

Scilab Textbook Companion for
Engineering Physics
by M.Arumugam¹

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

August 27, 2016

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

Book Description

Title: 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	4
1 INTERFERENCE AND DIFFRACTION OF LIGHT	5
2 POLARIZATION AND ULTRASONIC	14
3 ACOUSTICS AND SUPERCONDUCTIVITY	20
4 Lasers	28
5 FIBER OPTICS	32
6 MAGNETIC PROPERTIES AND CRYSTAL STRUCTURES	42
7 CRYSTAL STRUCTURES AND XRAY DIFFRACTION	51
8 DEFECTS IN SOLIDS	61

List of Scilab Codes

Exa 1.1	Fringe width	5
Exa 1.2	Thickness of glass	6
Exa 1.3	No of lines	6
Exa 1.4	Thickness of glass	7
Exa 1.6	Thickness of coating	7
Exa 1.7	Wavelength	8
Exa 1.8	Thickness of plate	8
Exa 1.9	Refractive index	9
Exa 1.10	Number of rulings	9
Exa 1.11	Angular separation	10
Exa 1.12	Slit width	11
Exa 1.13	Ratio of the amplitudes	11
Exa 1.14	Thickness of plate	12
Exa 1.15	Refractive index	12
Exa 1.16	No of orders	13
Exa 2.1	Value of theta	14
Exa 2.2	Value of i_p	15
Exa 2.3	Phase difference	15
Exa 2.4	Thickness of plate	16
Exa 2.5	Birefringence of the crystal	16
Exa 2.6	Refractive index	17
Exa 2.7	Ultrasonic wavelength	17
Exa 2.8	Velocity of blood flow	18
Exa 2.9	Fundamental frequency	18
Exa 2.10	Depth of the sea	19
Exa 3.1	Reverbration time	20
Exa 3.2	Total absorption	21
Exa 3.3	Total absorption	22

Exa 3.4	Change in Reverbration time	22
Exa 3.6	Critical field	23
Exa 3.7	Frequency of generated microwaves	23
Exa 3.8	Penetration depth	24
Exa 3.9	Wavelength	25
Exa 3.10	Critical current density	25
Exa 3.11	Critical temperature	26
Exa 4.1	Divergence	28
Exa 4.2	Relative Population	28
Exa 4.3	Power density	29
Exa 4.4	Wavelength	30
Exa 4.5	Bandgap	30
Exa 5.1	Acceptance angle	32
Exa 5.2	No of propagated modes	33
Exa 5.3	Numerical aperture	33
Exa 5.4	Numerical aperture	34
Exa 5.5	No of propagated modes	34
Exa 5.6	No of modes	35
Exa 5.7	Cutoff Wavellength	36
Exa 5.8	Maximum core radius	36
Exa 5.9	Acceptance angle	37
Exa 5.10	Numerical aperture	37
Exa 5.11	Core radius	38
Exa 5.12	Reflections per meter	38
Exa 5.13	No of modes	39
Exa 5.14	Signal attenuation	40
Exa 5.15	Total dispersion	40
Exa 6.1	Temperature rise	42
Exa 6.2	Dipole moment	43
Exa 6.3	Flux density	43
Exa 6.4	Bohr magnetons	44
Exa 6.5	Magnetic moment	45
Exa 6.6	Hysteresis loss	45
Exa 6.7	Density of electron	46
Exa 6.8	Radius of largest sphere	47
Exa 6.9	Volume and density change	47
Exa 6.10	Spacing between ions	48
Exa 6.11	Lattice constant	49

Exa 6.12	Density of iron	50
Exa 7.1	Number of atoms	51
Exa 7.2	Surface area	52
Exa 7.3	Find depth	52
Exa 7.4	Spacing	53
Exa 7.5	Spacing	54
Exa 7.7	Spacing	54
Exa 7.8	Spacing	55
Exa 7.9	Number of atoms	56
Exa 7.10	Spacing	56
Exa 7.11	Aperture	57
Exa 7.12	Spacing	57
Exa 7.13	Spacing	58
Exa 7.14	Spacing	59
Exa 8.1	Number of vacancies	61
Exa 8.2	Fraction of vacancies	62
Exa 8.3	Volume of unit cell	62
Exa 8.4	Amount of climb	63

Chapter 1

INTERFERENCE AND DIFFRACTION OF LIGHT

Scilab code Exa 1.1 Fringe width

```
1 //Example number 1.1, Page number 1.35
2
3 clc;clear;close
4
5
6 //Variable declaration
7 D=1 //Distance in metre
8 lamda=589*10**-9 //nm to metres
9 d=2*10**-3 //mm to metre
10
11 //Calculation
12 Beta=(D*lamda)/d // in mm
13
14 //Result
15 printf("The fringe width beta=%0.4 f mm", (Beta*10**3)
    )
```

Scilab code Exa 1.2 Thickness of glass

```
1 //Example number 1.2, Page number 1.35
2 clc;clear;close
3
4 //Variable declaration
5 N=3 //position
6 lamda=5450*10**-10 //Wavelength in Armstrong to
   metre
7 mu=1.5 // unitless
8
9 //Calculation
10 t=(N*lamda)/(mu-1) // micron
11
12 //Result
13 printf("Thickness of glass plate=%0.2f micron", (t
   *10**6))
```

Scilab code Exa 1.3 No of lines

```
1 //Example number 1.3, Page number 1.36
2
3 clc;clear;close
4
5
6 //Variable declaration
7 w=0.02 // in m
8 n=1
9 lamda=6.56*10**-7 // in m
10 theta=(18+(14/60))*%pi/180 // in radian
11
12 //Calculation
13 N=(w*sin(theta))/(n*lamda) // no. of lines
14
15 //Result
```

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

Scilab code Exa 1.4 Thickness of glass

```
1 //Example number 1.4, Page number 1.36
2
3 clc;clear;close
4
5
6 //Variable declaration
7 lamda=5893*10**-10 //Angstroms to mts
8 x=4*10**-2 // unitless
9 Beta=1*10**-3 // unitless
10
11 //Calculation
12 t=(lamda*x)/(2*Beta)
13
14 //Result
15 printf("t=%0.3f micron" ,(t*10**6))
```

Scilab code Exa 1.6 Thickness of coating

```
1 //Example number 1.6, Page number 1.36
2
3 clc;clear;close
4
5
6 //Variable declaration
7 lamda=5500 // Angstrom
8 nf=1.38 //unitless
9
```

```

10 // Calculation
11 t=lamda/(4*nf) // Angstrom
12
13 // Result
14 printf("The minimum thickness of coating ,t=%0.1f
        Angstrom" ,t)

```

Scilab code Exa 1.7 Wavelength

```

1 //Example number 1.7, Page number 1.37
2
3 clc;clear;close
4
5
6 //Variable declaration
7 Beta=0.00227 //distance between adjacent green
  lines
8 D=2.5 // in m
9 d=0.0006 //distance between narrow slits
10
11 // Calculation
12 lamda=(Beta*d)/D // in m
13
14 // Result
15 printf("Wavelength ,lamda=%0.4 e m" ,(lamda))
16 //Answer varies due to rounding of number"

```

Scilab code Exa 1.8 Thickness of plate

```

1 //Example number 1.8, Page number 1.37
2
3 clc;clear;close
4

```

```

5
6 //Variable declaration
7 lamda=5890*10**-10 // in m
8 mu=1.5 // unitless
9 theta=60*%pi/180 //Converting in to degrees
10
11 //Calculation
12
13 t=(lamda)/(2*mu*(cos(theta))) // in m
14
15 //Result
16 printf("Smallest thickness of plate ,t=%0.4 e m",t)

```

Scilab code Exa 1.9 Refractive index

```

1 //Example number 1.9, Page number 1.37
2
3 clc;clear;close
4
5
6 //Variable declaration
7 R=1// unitless
8 n=5// unitless
9 lamda=5.895*10**-7 // in m
10 dn=0.003 // in m
11
12 //Calculation
13 mu=(4*R*n*lamda)/(dn**2)
14
15 //Result
16 printf("Refractive index ,mu = %0.2 f",mu )

```

Scilab code Exa 1.10 Number of rulings

```

1 //Example number 1.10, Page number 1.38
2
3 clc;clear;close
4
5
6 //Variable declaration
7 lamda=5893; // in micron
8 n=3 // unitless
9 d_lamda=6 // in micron
10
11 //Calculation
12 N=(lamda)/(n*d_lamda) // number of rulings
13
14 //Result
15 printf("N = %0.1 f",N)
16 printf("\nThe number of rulings needed is 328. This
    is the minimum requirement.")

```

Scilab code Exa 1.11 Angular separation

```

1 //Example number 1.11, Page number 1.38
2
3 clc;clear;close
4
5
6 //Variable declaration
7 lamda=5.5*10**-7 // in m
8 d=2.54 // in m
9 x=1.22// unitless
10 //Calculation
11 dtheta=(x*lamda)/d // radian
12
13 //Result
14 printf("Smallest angular separation of two stars =
    %0.3e radian",dtheta)

```

Scilab code Exa 1.12 Slit width

```
1 //Example number 1.12, Page number 1.38
2
3 clc;clear;close
4
5
6 //Variable declaration
7 lamda=6500 // in Angstrom
8 theta=30*%pi/180 // radian
9
10 //Calculation
11 a=lamda/sin(theta) // Angstrom
12
13 //Result
14 printf("Slit width value, a= %0.f Angstroms",a)
15 printf("\nor a = %0.1f micron", (a*10^-4))
```

Scilab code Exa 1.13 Ratio of the amplitudes

```
1 //Example number 1.13, Page number 1.38
2
3 clc;clear;close
4
5
6 //Variable declaration
7 a2=1 // amplitude
8 a1=2*a2 // amplitude
9 //Calculation
10 r=a1/a2 // ratio
11
```

```

12 //Result
13 printf("r=%0.f/1",r) //r = r/1 = r:1
14 printf("\nHence the ratio of the amplitudes= 2:1")

```

Scilab code Exa 1.14 Thickness of plate

```

1 //Example number 1.14, Page number 1.39
2
3 clc;clear;close
4
5
6 //Variable declaration
7 theta=5*10**-3/2// unitless
8 lamda=5*10**-7 // in m
9
10 //Calculation
11 a=(lamda)/theta // in m
12
13 printf("a=%0.e m", (a))
14 printf("\n a=%0.1 f mm", a*10**3)

```

Scilab code Exa 1.15 Refractive index

```

1 //Example number 1.15, Page number 1.39
2
3 clc;clear;close
4
5
6 //Variable declaration
7 N=20// unitless
8 lamda=5000*10**-10 //Angstroms to meters
9 t=2.5*10**-5 // in m
10

```

```

11 // Calculation
12 mu_1=(N*lamda)/t// unitless
13 mu=1+(mu_1)// unitless
14
15 // Result
16 printf("mu-1=%0.1 f" ,mu_1)
17 printf("\nRefractive index , mu=%1f" ,mu)

```

Scilab code Exa 1.16 No of orders

```

1 //Example number 1.16 , Page number 1.39
2 clc;clear;close
3
4
5 //Variable declaration
6 theta=90*%pi/180 //theta=90 degrees to get
   maximum number of orders assume
7 lamda=5890*10**-10 // in m
8 aplusb=2*10**-6 //micro mts to mts
9
10 // Calculation
11 n=(apusb*sin(theta))/lamda // order
12
13 // Result
14 printf("Maximum number of orders=%d" ,n)

```

Chapter 2

POLARIZATION AND ULTRASONIC

Scilab code Exa 2.1 Value of theta

```
1 //Example number 2.1, Page number 2.33
2
3
4 clc;clear;close
5
6 // Variable declaration
7 I=1/2 // unitless
8
9 // Calculation
10 theta1=acos(1/sqrt(2))*(180/%pi) // radian
11 theta2=acos(-1/sqrt(2))*(180/%pi) // radian
12 // Result
13 printf("theta=%0.f degrees",theta1)
14 printf("\ntheta=%0.f degrees",theta2)
15 printf("\n\n The value of theta can be +(or)- 45
    degrees and +(or)-135 degrees.")
```

Scilab code Exa 2.2 Value of ip

```
1 //Example number 2.2, Page number 2.33
2
3
4 clc;clear;close
5
6 // Calculation
7 ip=atan(1.732)*(180/%pi) // radian
8
9 // Result
10 printf("ip=%f degrees",ip)
```

Scilab code Exa 2.3 Phase difference

```
1 //Example number 2.3, Page number 2.33
2
3
4 clc;clear;close
5
6 // Variable declaration
7 d=1*10**-3 // in m
8 lamda=6000*10**-10 // in m
9 nd=0.01 // difference between the
   refractive indices(n1 - n2)
10
11 // Calculation
12 phi=(2*%pi*d*nd)/lamda // radian
13
14 // Result
15 printf("phi=%f radian",phi)
16 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

```
1 //Example number 2.4, Page number 2.33
2
3
4 clc;clear;close
5
6 // Variable declaration
7 lamda=5000*10**-10 // in m
8 mu_0=1.5533 // unitless
9 mu_1=1.5442// unitless
10
11 // Calculations
12 t=lamda/(2*(mu_0 - mu_1)) // in m
13
14 // Result
15 printf(" Thickness ,t=%0.2 f micro m" ,(t*10**6))
```

Scilab code Exa 2.5 Birefringence of the crystal

```
1 //Example number 2.5, Page number 2.34
2
3
4 clc;clear;close
5
6 // Variable declaration
7 lamda=6000*10**-10 // in m
8 t=0.003*10**-2 // in m
9
10 // Calculations
11 delta_mu=lamda/(4*t) // unitless
12
```

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

Scilab code Exa 2.6 Refractive index

```
1 //Example number 2.6, Page number 2.34
2
3
4 clc;clear;close
5
6 // Variable declaration
7 theta=60*(%pi/180) // When the angle of
    refraction is 30degrees, angle of reflection will
    be 60degrees
8
9 // Calculation
10 mu=tan(theta) // unitless
11
12 // Result
13 printf("Refractive index of medium=%0.3 f",mu)
```

Scilab code Exa 2.7 Ultrasonic wavelength

```
1 //Example number 2.7, Page number 2.34
2
3
4 clc;clear;close
5
6 // Variable declaration
7 m=1 // unitless
8 lamda_l=6000*10**-10 // in m
9 theta=0.046*(%pi/180) // radian
```

```

10 n=2*10**6 // unitless
11
12 // Calculation
13 lamda_s=(m*lamda_l)/(sin(theta)) // in m
14 v=n*lamda_s // in m/s
15
16 // Result
17 printf("Ultrasonic wavelength, lamda s =%0.2e m", (
    lamda_s))
18 printf("\nVelocity of ultrasonic waves in liquid =
    %0.f ms-1",v)
19 // Answer varies due to rounding of numbers

```

Scilab code Exa 2.8 Velocity of blood flow

```

1 //Example number 2.8, Page number 2.35
2
3
4 clc;clear;close
5
6 // Variable declaration
7 C=1500 // in m
8 Df=267 // unitless
9 f=2*10**6
10 theta=0*pi/180 // degrees
11
12 // Calculation
13 V=(C*Df)/(2*f*cos(theta)) // in m/s
14
15 // Result
16 printf("Velocity of blood flow = %0.4 f m-s-1",V)

```

Scilab code Exa 2.9 Fundamental frequency

```

1 //Example number 2.9, Page number 2.35
2
3 clc;clear;close
4
5 // Variable declaration
6 t=0.7*10**-3 // in s
7 E=8.8*10**10 // V
8 rho=2800 // kg/m^3
9
10 // Calculation
11 f=(1/(2*t))*sqrt(E/rho) // Fundamental
    frequency
12
13 // Result
14 printf("Fundamental frequency , f = %.e Hz",f)

```

Scilab code Exa 2.10 Depth of the sea

```

1 //Example number 2.10, Page number 2.35
2
3
4 clc;clear;close
5
6 // Variable declaration
7 v=1500 // in m/s
8 t=1.33 // in s
9
10 // Calculation
11 d=(v*t)/2 // in m
12
13 // Result
14 printf("The depth of the sea = %.1 f m",d)

```

Chapter 3

ACOUSTICS AND SUPERCONDUCTIVITY

Scilab code Exa 3.1 Reverbration time

```
1 //Example number 3.1, Page number 3.32
2
3 // importing modules
4 clc; clear; close
5
6 // Variable declaration
7 V=2265 // m^3
8 A=92.9 // Coefficient
9 x=2 // The absorption become 2*A of
   open window
10
11 // Calculation
12 T=(0.16*V)/A // Sabine's formula
13 T2=(0.16*V)/(x*A) // in s
14
15 // Result
16 printf("Reverbration time = %0.1f s",T)
17 printf("\\nFinal Reverbration time = %0.2f s",T2)
18 printf("\\nThus the reverbration time is reduced to
```

one-half of its initial value”)

Scilab code Exa 3.2 Total absorption

```
1 //Example number 3.2, Page number 3.32
2
3
4 clc; clear; close
5
6 // Variable declaration
7 a1=450 // Area of plastered wall
8 a2=360 // Area of wooden floor and wooden doors
9 a3=24 // Area of Glass
10 a4=600 // Area of seats
11 a5=500 // Area of audience when they are in seats
12 c1=0.03 // Coefficient of absorption of plastered
    wall
13 c2=0.06 // Coefficient of absorption of wooden
    floor and wooden doors
14 c3=0.025 // Coefficient of absorption of Glass
15 c4=0.3 // Coefficient of absorption of seats
16 c5=0.43 // Coefficient of absorption of audience
    when they are in seats
17 l=12 // in m
18 b=30 // in m
19 h=6 // in m
20
21 // Calculation
22 V=l*b*h // volume of the hall
23 A=(a1*c1)+(a2*c2)+(a3*c3)+(a4*c4)+(a5*c5) // Total
    absorption
24 T=(0.16*V)/A // Reverbration time
25
26 // Result
27 printf("Volume of the hall = %.f m^3",V)
```



```

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

```

Scilab code Exa 3.3 Total absorption

```

1 //Example number 3.3, Page number 3.33
2
3
4 clc;clear;close
5
6 // Variable declaration
7 T=1.2 // in s
8 V=7500 // in m^3
9
10 // Calculation
11 A=(0.16*V)/T // in m^2
12
13 // Result
14 printf("Total absorpttion = %.f m**2 of O.W.U.",A)

```

Scilab code Exa 3.4 Change in Reverbration time

```

1 //Example number 3.4, Page number 3.34
2
3 clc;clear;close
4
5 // Variable declaration
6 V=12*10**4 // in m^3
7 A=13200 // in m^2
8 x=2 // The absorption become 2*A of
    open window

```

```

9
10 // Calculation
11 T1=(0.16*V)/A // Sabine's formula
12 T2=(0.16*V)/(x*A) // in s
13 Td=T1-T2 // in s
14
15 // Result
16 printf("T1 = %0.2 f second",T1)
17 printf("\nT2 = %0.2 f second",T2)
18 printf("\nChange in Reverbration time = %0.3 f second
    ",Td)

```

Scilab code Exa 3.6 Critical field

```

1 //Example number 3.6, Page number 3.34
2
3
4 clc;clear;close
5
6 // Variable declaration
7 H0=64*10**3; // initial field(ampere/m)
8 T=5; // temperature(K)
9 Tc=7.26; // transition temperature(K)
10
11 // Calculation
12 H=H0*(1-(T/Tc)**2); // critical field(ampere/m)
13
14 // Result
15 printf("critical field is : %0.3e ampere/m",H)

```

Scilab code Exa 3.7 Frequency of generated microwaves

```

1 //Example number 3.7, Page number 3.34

```

```

2
3 clc;clear;close
4
5 // Variable declaration
6 e=1.6*10**-19 // eV
7 V=1*10 // in m^3
8 h=6.625*10**-34
9
10 // Calculations
11 v=(2*e*V**-3)/h // Hz
12
13 // Result
14 printf("Frequency of generated microwaves = %.2e Hz"
, v)

```

Scilab code Exa 3.8 Penetration depth

```

1 //Example number 3.8, Page number 3.34
2
3 clc;clear;close
4
5 // Variable declaration
6 d=7300 // density in (kg/m**3)
7 N=6.02*10**26 // Avagadro Number
8 A=118.7 // Atomic Weight
9 E=1.9 // Effective mass
10 e=1.6*10**-19
11
12 // Calculations
13 n=(d*N)/A // no. of electrons
14 m=E*9.1*10**-31 // in kg
15 x=4*%pi*10**-7*n*e**2 // in kg/m^2
16 lamda_L=sqrt(m/x) // in m
17
18 // Result

```

```

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

```

Scilab code Exa 3.9 Wavelength

```

1 //Example number 3.9, Page number 3.35
2
3
4 clc;clear;close
5
6 // Variable declaration
7 lamda_L1=39.6*10**-9 // in m
8 lamda_L2=173*10**-9 // in m
9 T1=7.1 // in s
10 T2=3 // in s
11
12 // Calculations
13 x=(lamda_L1/lamda_L2)**2 // in kg/m^2
14 Tc4=(T1**4)-((T2**4)*x)/(1-x) // in K
15 Tc=(Tc4)**(1/4) // in K
16 printf("Tc = %0.4f K",Tc)
17 printf("\nlamda0 = %.f nm",round((sqrt(1-(T2/Tc)**4)
    *lamda_L1)*10**9))

```

Scilab code Exa 3.10 Critical current density

```

1 //Example number 3.10, Page number 3.35
2

```

```

3
4 clc;clear;close
5
6 // Variable declaration
7 H0=6.5*10**4 // (ampere/metre)
8 T=4.2 // K
9 Tc=7.18 // K
10 r=0.5*10**-3
11
12 // Calculations
13 Hc=H0*(1-(T/Tc)**2) // unitless
14 Ic=(2*pi*r)*Hc // A
15 A=pi*r**2 // m^2
16 Jc=Ic/A // Critical current density
17
18 // Result
19 printf("Hc = %0.4e",Hc)
20 printf("\\nCritical current density ,Jc = %0.2e ampere
/metre^2",Jc)

```

Scilab code Exa 3.11 Critical temperature

```

1 //Example number 3.11, Page number 6.36
2
3 clc;clear;close
4
5 // Variable declaration
6 Tc1=4.185 // K
7 M1=199.5 // unitless
8 M2=203.4 // unitless
9
10 // Calculations
11 Tc2=Tc1*(M1/M2)**(1/2) // in K
12
13 // Result

```

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

Chapter 4

Lasers

Scilab code Exa 4.1 Divergence

```
1 //Example 4.1, Page number 4.32
2
3 clc;clear;close
4
5 // variable declaration
6 r1 = 2;           // in radians
7 r2 = 3;           // in radians
8 d1 = 4;           // Converting from mm to
   radians
9 d2 = 6;           // Converting from mm to
   radians
10
11 // calculations
12 D = (r2-r1)/(d2*10**3-d1*10**3) // Divergence
13
14 // Result
15 printf("Divergence = %0.1e radian",D)
```

Scilab code Exa 4.2 Relative Population

```

1 //Example 4.2, Page number 4.32
2
3 clc;clear;close
4
5 // variable declaration
6 C=3*10**8 // The speed of light
7 Lamda=6943 // Wavelength
8 T=300 // Temperature in
   Kelvin
9 h=6.626*10**-34 // Planck constant
10 k=1.38*10**-23 // Boltzmann's constant
11
12 // Calculations
13
14 V=(C)/(Lamda*10**-10) // Frequency
15 R=exp(h*V/(k*T)) // Relative population
16
17 // Result
18 printf("Frequency (V) = %0.2e Hz",V)
19 printf("\nRelative Population = %.3e",R)

```

Scilab code Exa 4.3 Power density

```

1 //Example 4.3, Page number 4.32
2
3 clc;clear;close
4
5 // variable declaration
6 C=3*10**8 // Velocity of light m
   /s
7 W=632.8*10**-9 // wavelength in m
8 P=2.3
9 t=1
10 h=6.626*10**-34 // Planck constant
11 S=1*10**-6

```



```

12
13 // Calculations
14 V=C/W // Frequency
15 n=((P*10**-3)*t)/(h*V) // no.of photons
    emitted
16 PD=P*10**-3/S // Power density
17
18 // Result
19 printf("Frequency = %0.2e Hz",V)
20 printf("\nno.of photons emitted = %0.2e photons/sec"
    ,n)
21 printf("\nPower density = %0.1 f kWm^-2" ,(PD/1000))

```

Scilab code Exa 4.4 Wavelength

```

1 //Example 4.4, Page number 4.33
2
3 clc;clear;close
4
5 // variable declaration
6 h=6.626*10**-34 // Planck constant
7 C=3*10**8 // Velocity of light
8 E_g=1.44 // bandgap
9
10 // calculations
11 lamda=(h*C)*10**10/(E_g*1.6*10**-19) //
    Wavelength
12
13 // Result
14 printf("Wavelength = %. f Angstrom" ,(lamda))

```

Scilab code Exa 4.5 Bandgap

```
1 //Example 4.5, Page number 4.33
2
3 clc;clear;close
4
5 // variable declaration
6 W=1.55 // wavelength
7
8 // Calculations
9 E_g=(1.24)/W // Bandgap in eV
10
11 // Result
12 printf("Band gap = %0.1 f eV",E_g)
```

Chapter 5

FIBER OPTICS

Scilab code Exa 5.1 Acceptance angle

```
1 //Example 5.1, Page number 5.28
2
3 clc;clear;close
4
5 //variable declaration
6 n1=1.50           //Core refractive index
7 n2=1.47           //Cladding refractive index
8
9 //Calculations
10 C_a=asin(n2/n1)   //Critical angle
11 N_a=(n1**2-n2**2)**(1/2) // Numerical Aperture
12 A_a=asin(N_a) // degree
13
14 //Results
15 printf("The Critical angle = %0.1f degrees", (C_a
    *180/%pi))
16 printf("\nThe numerical aperture = %0.2f", N_a)
17 printf("\nThe acceptance angle = %0.1f degree", (A_a
    *180/%pi))
```

Scilab code Exa 5.2 No of propagated modes

```
1 //Example 5.2, Page number 5.28
2
3 clc;clear;close
4
5 //variable declaration
6 d=50 //diameter
7 N_a=0.2 //Numerical aperture
8 lamda=1 //wavelength
9
10 //Calculations
11 N=4.9*(((d*10**-6*N_a)/(lamda*10**-6))**2) //
    unitless
12
13 //Result
14 printf("N = %.f ",N)
15 printf("\nFiber can support: %d guided modes",N)
16 printf("\nIn graded index fiber, No. of modes
    propagated inside the fiber = %.f only", (N/2))
```

Scilab code Exa 5.3 Numerical aperture

```
1 //Example 5.3, Page number 5.29
2
3 clc;clear;close
4
5 // variable declaration
6 d=50 // diameter
7 n1=1.450 // unitless
8 n2=1.447 // unitless
9 lamda=1 // wavelength
```

```

10
11 // Calculations
12 N_a=(n1**2-n2**2) // Numerical aperture
13 N=4.9*(((d*10**-6*N_a)/(lamda*10**-6))**2) //
    Numerical aperture
14
15 // Results
16 printf("Numerical aperture = %.5f",N_a)
17 printf("\nNo. of modes that can be propogated = %.f"
    ,N)

```

Scilab code Exa 5.4 Numerical aperture

```

1 //Example 5.4, Page number 5.29
2
3
4 clc;clear;close
5
6 // variable declaration
7 delta=0.05 //unitless
8 n1=1.46//unitless
9
10 // Calculation
11 N_a=n1*(2*delta)**(1/2) // Numerical aperture
12
13 // Result
14 printf("Numerical aperture = %.2f",N_a)

```

Scilab code Exa 5.5 No of propagated modes

```

1 //Example 5.5, Page number 5.29
2
3 clc;clear;close

```

```

4
5 // variable declaration
6 a=50 //unitless
7 n1=1.53 //unitless
8 n2=1.50 //unitless
9 lamda=1 // wavelength
10
11 // Calculations
12 N_a=(n1**2-n2**2) // Numerical aperture
13 V=((2*pi*a)/lamda)*N_a**(1/2) // V number
14
15 // Result
16 printf("V number = %.2f",V)
17 printf("\nmaximum no.of modes propagating through
    fiber = %.f",V)

```

Scilab code Exa 5.6 No of modes

```

1 //Example 5.6, Page number 5.29
2
3 clc;clear;close
4
5 // variable declaration
6 a=100//unitless
7 N_a=0.3 // Numerical aperture
8 lamda=850 // wavelength
9
10 // Calculations
11 V_n=(2*(%pi)**2*a**2*10**-12*N_a**2)/lamda
    **2*10**-18 // number of modes
12 // Result
13 printf("Number of modes = %d modes",round(V_n
    /10**-36))
14 printf("\nNo.of modes is doubled to account for the
    two possible polarisations")

```

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

Scilab code Exa 5.7 Cutoff Wavelength

```
1 //Example 5.7, Page number 5.29
2
3 clc;clear;close
4 // variable declaration
5 a=5;//unitless
6 n1=1.48;//unitless
7 delta=0.01;//unitless
8 V=25;// V number
9
10 // Calculation
11 lamda=(%pi*(a*10**-6)*n1*sqrt(2*delta))/V //
    Cutoff Wavelength
12
13 // Result
14 printf("Cutoff Wavelength = %.3f micro-m", (lamda
    *10**7))
```

Scilab code Exa 5.8 Maximum core radius

```
1 //Example 5.8, Page number 5.30
2
3 clc;clear;close
4
5 // variable declaration
6 V=2.405//unitless
7 lamda=1.3 // in m
8 N_a=0.05//unitless
9
```

```

10 // Calculations
11 a_max=(V*lamda)/(2*pi*N_a) // in m
12
13 // Result
14 printf("Maximum core radius = %.2f micro-m", (a_max))

```

Scilab code Exa 5.9 Acceptance angle

```

1 //Example 5.9, Page number 5.30
2
3 clc;clear;close
4
5 // variable declaration
6 N_a=0.3 // numerical aperture
7 Gamma=45 // coefficient
8
9 // Calculations
10 theta_a=asin(N_a) // degree
11 theta_as=asin((N_a)/cos(Gamma)) // degree
12
13 // Results
14 printf("Acceptance angle, theta_a = %.2f degrees", (
    theta_a*180/pi))
15 printf("\nFor skew rays, theta_as = %.2f degrees", (
    theta_as*180/pi))
16 // Answer given in the textbook is wrong

```

Scilab code Exa 5.10 Numerical aperture

```

1 //Example 5.10, Page number 5.30
2
3 clc;clear;close
4

```



```

5 // variable declaration
6 n1=1.53 //unitless
7 delta=0.0196//unitless
8
9 // Calculations
10 N_a=n1*(2*delta)**(1/2) // numerical aperture
11 A_a=asin(N_a)//degree
12 // Result
13 printf("Numerical aperture = %.3f",N_a)
14 printf("\nAcceptance angle = %.2f degrees", (A_a*180/
    %pi))

```

Scilab code Exa 5.11 Core radius

```

1 //Example 5.11, Page number 5.30
2
3 clc;clear;close
4
5 // variable declaration
6 n1=1.480 //unitless
7 n2=1.465 //unitless
8 V=2.405 //unitless
9 lamda=850*10**-9 // in m
10
11 // Calculations
12 delta=(n1**2-n2**2)/(2*n1**2) //unitless
13 a=(V*lamda*10**-9)/(2*%pi*n1*sqrt(2*delta)) // in m
14
15 // Results
16 printf("delta = %.2f", (delta))
17 printf("\nCore radius , a = %.2f micro-m", (a*10**15))

```

Scilab code Exa 5.12 Reflections per meter

```

1 //Example 5.12, Page number 5.31
2
3 clc;clear;close
4
5 // variable declaration
6 n1=1.5 //unitless
7 n2=1.49//unitless
8 a=25 // in m
9
10 // Calculations
11 C_a=asin(n2/n1) // Critical angle
12 L=2*a*tan(C_a) // in m
13 N_r=10**6/L // reflections/m
14
15 // Result
16 printf("Critical angle = %.2f degrees",(C_a*180/%pi)
17 )
18 printf("\nFiber length covered in one reflection = %
19 .2f micro-m",L)
20 printf("\nTotal no. of reflections per meter = %.f",
21 (N_r))
22 printf("\nSince L=1m, Total dist. travelled by light
23 over one metre of fiber = %.4f m", (1/sin(C_a)))

```

Scilab code Exa 5.13 No of modes

```

1 //Example 5.13, Page number 5.31
2
3 clc;clear;close
4
5 // variable declaration
6 alpha=1.85//unitless
7 lamda=1.3*10**-6 // in m
8 a=25*10**-6 // in m
9 N_a=0.21 // numerical aperture

```

```

10
11 // Calculations
12 V_n=((2*%pi**2)*a**2*N_a**2)/lamda**2 // V number
13 N_m=(alpha/(alpha+2))*V_n // unitless
14
15 printf("No. of modes = %.2f =155(approx)",N_m)
16 printf("\nTaking the two possible polarizations ,
    Total No. of nodes = %.2f", (N_m*2))

```

Scilab code Exa 5.14 Signal attenuation

```

1 //Example 5.14, Page number 5.32
2
3 clc;clear;close
4
5 // variable declaration
6 P_i=100 // input
7 P_o=2 // output
8 L=10 // in km
9
10 // Calculations
11 S=(10/L)*log(P_i/P_o) // dB/km
12 O=S*L // dB
13
14 // Result
15 printf("a) Signal attention per unit length = %.1f dB
    -km^-1",S)
16 printf("\nb) Overall signal attenuation = %.f dB",O)
17 // Answer given in the textbook is wrong

```

Scilab code Exa 5.15 Total dispersion

```

1 //Example 5.15, Page number 5.32

```

```

2
3 clc;clear;close
4
5 // variable declaration
6 L=10 // in km
7 n1=1.55 //unitless
8 delta=0.026//unitless
9 C=3*10**5
10
11 // Calculations
12 delta_T=(L*n1*delta)/C //unitless
13 B_W=10/(2*delta_T) // Hz/km
14
15 // Result
16 printf("Total dispersion = %.1f ns", (delta_T/10**-9)
    )
17 printf("Bandwidth length product = %.2f Hz-km", (
    B_W/10**5))
18
19 // Answer given in the text book is wrong"

```

Chapter 6

MAGNETIC PROPERTIES AND CRYSTAL STRUCTURES

Scilab code Exa 6.1 Temperature rise

```
1 //Example number 6.1, Page number 6.46
2
3 clc;clear;close
4
5
6 // Variable declaration
7 E1=10**-2*50; // energy loss(J)
8 H=E1*60; // heat produced(J)
9 d=7.7*10**3; // iron rod(kg/m**3)
10 s=0.462*10**-3; // specific heat(J/kg K)
11
12 // Calculation
13 theta=H/(d*s); // temperature rise(K)
14
15 // Result
16 printf("temperature rise is %.2f K", (theta))
```

Scilab code Exa 6.2 Dipole moment

```
1 //Example number 6.2, Page number 6.46
2
3 clc;clear;close
4
5
6 // Variable declaration
7 e=1.6*10**-19; // charge(coulomb)
8 new=6.8*10**15; // frequency(revolutions per
   second)
9 mew0=4*%pi*10**-7; // coefficient
10 R=5.1*10**-11; // radius(m)
11
12 // Calculation
13 i=(e*new); // current(ampere)
14 B=mew0*i/(2*R); // magnetic field at the
   centre(weber/m**2)
15 A=%pi*R**2; // in m^2
16 d=i*A; // dipole moment(ampere/m
   **2)
17
18 // Result
19 printf("magnetic field at the centre is : %.f weber/
   m**2",B)
20 printf("\ndipole moment is : %.e Ampere/m**2",d))
```

Scilab code Exa 6.3 Flux density

```
1 //Example number 6.3, Page number 6.46
2
3 clc;clear;close
```

```

4
5
6 // Variable declaration
7 chi=0.5*10**-5; // magnetic susceptibility
8 H=10**6; // field strength(ampere/m)
9 mew0=4*pi*10**-7; // coefficient
10
11 // Calculation
12 I=chi*H; // intensity of magnetisation(ampere/m)
13 B=mew0*(I+H); // flux density in material(weber/m
    **2)
14
15 // Result
16 printf("intensity of magnetisation is : %.f Ampere/m
    ",I)
17 printf("\nflux density in material is : %.3f weber/m
    ^2",B)

```

Scilab code Exa 6.4 Bohr magnetons

```

1 //Example number 6.4, Page number 6.47
2
3 clc;clear;close
4
5
6 // Variable declaration
7 B=9.27*10**-24; // bohr magneton(ampere m**2)
8 a=2.86*10**-10; // edge(m)
9 Is=1.76*10**6; // saturation value of
    magnetisation(ampere/m)
10
11 // Calculation
12 N=2/a**3;
13 mew_bar=Is/N; // number of Bohr magnetons(
    ampere m**2)

```

```

14 mew_bar=mew_bar/B;          // number of Bohr magnetons(
    bohr magneon/atom)
15
16 // Result
17 printf("number of Bohr magnetons is : %.2f"+" bohr
    magneon/atom" ,(mew_bar))

```

Scilab code Exa 6.5 Magnetic moment

```

1 //Example number 6.5, Page number 6.47
2
3 clc;clear;close
4
5
6 // Variable declaration
7 mew0=4*pi*10**-7; // coefficient
8 H=9.27*10**-24;   // bohr magneton(ampere m**2)
9 Beta=10**6;      // field(ampere/m)
10 k=1.38*10**-23;  // boltzmann constant
11 T=303;          // temperature(K)
12
13 // Calculation
14 mm=mew0*H*Beta/(k*T); // average magnetic moment(
    bohr magneton/spin)
15
16 // Result
17 printf("average magnetic moment is: %.2e bohr
    magneton/spin" ,(mm))

```

Scilab code Exa 6.6 Hysteresis loss

```

1 //Example number 6.6, Page number 6.48
2

```



```

3  clc;clear;close
4
5
6  // Variable declaration
7  A=94;           // area(m**2)
8  vy=0.1;        // value of length(weber/m**2)
9  vx=20;         // value of unit length
10 n=50;          // number of magnetization cycles
11 d=7650;        // density(kg/m**3)
12
13 // Calculation
14 h=A*vy*vx;     // hysteresis loss per cycle(J/m**3)
15 hs=h*n;        // hysteresis loss per second(watt/m
    **3)
16 pl=hs/d;      // power loss(watt/kg)
17
18 // Result
19 printf("hysteresis loss per cycle is : %.f J/m^3",h)
20 printf("\nhysteresis loss per second is: %.f watt/m
    **3",hs)
21 printf("\npower loss is : %.2f watt/kg",(pl))

```

Scilab code Exa 6.7 Density of electron

```

1  //Example number 6.7, Page number 6.48
2
3  clc;clear;close
4
5  // variable declaration
6  d=2.351           // bond length
7  N=6.02*10^26     // Avagadro number
8  n=8               // number of atoms in unit
    cell
9  A=28.09           // Atomin mass of silicon
10 m=6.02*10^26     // 1mole

```

```

11
12 // Calculations
13 a=(4*d)/sqrt(3) // in m
14 p=(n*A)/((a*10^-10)*m) // density
15
16 // Result
17 printf("a=%0.2f Angstorm", (a))
18 printf("\ndensity = %0.2f kg/m^3", (p*10^16))
19 // Answer given in the textbook is wrong"

```

Scilab code Exa 6.8 Radius of largest sphere

```

1 //Example number 6.8, Page number 6.48
2
3 clc;clear;close
4
5 // Variable declaration
6 r=poly([0], 'r')
7
8 // Calculation
9 a1=4*r/sqrt(3); // in m
10 R1=(a1/2)-r; // radius of largest sphere
11 a2=4*r/sqrt(2); //in m
12 R2=(a2/2)-r; // maximum radius of sphere
13
14
15 // Result
16 disp(R1,"radius of largest sphere is")
17 disp(R2,"maximum radius of sphere is")

```

Scilab code Exa 6.9 Volume and density change

```

1 //Example number 6.9, Page number 6.49

```

```

2
3 clc;clear;close
4
5 // variable declaration
6 r1=1.258 // Atomic radius of BCC
7 r2=1.292 // Atomic radius of FCC
8
9 // calculations
10 a1=(4*r1)/sqrt(3) // in BCC
11 b1=((a1)^3)*10^-30 // Unit cell volume
12 v1=(b1)/2 // Volume occupied by
    one atom
13 a2=2*sqrt(2)*r2 // in FCC
14 b2=(a2)^3*10^-30 // Unit cell volume
15 v2=(b2)/4 // Volume occupied by
    one atom
16 v_c=((v1)-(v2))*100/(v1) // Volume Change in %
17 d_c=((v1)-(v2))*100/(v2) // Density Change in %
18
19 // Results
20 printf("a1=%0.3f Angstrom" ,(a1))
21 printf("\nUnit cell volume = a1^3 = %0.3e m^3",b1)
22 printf("\nVolume occupied by one atom = %0.2e m^3",v1
    )
23 printf("\na2 = %0.3f Angstrom",a2)
24 printf("\nUnit cell volume =a2^3 = %0.3e m^3",b2)
25 printf("\nVolume occupied by one atom = %0.2e m^3",v2
    )
26 printf("\nVolume Change in %% = %0.3f",v_c)
27 printf("\nDensity Change in %% = %0.2f",d_c)
28 printf("\nThus the increase of density or the
    decrease of volume is about 0.5%%")

```

Scilab code Exa 6.10 Spacing between ions

```

1 //Example number 6.10, Page number 6.50
2
3 clc;clear;close
4
5 // variable declaration
6 n=4 // unitless
7 M=58.5 // Molecular wt. of NaCl
8 N=6.02*10^26 // Avagadro number
9 rho=2180 // density
10
11 // Calculations
12 a=((n*M)/(N*rho))^(1/3) // in m
13 s=a/2 // in m
14
15 // Result
16 printf("a= %.3e m",a)
17 printf("\nspacing between the nearest neighbouring
    ions = %.4f nm", (s/10^-9))

```

Scilab code Exa 6.11 Lattice constant

```

1 //Example number 6.11, Page number 6.51
2
3 clc;clear;close
4
5 // variable declaration
6 n=4 // unitless
7 A=63.55 // Atomic wt. of NaCl
8 N=6.02*10^26 // Avagadro number
9 rho=8930 // density
10
11 // Calculations
12 a=((n*A)/(N*rho))^(1/3) // Lattice Constant
13
14 // Result

```

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

Scilab code Exa 6.12 Density of iron

```
1 //Example number 6.12, Page number 6.51
2
3 clc;clear;close
4
5 // variable declaration
6 r=0.123 // Atomic radius
7 n=4
8 A=55.8 // Atomic wt
9 a=2*sqrt(2)
10 N=6.02*10**26 // Avagadro number
11
12 // Calculations
13 rho=(n*A)/((a*r*10**-9)**3*N) // kg/m^3
14
15 // Result
16 printf("Density of iron = %.f kg/m^-3",rho)
```

Chapter 7

CRYSTAL STRUCTURES AND XRAY DIFFRACTION

Scilab code Exa 7.1 Number of atoms

```
1
2 //Example number 7.1, Page number 7.12
3
4 clc;clear;close
5
6 // Variable declaration
7 R=poly([0], 'R')
8 a=2*R // // unitless
9
10 // Results
11 disp(1/a^2," i)Number of atoms per unit area of (100)
    plane=")
12 disp(1/sqrt(2)*a^2," ii)Number of atoms per unit area
    of (110)plane=")
13 disp(1/sqrt(3)*a^2," iii)Number of atoms per unit
    area of (111)plane=")
```

Scilab code Exa 7.2 Surface area

```
1 //Example number 7.2, Page number 7.13
2
3 clc;clear;close
4
5
6 // Variable declaration
7 a=3.61*10^-7 // in m
8 BC=sqrt(2)/2 //in m
9 AD=(sqrt(6))/2// in m
10 // Result
11 printf("i) Surface area of the face ABCD = %.e mm^2"
        ,(a^2))
12 printf("\nii) Surface area of plane (110) = %.2e
        atoms/mm^2",((2/(a*sqrt(2)*a))))
13 printf("\niii) Surface area of plane(111)= %.3e atoms
        /mm^2", (2/(BC*AD*a^2)))
```

Scilab code Exa 7.3 Find depth

```
1 //Example number 7.3, Page number 7.14
2
3 clc;clear;close
4
5 // Variable declaration
6
7 //dimensions in m
8 h1=1
9 k1=0
10 l1=0
11 h2=1
12 k2=1
13 l2=0
14 h3=1
```

```

15 k3=1
16 l3=1
17 a=1 // in m
18
19 // Calculations
20 d1=a/(sqrt(h1^2+k1^2+l1^2)) // in m
21 d2=a/(sqrt(h2^2+k2^2+l2^2)) // in m
22 d3=a/(sqrt(h3^2+k3^2+l3^2)) // in m
23
24 // Result
25 printf("d1 = %.1 f m",d1)
26 printf("\nd2 = %.3 f m",d2)
27 printf("\nd3 = %.3 f m",d3)
28 printf("\n ratio d1:d2:d3 = %.f:%0.3 f:%0.3 f",d1,d2,d3
    )

```

Scilab code Exa 7.4 Spacing

```

1 //Example number 7.4, Page number 7.15
2
3 clc;clear;close
4
5 // Variable declaration
6 h=2 // in m
7 k=2// in m
8 l=0// in m
9 a=450 // in m
10
11 // Calculations
12 d=a/(sqrt(h^2+k^2+l^2)) // in m
13
14 // Result
15 printf("d(220) = %.1 f pm",d)

```

Scilab code Exa 7.5 Spacing

```
1 //Example number 7.5, Page number 7.15
2
3 clc;clear;close
4
5 // Variable declaration
6 a=3.615 // in m
7 r=1.278// in m
8 h=1// in m
9 k=1// in m
10 l=1// in m
11
12 // Calculations
13 a=(4*r)/sqrt(2) // in m
14 d=a/(sqrt(h^2+k^2+l^2)) // in m
15
16 // Result
17 printf("a = %.3 f Angstroms",a)
18 printf("\\nd = %.3 f Angstroms",d)
```

Scilab code Exa 7.7 Spacing

```
1 //Example number 7.7, Page number 7.15
2
3 clc;clear;close
4
5 // Variable declaration
6 n=1 // unitless
7 lamda=1.54// in m
8 theta=32*%pi/180 // radian
9 h=2// in m
```

```

10 k=2// in m
11 l=0// in m
12
13 // Calculations
14 d=(n*lamda*10^-10)/(2*sin(theta)) // derived from
    2dsin(theta)=n*l
15 a=d*(sqrt(h^2+k^2+l^2))//in m
16
17 // Results
18 printf("d = %.2e m",d)
19 printf("\na = %.1e m",a)

```

Scilab code Exa 7.8 Spacing

```

1 //Example number 7.8, Page number 7.16
2
3 clc;clear;close
4
5 // Variable declaration
6 lamda=0.58 // in m
7 theta1=6.45*%pi/180 // in radian
8 theta2=9.15*%pi/180 // in radian
9 theta3=13*%pi/180 // in radian
10
11 // Calculations
12 dbyn1=lamda/(2*(sin(theta1))) // in Angstrom
13 dbyn2=lamda/(2*sin(theta2))// in Angstrom
14 dbyn3=lamda/(2*sin(theta3))// in Angstrom
15
16 // Results
17 printf("i. d/n = %.3f Angstroms ",dbyn1)
18 printf("\nii. d/n = %.3f Angstroms", (dbyn2))
19 printf("\niii. d/n = %.3f Angstroms", (dbyn3))

```

Scilab code Exa 7.9 Number of atoms

```
1 //Example number 7.9, Page number 7.16
2
3 clc;clear;close
4
5 // Variable declaration
6 d=1.18 // in m
7 theta=90*%pi/180 // in radian
8 lamda=1.540 // in m
9
10 // Calculations
11 n=(2*d*sin(theta))/lamda // unitless
12
13 // Result
14 printf("n = %0.2 f",n)
```

Scilab code Exa 7.10 Spacing

```
1 //Example number 7.10, Page number 7.17
2
3 clc;clear;close
4
5 // Variable declaration
6 lamda=0.58 // in m
7 theta=9.5*%pi/180 // in radian
8 n=1 // unitless
9 d=0.5 // d200=a/sqrt(2^2+0^2+0^2)=0.5a
10 // Calculations
11 a=n*lamda/(2*d*sin(theta)) // 2*d*sin(theta)=n*
    lamda
12
```

```
13 // Result
14 printf("a = %.2f Angstroms", a)
```

Scilab code Exa 7.11 Aperture

```
1 //Example number 7.11, Page number 7.17
2
3 clc;clear;close
4
5 // Variable declaration
6 lamda=0.842 // in m
7 n1=1 // unitless
8 q=(8+(35/60))*(%pi/180) // unitless
9 n2=3 // unitless
10 d=1 // in m
11 // Calculations
12 // n*lamda=2*d*sin(theta)
13 // n1*0.842=2*d*sin(q)
14 // n3*0.842=2*d*sin(theta3)
15 // Dividing both the equations, we get
16 // (n2*lamda)/(n1*lamda)=2*d*sin(theta3)/2*d*sin(q)
17 theta3=asin(((n2*lamda)/(n1*lamda))*(2*d*sin(q))
    /(2*d)) // radian
18 d=theta3*180/%pi; // in m
19 a_d=int32(d); // // unitless
20 a_m=(d-int(d))*60 // // unitless
21
22 // Result
23 printf("sin(theta3) = %.f or %.3f ", a_d, a_m)
```

Scilab code Exa 7.12 Spacing

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

```

2
3 clc;clear;close
4
5 // Variable declaration
6 a=3.16 // in m
7 lamda=1.54 // in m
8 n=1// unitless
9 theta=20.3*%pi/180 // radian
10
11 // Calculations
12 d=(n*lamda)/(2*sin(theta)) // in m
13 x=a/d // let sqrt(h^2+k
    ^2+l^2)=x
14
15 // Result
16 printf("d = %.2f Angstrom",d)
17 printf(" \nsqrt(h^2+k^2+l^2) = %.3f ",x)
18 printf(" \nTherefore , h^2+k^2+l^2 =sqrt(2)")
19 printf(" \nh =1, k=1")

```

Scilab code Exa 7.13 Spacing

```

1 //Example number 7.13, Page number 7.18
2
3 clc;clear;close
4
5 // Variable declaration
6 n=4// unitless
7 A=107.87 // in m
8 rho=10500 // kg/m^3
9 N=6.02*10^26// unitless
10 h=1; // in m
11 k=1; // in m
12 l=1; // in m
13 H=6.625*10^-34 // planks constant

```

```

14 e=1.6*10^-19 // Charge
15 theta=(19+(12/60))*%pi/180 // radian
16 C=3*10^8 // in m/s
17 // Calculations
18 a=((n*A)/(rho*N))^(1/3)*10^10 // in m
19 d=a/sqrt(h^2+k^2+l^2) // in m
20 lamda=2*d*sin(theta)// in m
21 E=(H*C)/(lamda*10^-10*e) // eV
22
23 // Result
24 printf("a = %.2f Angstroms",a)
25 printf("\nd = %.2f Angstroms",d)
26 printf("\nlamda = %.3f Angstroms",lamda)
27 printf("\nE = %.e eV",E)

```

Scilab code Exa 7.14 Spacing

```

1 //Example number 7.14, Page number 7.19
2
3 clc;clear;close
4
5 // Variable declaration
6 a=4.57 // in m
7 h=1// in m
8 k=1// in m
9 l=1// in m
10 lamda=1.52 //in m
11 twotheta=33.5*%pi/180// radian
12 r=5 // radius
13 // Calculations
14 d=a/(h^2+k^2+l^2)^(1/2) // in m
15 sintheta=lamda/(2*d)// // unitless
16 X=r/tan(twotheta)// in cm
17
18 // Result

```

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

Chapter 8

DEFECTS IN SOLIDS

Scilab code Exa 8.1 Number of vacancies

```
1 //Example number 8.1, Page number 8.16
2
3
4 clc;clear;close
5
6 // Variable declaration
7 N=6.023*10**26 // unitless
8 deltaHv=120 //unitless
9 B=1.38*10**-23 //unitless
10 k=6.023*10**23//unitless
11
12 // Calculations
13 n0=0 // 0
14     in denominator
15 n300=N*exp(-deltaHv*10**3/(k*B*300)) // The
16     number of vacancies per kilomole of copper
17 n900=N*exp(-(deltaHv*10**3)/(k*B*900)) // The number
18     of vacancies per kilomole of copper
19
20 // Results
21 printf("at 0K, The number of vacancies per kilomole
```



```

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

```

Scilab code Exa 8.2 Fraction of vacancies

```

1 //Example number 8.2, Page number 8.17
2
3 clc;clear;close
4
5 // Variable declaration
6 F_500=1*10** -10 //unitless
7 k=poly([0], 'k')
8 T1=500+273 // in K
9 T2=1000+273 // in K
10
11
12 // Calculations
13 lnx=log(F_500)*T1/T2; // vacancies
14 x=exp(lnx) //Fraction of vacancies
15
16 printf("Fraction of vacancies at 1000 degrees C = %
    .1e",x)

```

Scilab code Exa 8.3 Volume of unit cell

```

1 //Example number 8.3, Page number 8.17
2
3
4 clc;clear;close
5

```

```

6 // Variable declaration
7 a=(2*2.82*10^-10) // in m
8 delta_Hs=1.971*1.6*10^-19 // unitless
9 k=1.38*10^-23 // Constant
10 T=300 // in K
11
12 // Calculations
13 V=a^3 // Volume of unit
    cell of NaCl
14 N=4/V // Total number of
    ion pairs
15 n=N*exp(-delta_Hs/(2*k*T)) //concentration in per m
    ^3
16
17 // Result
18 printf("Volume of unit cell of NaCl = %.3e m^3",V)
19 printf("\nTotal number of ion pairs ''N'' = %.2e",N)
20 printf("\nThe concentration of Schottky defects per
    m^3 at 300K = %.2e",n)

```

Scilab code Exa 8.4 Amount of climb

```

1 //Example number 8.4, Page number 8.18
2
3 clc;clear;close
4
5 // Variable declaration
6 N=6.023*10^23 // constant
7 delta_Hv=1.6*10^-19 //unitless
8 k=1.38*10^-23 //constant
9 T=500 // in K
10 mv=5.55; // molar volume
11 x=2*10^-8; // number of cm in 1 angstrom
12
13 // Calculations

```

```
14 n=N*exp(-delta_Hv/(k*T))/mv // in per cm^3
15 a=(n/(5*10^7*10^6))*x; // in cm
16
17 // Result
18 printf("The number that must be created on heating
        from 0 to 500K is n = %.2e per cm^3",n) // into
        cm^3
19 printf("\nAs one step is 2 Angstroms, 5*10^7
        vacancies are required for 1cm")
20 printf("\nThe amount of climb down by the
        dislocation is : %.4f cm",a*10^8)
```
