(lamda)/theta // in m

printf("a=%0.e m",(a))
printf("\n a=%.1f mm",a*10**3)
```

Scilab code Exa 1.15 Refractive index

```
//Example number 1.15, Page number 1.39

clc;clear;close


//Variable declaration
N=20// unitless
lamda=5000*10**-10        //Angstroms to meters
t=2.5*10**-5 // in m

```

```
//Calculation
mu_1=(N*lamda)/t// unitless
mu=1+(mu_1)// unitless

//Result
printf("mu-1=%.1f",mu_1)
printf("\nRefractive index, mu=%1f",mu)
```

Scilab code Exa 1.16 No of orders

```
//Example number 1.16, Page number 1.39
clc;clear;close


//Variable declaration
theta=90*%pi/180     //theta=90 degrees to get
    maximum number of orders assume
lamda=5890*10**-10 // in m
aplusb=2*10**-6        //micro mts to mts

//Calculation
n=(aplusb*sin(theta))/lamda // order

//Result
printf("Maximum number of orders=%d",n)
```

Chapter 2

POLARIZATION AND ULTRASONIC

Scilab code Exa 2.1 `Value of theta`

```
//Example number 2.1, Page number 2.33


clc;clear;close

// Variable declaration
I=1/2 // unitless

// Calculation
theta1=acos(1/sqrt(2))*(180/%pi) // radian
theta2=acos(-1/sqrt(2))*(180/%pi) // radian
// Result
printf("theta=%.f degrees",theta1)
printf("\ntheta=%.f degrees",theta2)
printf("\n\n The value of theta can be +(or)- 45
    degrees and +(or)-135 degrees.")
```

Scilab code Exa 2.2 Value of ip

```
//Example number 2.2, Page number 2.33


clc;clear;close

// Calculation
ip=atan(1.732)*(180/%pi) // radian

// Result
printf("ip=%.f degrees",ip)
```

Scilab code Exa 2.3 Phase difference

```
//Example number 2.3, Page number 2.33


clc;clear;close

// Variable declaration
d=1*10**-3 // in m
lamda=6000*10**-10 // in m
nd=0.01                    // difference between the
   refractive indices(n1 - n2)

// Calculation
phi=(2*%pi*d*nd)/lamda // radian

// Result
printf("phi=%.1f radian",phi)
printf("\n\nSince the phase difference should be
   with in 2pi radius, we get phi=4.169 rad.")
```

Scilab code Exa 2.4 `Thickness of plate`

```
//Example number 2.4, Page number 2.33


clc;clear;close

// Variable declaration
lamda=5000*10**-10 // in m
mu_0=1.5533 // unitless
mu_1=1.5442// unitless

// Calculations
t=lamda/(2*(mu_0 - mu_1)) // in m

// Result
printf("Thickness,t=%0.2f micro m",(t*10**6))
```

Scilab code Exa 2.5 `Birefringence of the crystal`

```
//Example number 2.5, Page number 2.34


clc;clear;close

// Variable declaration
lamda=6000*10**-10 // in m
t=0.003*10**-2 // in m

// Calculations
delta_mu=lamda/(4*t) // unitless

```

```
// Result
printf("Birefringence of the crystal delta/mu=%0.3f"
   ,delta_mu)
```

Scilab code Exa 2.6 Refractive index

```
//Example number 2.6, Page number 2.34


clc;clear;close

// Variable declaration
theta=60*(%pi/180)        // When the angle of
   refraction is 30degrees, angle of reflection will
    be 60degrees

// Calculation
mu=tan(theta) // unitless

// Result
printf("Refractive index of medium=%0.3f",mu)
```

Scilab code Exa 2.7 Ultrasonic wavelength

```
//Example number 2.7, Page number 2.34


clc;clear;close

// Variable declaration
m=1 // unitless
lamda_l=6000*10**-10 // in m
theta=0.046*(%pi/180) // radian
```

```
n=2*10**6// unitless

// Calculation
lamda_s=(m*lamda_l)/(sin(theta)) // in m
v=n*lamda_s // in m/s

// Result
printf("Ultrasonic wavelength,lamda s =%0.2e m",(
    lamda_s))
printf("\nVelocity of ultrasonic waves in liquid =
    %0.f ms^-1",v)
// Answer varies due to rounding of numbers
```

Scilab code Exa 2.8 `Velocity of blood flow`

```
//Example number 2.8, Page number 2.35


clc;clear;close

// Variable declaration
C=1500 // in m
Df=267// unitless
f=2*10**6
theta=0*%pi/180          // degrees

// Calculation
V=(C*Df)/(2*f*cos(theta)) // in m/s

// Result
printf("Velocity of blood flow = %0.4f m-s^-1",V)
```

Scilab code Exa 2.9 `Fundamental frequency`

```
//Example number 2.9, Page number 2.35

clc;clear;close

// Variable declaration
t=0.7*10**-3 // in s
E=8.8*10**10 // V
rho=2800 // kg/m^3

// Calculation
f=(1/(2*t))*sqrt(E/rho)       // Fundamental
   frequency

// Result
printf("Fundamental frequency,f = %.e Hz",f)
```

Scilab code Exa 2.10 `Depth of the sea`

```
//Example number 2.10, Page number 2.35


clc;clear;close

// Variable declaration
v=1500 // in m/s
t=1.33 // in  s

// Calculation
d=(v*t)/2 // in m

// Result
printf("The depth of the sea = %.1f m",d)
```

Chapter 3

ACOUSTICS AND SUPERCONDUCTIVITY

Scilab code Exa 3.1 Reverbration time

```
//Example number 3.1, Page number 3.32

// importing modules
clc;clear;close

// Variable declaration
V=2265 // m^3
A=92.9 // Coefficient
x=2                  // The absorption become 2*A of
   open window

// Calculation
T=(0.16*V)/A       // Sabine's formula
T2=(0.16*V)/(x*A) // in s

// Result
printf("Reverbration time = %0.1f s",T)
printf("\nFinal Reverbration time = %0.2f s",T2)
printf("\nThus the reverbration time is reduced to
```

```
one-half of its initial value")
```

Scilab code Exa 3.2 Total absorption

```
//Example number 3.2 , Page number 3.32


clc;clear;close

// Variable declaration
a1=450     // Area of plastered wall
a2=360     // Area of wooden floor and wooden doors
a3=24      // Area of Glass
a4=600     // Area of seats
a5=500     // Area of audience when they are in seats
c1=0.03    // Coefficient of absorption of plastered
   wall
c2=0.06    // Coefficient of absorption of wooden
   floor and wooden doors
c3=0.025   // Coefficient of absorption of Glass
c4=0.3     // Coefficient of absorption of seats
c5=0.43    // Coefficient of absorption of audience
   when they are in seats
l=12 // in m
b=30 // in m
h=6 // in m

// Calculation
V=l*b*h          // volume of the hall
A=(a1*c1)+(a2*c2)+(a3*c3)+(a4*c4)+(a5*c5)  // Total
   absorption
T=(0.16*V)/A    // Reverbration time

// Result
printf("Volume of the hall = %.f m^3",V)
```

```
printf("\nTotal absorption = %0.1f m^2",A)
printf("\nReverbration time = %0.1f second",T)
// Answer given for the Reverbration time in the
    text book is wrong
```

Scilab code Exa 3.3 Total absorption

```
//Example number 3.3, Page number 3.33


clc;clear;close

// Variable declaration
T=1.2 // in s
V=7500 // in m^3

// Calculation
A=(0.16*V)/T // in m^2

// Result
printf("Total absorpttion = %.f m**2 of O.W.U.",A)
```

Scilab code Exa 3.4 Change in Reverbration time

```
//Example number 3.4, Page number 3.34

clc;clear;close

// Variable declaration
V=12*10**4 // in m^3
A=13200 // in m^2
x=2                    // The absorption become 2*A of
    open window
```

```
// Calculation
T1=(0.16*V)/A       // Sabine's formula
T2=(0.16*V)/(x*A)  // in s
Td=T1-T2 // in s

// Result
printf("T1 = %0.2f second",T1)
printf("\nT2 = %0.2f second",T2)
printf("\nChange in Reverbration time = %0.3f second
   ",Td)
```

Scilab code Exa 3.6 Critical field

```
//Example number 3.6, Page number 3.34


clc;clear;close

// Variable declaration
H0=64*10**3;     // initial field(ampere/m)
T=5;     // temperature(K)
Tc=7.26;    // transition temperature(K)

// Calculation
H=H0*(1-(T/Tc)**2);      // critical field(ampere/m)

// Result
printf("critical field is : %0.3e ampere/m",H)
```

Scilab code Exa 3.7 Frequency of generated microwaves

```
//Example number 3.7, Page number 3.34
```

```
clc;clear;close

// Variable declaration
e=1.6*10**-19 // eV
V=1*10 // in m^3
h=6.625*10**-34

// Calculations
v=(2*e*V**-3)/h // Hz

// Result
printf("Frequency of generated microwaves = %.2e Hz"
    ,v)
```

Scilab code Exa 3.8 Penetration depth

```
//Example number 3.8, Page number 3.34

clc;clear;close

// Variable declaration
d=7300                        // density in (kg/m**3)
N=6.02*10**26                 // Avagadro Number
A=118.7                       // Atomic Weight
E=1.9                       // Effective mass
e=1.6*10**-19

// Calculations
n=(d*N)/A // no. of electrons
m=E*9.1*10**-31 // in kg
x=4*%pi*10**-7*n*e**2 // in kg/m^2
lamda_L=sqrt(m/x) // in m

// Result
```

```
printf("Number of electrons per unit volume = %0.1e per m^3",n)
printf("\nEffective mass of electron ''m*'' = %0.2e kg",m)
printf("\nPenetration depth = %0.5f Angstroms",(lamda_L*10**8))
// The answer given in the text book is wrong
```

Scilab code Exa 3.9 Wavelength

```
//Example number 3.9, Page number 3.35


clc;clear;close

// Variable declaration
lamda_L1=39.6*10**-9 // in m
lamda_L2=173*10**-9 // in m
T1=7.1 // in s
T2=3 // in s

// Calculations
x=(lamda_L1/lamda_L2)**2 // in kg/m^2
Tc4=(T1**4)-((T2**4)*x)/(1-x) // in K
Tc=(Tc4)**(1/4) // in K
printf("Tc = %0.4f K",Tc)
printf("\nlamda0 = %.f nm",round((sqrt(1-(T2/Tc)**4)*lamda_L1)*10**9))
```

Scilab code Exa 3.10 Critical current density

```
//Example number 3.10, Page number 3.35

```

```
clc;clear;close

// Variable declaration
H0=6.5*10**4              // (ampere/metre)
T=4.2                     // K
Tc=7.18                   // K
r=0.5*10**-3

// Calculations
Hc=H0*(1-(T/Tc)**2) // unitless
Ic=(2*%pi*r)*Hc // A
A=%pi*r**2 // m^2
Jc=Ic/A                   // Critical current density

// Result
printf("Hc = %0.4e",Hc)
printf("\nCritical current density,Jc = %0.2e ampere
   /metre^2",Jc)
```

Scilab code Exa 3.11 Critical temperature

```
//Example number 3.11, Page number 6.36

clc;clear;close

// Variable declaration
Tc1=4.185 // K
M1=199.5// unitless
M2=203.4// unitless

// Calculations
Tc2=Tc1*(M1/M2)**(1/2) // in K

// Result
```

```
printf("New critical temperature for mercury = %0.3f
    K",Tc2)
```

Chapter 4

Lasers

Scilab code Exa 4.1 `Divergence`

```
//Example 4.1, Page number 4.32

clc;clear;close

// variable declaration
r1 = 2;                     // in radians
r2 = 3;                     // in radians
d1 = 4;                     // Converting from mm to radians
d2 = 6;                     // Converting from mm to radians

// calculations
D = (r2-r1)/(d2*10**3-d1*10**3)    // Divergence

// Result
printf("Divergence = %0.1e radian",D)
```

Scilab code Exa 4.2 `Relative Population`

```
//Example 4.2, Page number 4.32

clc;clear;close

// variable declaration
C=3*10**8                          // The speed of light
Lamda=6943                         // Wavelength
T=300                              // Temperature in
   Kelvin
h=6.626*10**-34                    // Planck constant
k=1.38*10**-23                     // Boltzmann's constant

// Calculations

V=(C)/(Lamda*10**-10)           // Frequency
R=exp(h*V/(k*T))        // Relative population

// Result
printf("Frequency (V) = %0.2e Hz",V)
printf("\nRelative Population = %.3e",R)
```

Scilab code Exa 4.3 Power density

```
//Example 4.3, Page number 4.32

clc;clear;close

// variable declaration
C=3*10**8                           // Velocity of light m
   /s
W=632.8*10**-9                      // wavelength in m
P=2.3
t=1
h=6.626*10**-34                    // Planck constant
S=1*10**-6
```

```

// Calculations
V=C/W                                   // Frequency
n=((P*10**-3)*t)/(h*V)                  // no.of photons
    emitted
PD=P*10**-3/S                           // Power density

// Result
printf("Frequency = %0.2e Hz",V)
printf("\nno.of photons emitted = %0.2e photons/sec"
    ,n)
printf("\nPower density = %0.1f kWm^-2",(PD/1000))
```

Scilab code Exa 4.4 Wavelength

```
//Example 4.4, Page number 4.33

clc;clear;close

// variable declaration
h=6.626*10**-34                         // Planck constant
C=3*10**8                               // Velocity of light
E_g=1.44                                // bandgap

// calculations
lamda=(h*C)*10**10/(E_g*1.6*10**-19)        //
    Wavelenght

// Result
printf("Wavelength = %.f Angstrom",(lamda))
```

Scilab code Exa 4.5 Bandgap

```
//Example 4.5, Page number 4.33

clc;clear;close

// variable declaration
W=1.55                              // wavelength

// Calculations
E_g=(1.24)/W                        // Bandgap in eV

// Result
printf("Band gap = %0.1f eV",E_g)
```

Chapter 5

FIBER OPTICS

Scilab code Exa 5.1 Acceptance angle

```
//Example 5.1, Page number 5.28

clc;clear;close

//variable declaration
n1=1.50          //Core refractive index
n2=1.47          //Cladding refractive index

//Calculations
C_a=asin(n2/n1)         //Critical angle
N_a=(n1**2-n2**2)**(1/2) // Numerical Aperture
A_a=asin(N_a) // degree

//Results
printf("The Critical angle = %0.1f degrees",(C_a
   *180/%pi))
printf("\nThe numerical aperture = %0.2f",N_a)
printf("\nThe acceptance angle = %0.1f degress",(A_a
   *180/%pi))
```

Scilab code Exa 5.2 No of propagated modes

```
//Example 5.2, Page number 5.28

clc;clear;close

//variable declaration
d=50                    //diameter
N_a=0.2                 //Numerical aperture
lamda=1                 //wavelength

//Calculations
N=4.9*(((d*10**-6*N_a)/(lamda*10**-6))**2)//
   unitless

//Result
printf("N = %.f ",N)
printf("\nFiber can support: %d guided modes",N)
printf("\nIn graded index fiber, No.of modes
   propagated inside the fiber = %.f only",(N/2))
```

Scilab code Exa 5.3 Numerical aperture

```
//Example 5.3, Page number 5.29

clc;clear;close

// variable declaration
d=50                    // diameter
n1=1.450// unitless
n2=1.447// unitless
lamda=1                 // wavelength
```

```

// Calculations
N_a=(n1**2-n2**2)    // Numerical aperture
N=4.9*(((d*10**-6*N_a)/(lamda*10**-6))**2)//
    Numerical aperture

// Results
printf("Numerical aperture = %.5f",N_a)
printf("\nNo. of modes that can be propogated = %.f"
    ,N)
```

Scilab code Exa 5.4 `Numerical aperture`

```
//Example 5.4, Page number 5.29


clc;clear;close

// variable declaration
delta=0.05            //unitless
n1=1.46//unitless

// Calculation
N_a=n1*(2*delta)**(1/2)      // Numerical aperture

// Result
printf("Numerical aperture = %.2f",N_a)
```

Scilab code Exa 5.5 `No of propagated modes`

```
//Example 5.5, Page number 5.29

clc;clear;close
```

```

// variable declaration
a=50 //unitless
n1=1.53 //unitless
n2=1.50 //unitless
lamda=1              // wavelength

// Calculations
N_a=(n1**2-n2**2)    // Numerical aperture
V=((2*%pi*a)/lamda)*N_a**(1/2) // V number

// Result
printf("V number = %.2f",V)
printf("\nmaximum no.of modes propagating through
   fiber = %.f",V)
```

Scilab code Exa 5.6 No of modes

```
//Example 5.6, Page number 5.29

clc;clear;close

// variable declaration
a=100//unitless
N_a=0.3              // Numerical aperture
lamda=850            // wavelength

// Calculations
V_n=(2*(%pi)**2*a**2*10**-12*N_a**2)/lamda
   **2*10**-18 // number of modes
// Result
printf("Number of modes = %d modes",round(V_n
   /10**-36))
printf("\nNo.of modes is doubled to account for the
   two possible polarisations")
```

```
printf("\nTotal No. of modes = %d",round(V_n/10**-36)*2)
```

Scilab code Exa 5.7 Cutoff Wavellength

```
//Example 5.7, Page number 5.29

clc;clear;close
// variable declaration
a=5;//unitless
n1=1.48;//unitless
delta=0.01;//unitless
V=25;// V number

// Calculation
lamda=(%pi*(a*10**-6)*n1*sqrt(2*delta))/V    //Cutoff Wavelength

// Result
printf("Cutoff Wavellength = %.3f micro-m",(lamda*10**7))
```

Scilab code Exa 5.8 Maximum core radius

```
//Example 5.8, Page number 5.30

clc;clear;close

// variable declaration
V=2.405//unitless
lamda=1.3 // in m
N_a=0.05//unitless

```

```
// Calculations
a_max=(V*lamda)/(2*%pi*N_a) // in m

// Result
printf("Maximum core radius = %.2f micro-m",(a_max))
```

Scilab code Exa 5.9 Acceptance angle

```
//Example 5.9, Page number 5.30

clc;clear;close

// variable declaration
N_a=0.3 // numerical aperture
Gamma=45 // coefficient

// Calculations
theta_a=asin(N_a) // degree
theta_as=asin((N_a)/cos(Gamma)) // degree

// Results
printf("Acceptance angle, theta_a = %.2f degrees",(
    theta_a*180/%pi))
printf("\nFor skew rays,theta_as = %.2f degrees",(
    theta_as*180/%pi))
// Answer given in the textbook is wrong
```

Scilab code Exa 5.10 Numerical aperture

```
//Example 5.10, Page number 5.30

clc;clear;close

```

```
// variable declaration
n1=1.53 //unitless
delta=0.0196//unitless

// Calculations
N_a=n1*(2*delta)**(1/2) // numerical aperture
A_a=asin(N_a)//degree
// Result
printf("Numerical aperture = %.3f",N_a)
printf("\nAcceptance angle = %.2f degrees",(A_a*180/
    %pi))
```

Scilab code Exa 5.11 Core radius

```
//Example 5.11, Page number 5.30

clc;clear;close

// variable declaration
n1=1.480 //unitless
n2=1.465 //unitless
V=2.405 //unitless
lamda=850*10**-9 // in m

// Calculations
delta=(n1**2-n2**2)/(2*n1**2) //unitless
a=(V*lamda*10**-9)/(2*%pi*n1*sqrt(2*delta)) // in m

// Results
printf("delta = %.2f",(delta))
printf("\nCore radius,a = %.2f micro-m",(a*10**15))
```

Scilab code Exa 5.12 Reflections per meter

```
//Example 5.12, Page number 5.31

clc;clear;close

// variable declaration
n1=1.5 //unitless
n2=1.49//unitless
a=25 // in m

// Calculations
C_a=asin(n2/n1)              // Critical angle
L=2*a*tan(C_a)        // in m
N_r=10**6/L           // reflections/m

// Result
printf("Critical angle = %.2f degrees",(C_a*180/%pi)
   )
printf("\nFiber length covered in one reflection = %
   .2f micro-m",(L))
printf("\nTotal no.of reflections per meter = %.f",(
   N_r))
printf("\nSince L=1m, Total dist. travelled by light
    over one metre of fiber = %.4f m",(1/sin(C_a)))
```

Scilab code Exa 5.13 No of modes

```
//Example 5.13, Page number 5.31

clc;clear;close

// variable declaration
alpha=1.85//unitless
lamda=1.3*10**-6 // in m
a=25*10**-6 // in m
N_a=0.21 // numerical aperture
```

```
// Calculations
V_n=((2*%pi**2)*a**2*N_a**2)/lamda**2 // V number
N_m=(alpha/(alpha+2))*V_n //unitless

printf("No.of modes = %.2f =155(approx)",N_m)
printf("\nTaking the two possible polarizations,
    Total No.of nodes = %.2f",(N_m*2))
```

Scilab code Exa 5.14 `Signal attenuation`

```
//Example 5.14, Page number 5.32

clc;clear;close

// variable declaration
P_i=100 // input
P_o=2 // output
L=10 // in km

// Calculations
S=(10/L)*log(P_i/P_o) // dB/km
O=S*L // dB

// Result
printf("a)Signal attention per unit length = %.1f dB
    -km^-1",S)
printf("\nb)Overall signal attenuation = %.f dB",O)
// Answer given in the textbook is wrong
```

Scilab code Exa 5.15 `Total dispersion`

```
//Example 5.15, Page number 5.32
```

```

clc;clear;close

// variable declaration
L=10 // in km
n1=1.55 //unitless
delta=0.026//unitless
C=3*10**5

// Calculations
delta_T=(L*n1*delta)/C //unitless
B_W=10/(2*delta_T) // Hz/km

// Result
printf("Total dispersion = %.1f ns",(delta_T/10**-9))
printf("\nBandwidth length product = %.2f Hz-km",(B_W/10**5))

// Answer given in the text book is wrong"
```

Chapter 6

MAGNETIC PROPERTIES AND CRYSTAL STRUCTURES

Scilab code Exa 6.1 Temperature rise

```
//Example number 6.1, Page number 6.46

clc;clear;close


// Variable declaration
El=10**-2*50;        // energy loss(J)
H=El*60;             // heat produced(J)
d=7.7*10**3;         // iron rod(kg/m**3)
s=0.462*10**-3;      // specific heat(J/kg K)

// Calculation
theta=H/(d*s);       // temperature rise(K)

// Result
printf("temperature rise is %.2f K",(theta))
```

Scilab code Exa 6.2 `Dipole moment`

```
//Example number 6.2, Page number 6.46

clc;clear;close


// Variable declaration
e=1.6*10**-19;     // charge(coulomb)
new=6.8*10**15;    // frequency(revolutions per second)
mew0=4*%pi*10**-7; // coefficient
R=5.1*10**-11;      // radius(m)

// Calculation
i=(e*new);            // current(ampere)
B=mew0*i/(2*R);               // magnetic field at the centre(weber/m**2)
A=%pi*R**2; // in m^2
d=i*A;                          // dipole moment(ampere/m**2)

// Result
printf("magnetic field at the centre is : %.f weber/m**2",B)
printf("\ndipole moment is : %.e Ampere/m**2",(d))
```

Scilab code Exa 6.3 `Flux density`

```
//Example number 6.3, Page number 6.46

clc;clear;close
```

```
// Variable declaration
chi=0.5*10**-5;     // magnetic susceptibility
H=10**6;      // field strength(ampere/m)
mew0=4*%pi*10**-7; // coefficient

// Calculation
I=chi*H;      // intensity of magnetisation(ampere/m)
B=mew0*(I+H);     // flux density in material(weber/m**2)

// Result
printf("intensity of magnetisation is : %.f Ampere/m",I)
printf("\nflux density in material is : %.3f weber/m^2",B)
```

Scilab code Exa 6.4 `Bohr magnetons`

```
//Example number 6.4, Page number 6.47

clc;clear;close


// Variable declaration
B=9.27*10**-24;       // bohr magneton(ampere m**2)
a=2.86*10**-10;       // edge(m)
Is=1.76*10**6;        // saturation value of magnetisation(ampere/m)

// Calculation
N=2/a**3;
mew_bar=Is/N;       // number of Bohr magnetons(ampere m**2)
```

```
mew_bar=mew_bar/B;       // number of Bohr magnetons(
   bohr magneon/atom)

// Result
printf("number of Bohr magnetons is : %.2f"+" bohr
   magneon/atom",(mew_bar))
```

Scilab code Exa 6.5 Magnetic moment

```
//Example number 6.5, Page number 6.47

clc;clear;close


// Variable declaration
mew0=4*%pi*10**-7; // coefficient
H=9.27*10**-24;       // bohr magneton(ampere m**2)
Beta=10**6;       // field(ampere/m)
k=1.38*10**-23;     // boltzmann constant
T=303;     // temperature(K)

// Calculation
mm=mew0*H*Beta/(k*T);     // average magnetic moment(
   bohr magneton/spin)

// Result
printf("average magnetic moment is: %.2e bohr
   magneton/spin",(mm))
```

Scilab code Exa 6.6 Hysteresis loss

```
//Example number 6.6, Page number 6.48

```

```
clc;clear;close


// Variable declaration
A=94;        // area(m**2)
vy=0.1;      // value of length(weber/m**2)
vx=20;       // value of unit length
n=50;        // number of magnetization cycles
d=7650;      // density(kg/m**3)

// Calculation
h=A*vy*vx;       // hysteresis loss per cycle(J/m**3)
hs=h*n;         // hysteresis loss per second(watt/m**3)
pl=hs/d;        // power loss(watt/kg)

// Result
printf("hysteresis loss per cycle is : %.f J/m^3",h)
printf("\nhysteresis loss per second is: %.f watt/m**3",hs)
printf("\npower loss is : %.2f watt/kg",(pl))
```

Scilab code Exa 6.7 Density of electron

```
//Example number 6.7, Page number 6.48

clc;clear;close

// variable declaration
d=2.351                     // bond length
N=6.02*10^26               // Avagadro number
n=8                         // number of atoms in unit cell
A=28.09                     // Atomin mass of silicon
m=6.02*10^26               // 1mole
```

```

// Calculations
a=(4*d)/sqrt(3) // in m
p=(n*A)/((a*10^-10)*m)     // density

// Result
printf("a=%.2f Angstorm",(a))
printf("\ndensity = %.2f kg/m^3",(p*10^16))
// Answer given in the textbook is wrong"
```

Scilab code Exa 6.8 Radius of largest sphere

```
//Example number 6.8, Page number 6.48

clc;clear;close

// Variable declaration
r=poly([0],'r')

// Calculation
a1=4*r/sqrt(3); // in m
R1=(a1/2)-r;              // radius of largest sphere
a2=4*r/sqrt(2); //in m
R2=(a2/2)-r;          // maximum radius of sphere


// Result
disp(R1,"radius of largest sphere is")
disp(R2,"maximum radius of sphere is")
```

Scilab code Exa 6.9 Volume and density change

```
//Example number 6.9, Page number 6.49
```

```
clc;clear;close

// variable declaration
r1=1.258                // Atomic radius of BCC
r2=1.292                // Atomic radius of FCC

// calculations
a1=(4*r1)/sqrt(3)           // in BCC
b1=((a1)^3)*10^-30              // Unit cell volume
v1=(b1)/2                           // Volume occupied by
    one atom
a2=2*sqrt(2)*r2             // in FCC
b2=(a2)^3*10^-30                // Unit cell volume
v2=(b2)/4                           // Volume occupied by
    one atom
v_c=((v1)-(v2))*100/(v1)        // Volume Change in %
d_c=((v1)-(v2))*100/(v2)        // Density Change in %

// Results
printf("a1=%.3f Angstrom" ,(a1))
printf("\nUnit cell volume = a1^3 = %.3e m^3",b1)
printf("\nVolume occupied by one atom = %.2e m^3",v1
    )
printf("\na2 = %.3f Angstrom",a2)
printf("\nUnit cell volume =a2^3 = %.3e m^3",b2)
printf("\nVolume occupied by one atom = %.2e m^3",v2
    )
printf("\nVolume Change in %% = %.3f",v_c)
printf("\nDensity Change in %% = %.2f",d_c)
printf("\nThus the increase of density or the
    decrease of volume is about 0.5%%")
```

Scilab code Exa 6.10 Spacing between ions

```
//Example number 6.10, Page number 6.50

clc;clear;close

// variable declaration
n=4      // unitless
M=58.5                        // Molecular wt. of NaCl
N=6.02*10^26                 // Avagadro number
rho=2180                      // density

// Calculations
a=((n*M)/(N*rho))^(1/3)    // in m
s=a/2 // in m

// Result
printf("a= %.3e m",a)
printf("\nspacing between the nearest neighbouring
   ions = %.4f nm",(s/10^-9))
```

Scilab code Exa 6.11 Lattice constant

```
//Example number 6.11, Page number 6.51

clc;clear;close

// variable declaration
n=4      // unitless
A=63.55                       // Atomic wt. of NaCl
N=6.02*10^26                 // Avagadro number
rho=8930                      // density

// Calculations
a=((n*A)/(N*rho))^(1/3)       // Lattice Constant

// Result
```

```
printf("lattice constant, a = %.2f nm",(a*10^9))
```

Scilab code Exa 6.12 `Density of iron`

```
//Example number 6.12, Page number 6.51

clc;clear;close

// variable declaration
r=0.123                        // Atomic radius
n=4
A=55.8                         // Atomic wt
a=2*sqrt(2)
N=6.02*10**26                 // Avagadro number

// Calculations
rho=(n*A)/((a*r*10**-9)**3*N) // kg/m^3

// Result
printf("Density of iron = %.f kg/m^-3",rho)
```

Chapter 7

CRYSTAL STRUCTURES AND XRAY DIFFRACTION

Scilab code Exa 7.1 Number of atoms

```

//Example number 7.1, Page number 7.12

clc;clear;close

// Variable declaration
R=poly([0],'R')
a=2*R // // unitless

// Results
disp(1/a^2,"i)Number of atoms per unit area of (100)
    plane=")
disp(1/sqrt(2)*a^2,"ii)Number of atoms per unit area
    of (110)plane=")
disp(1/sqrt(3)*a^2,"iii)Number of atoms per unit
    area of (111)plane=")
```

Scilab code Exa 7.2 Surface area

```
//Example number 7.2, Page number 7.13

clc;clear;close


// Variable declaration
a=3.61*10^-7 // in m
BC=sqrt(2)/2 //in m
AD=(sqrt(6))/2// in m
// Result
printf("i)Surface area of the face ABCD = %.e mm^2"
   ,(a^2))
printf("\nii)Surface area of plane (110) = %.2e
   atoms/mm^2",((2/(a*sqrt(2)*a))))
printf("\niii)Surface area of plane(111)= %.3e atoms
   /mm^2",(2/(BC*AD*a^2)))
```

Scilab code Exa 7.3 Find depth

```
//Example number 7.3, Page number 7.14

clc;clear;close

// Variable declaration

//dimensions in m
h1=1
k1=0
l1=0
h2=1
k2=1
l2=0
h3=1
```

```
k3=1
l3=1
a=1 // in m

// Calculations
d1=a/(sqrt(h1^2+k1^2+l1^2)) // in m
d2=a/(sqrt(h2^2+k2^2+l2^2)) // in m
d3=a/(sqrt(h3^2+k3^2+l3^2)) // in m

// Result
printf("d1 = %.1f m",d1)
printf("\nd2 = %.3f m",d2)
printf("\nd3 = %.3f m",d3)
printf("\n ratio d1:d2:d3 = %.f:%0.3f:%.3f",d1,d2,d3
  )
```

Scilab code Exa 7.4 Spacing

```
//Example number 7.4, Page number 7.15

clc;clear;close

// Variable declaration
h=2 // in m
k=2// in m
l=0// in m
a=450 // in m

// Calculations
d=a/(sqrt(h^2+k^2+l^2)) // in m

// Result
printf("d(220) = %.1f pm",d)
```

Scilab code Exa 7.5 Spacing

```
//Example number 7.5, Page number 7.15

clc;clear;close

// Variable declaration
a=3.615 // in m
r=1.278// in m
h=1// in m
k=1// in m
l=1// in m

// Calculations
a=(4*r)/sqrt(2)// in m
d=a/(sqrt(h^2+k^2+l^2))// in m

// Result
printf("a = %.3f Angstroms",a)
printf("\nd = %.3f Angstroms",d)
```

Scilab code Exa 7.7 Spacing

```
//Example number 7.7, Page number 7.15

clc;clear;close

// Variable declaration
n=1 // unitless
lamda=1.54// in m
theta=32*%pi/180 // radian
h=2// in m
```

```
k=2// in m
l=0// in m

// Calculations
d=(n*lamda*10^-10)/(2*sin(theta))    // derived from
    2dsin(theta)=n*l
a=d*(sqrt(h^2+k^2+l^2))//in m

// Results
printf("d = %.2e m",d)
printf("\na = %.1e m",a)
```

Scilab code Exa 7.8 Spacing

```
//Example number 7.8, Page number 7.16

clc;clear;close

// Variable declaration
lamda=0.58 // in m
theta1=6.45*%pi/180 // in radian
theta2=9.15*%pi/180 // in radian
theta3=13*%pi/180 // in radian

// Calculations
dbyn1=lamda/(2*(sin(theta1))) // in Angstrom
dbyn2=lamda/(2*sin(theta2))// in Angstrom
dbyn3=lamda/(2*sin(theta3))// in Angstrom

// Results
printf("i.  d/n = %.3f Angstroms ",dbyn1)
printf("\nii. d/n = %.3f Angstroms",(dbyn2))
printf("\niii.d/n = %.3f Angstroms",(dbyn3))
```

Scilab code Exa 7.9 Number of atoms

```
//Example number 7.9, Page number 7.16

clc;clear;close

// Variable declaration
d=1.18 // in m
theta=90*%pi/180 // in radian
lamda=1.540 // in m

// Calculations
n=(2*d*sin(theta))/lamda // unitless

// Result
printf("n = %0.2f",n)
```

Scilab code Exa 7.10 Spacing

```
//Example number 7.10, Page number 7.17

clc;clear;close

// Variable declaration
lamda=0.58 // in m
theta=9.5*%pi/180 // in radian
n=1 // unitless
d=0.5            // d200=a/sqrt(2^2+0^2+0^2)=0.5a
// Calculations
a=n*lamda/(2*d*sin(theta))      // 2*d*sin(theta)=n*
   lamda

```

```
// Result
printf("a = %.2f Angstorms",a)
```

Scilab code Exa 7.11 Aperture

```
//Example number 7.11, Page number 7.17

clc;clear;close

// Variable declaration
lamda=0.842 // in m
n1=1 // unitless
q=(8+(35/60))*(%pi/180) // unitless
n2=3 // unitless
d=1 // in m
// Calculations
// n*lamda=2*d*sin(theta)
// n1*0.842=2*d*sin(q)
// n3*0.842=2*d*sin(theta3)
// Dividing both the eauations, we get
// (n2*lamda)/(n1*lamda)=2*d*sin(theta3)/2*d*sin(q)
theta3=asin((((n2*lamda)/(n1*lamda))*(2*d*sin(q)))
    /(2*d)) // radian
d=theta3*180/%pi; // in m
a_d=int32(d); // // unitless
a_m=(d-int(d))*60 // // unitless

// Result
printf("sin(theta3) = %.f or %.3f ",a_d,a_m)
```

Scilab code Exa 7.12 Spacing

```
//Example number 7.12, Page number 7.18
```

```
clc;clear;close

// Variable declaration
a=3.16 // in m
lamda=1.54 // in m
n=1// unitless
theta=20.3*%pi/180 // radian

// Calculations
d=(n*lamda)/(2*sin(theta)) // in m
x=a/d                                          // let sqrt(h^2+k^2+l^2)=x

// Result
printf("d = %.2f Angstrom",d)
printf("\nsqrt(h^2+k^2+l^2) = %.3f ",x)
printf("\nTherefore, h^2+k^2+l^2 =sqrt(2)")
printf("\nh =1, k=1")
```

Scilab code Exa 7.13 Spacing

```
//Example number 7.13, Page number 7.18

clc;clear;close

// Variable declaration
n=4// unitless
A=107.87 // in m
rho=10500 // kg/m^3
N=6.02*10^26// unitless
h=1;// in m
k=1;// in m
l=1;// in m
H=6.625*10^-34 // planks constant
```

```
e=1.6*10^-19 // Charge
theta=(19+(12/60))*%pi/180 // radian
C=3*10^8 // in m/s
// Calculations
a=((n*A)/(rho*N))^(1/3)*10^10 // in m
d=a/sqrt(h^2+k^2+l^2) // in m
lamda=2*d*sin(theta)// in m
E=(H*C)/(lamda*10^-10*e) // eV

// Result
printf("a = %.2f Angstroms",a)
printf("\nd = %.2f Angstroms",d)
printf("\nlamda = %.3f Angstroms",lamda)
printf("\nE = %.e eV",E)
```

Scilab code Exa 7.14 Spacing

```
//Example number 7.14, Page number 7.19

clc;clear;close

// Variable declaration
a=4.57 // in m
h=1// in m
k=1// in m
l=1// in m
lamda=1.52 //in m
twotheta=33.5*%pi/180// radian
r=5                    // radius
// Calculations
d=a/(h^2+k^2+l^2)^(1/2)// in m
sintheta=lamda/(2*d)// // unitless
X=r/tan(twotheta)// in cm

// Result
```

```
printf("d = %.2f  Angstorms",d)
printf("\nsin(theta) = %.3f",sintheta)
printf("\nX = %.3f  cm",X)
```

Chapter 8

DEFECTS IN SOLIDS

Scilab code Exa 8.1 Number of vacancies

```
//Example number 8.1, Page number 8.16


clc;clear;close

// Variable declaration
N=6.023*10**26 // unitless
deltaHv=120 //unitless
B=1.38*10**-23 //unitless
k=6.023*10**23//unitless

// Calculations
n0=0                                               //  0 in denominator
n300=N*exp(-deltaHv*10**3/(k*B*300))      // The number of vacancies per kilomole of copper
n900=N*exp(-(deltaHv*10**3)/(k*B*900)) // The number of vacancies per kilomole of copper

// Results
printf("at 0K, The number of vacancies per kilomole
```

```
of copper is : %.f",n0)
printf("\nat 300K, The number of vacancies per kilomole of copper is : %.3e",(n300))
printf("\nat 900K, The numbber of vacancies per kilomole of copper is : %.3e",(n900))
```

Scilab code Exa 8.2 Fraction of vacancies

```
//Example number 8.2, Page number 8.17

clc;clear;close

// Variable declaration
F_500=1*10**-10 //unitless
k=poly([0],'k')
T1=500+273 // in K
T2=1000+273 // in K


// Calculations
lnx=log(F_500)*T1/T2; // vacancies
x=exp(lnx) //Fraction of vacancies

printf("Fraction of vacancies at 1000 degrees C = %.1e",x)
```

Scilab code Exa 8.3 Volume of unit cell

```
//Example number 8.3, Page number 8.17


clc;clear;close

```

```
// Variable declaration
a=(2*2.82*10^-10) // in m
delta_Hs=1.971*1.6*10^-19 // unitless
k=1.38*10^-23 // Constant
T=300 // in K

// Calculations
V=a^3                                    // Volume of unit
    cell of NaCl
N=4/V                                    // Total number of
    ion pairs
n=N*exp(-delta_Hs/(2*k*T))  //concentration in per m
    ^3

// Result
printf("Volume of unit cell of NaCl = %.3e m^3",V)
printf("\nTotal number of ion pairs ''N'' = %.2e",N)
printf("\nThe concentration of Schottky defects per
    m^3 at 300K = %.2e",n)
```

Scilab code Exa 8.4 Amount of climb

```
//Example number 8.4, Page number 8.18

clc;clear;close

// Variable declaration
N=6.023*10^23 // constant
delta_Hv=1.6*10^-19 //unitless
k=1.38*10^-23 //constant
T=500 // in K
mv=5.55;      // molar volume
x=2*10^-8;      // numbber of cm in 1 angstrom

// Calculations
```

```
n=N*exp(-delta_Hv/(k*T))/mv // in per cm^3
a=(n/(5*10^7*10^6))*x; // in cm

// Result
printf("The number that must be created on heating from 0 to 500K is n = %.2e per cm^3",n) // into cm^3
printf("\nAs one step is 2 Angstorms, 5*10^7 vacancies are required for 1cm")
printf("\nThe amount of climb down by the dislocation is : %.4f cm",a*10^8)
```