Scilab Textbook Companion for Engineering Physics by V. Yadav[1]

Created by
Ujval Gupta
Modern Physics
Physics
Shri Mata Vaishno Devi University
College Teacher
Dr. Yugal Khajuria
Cross-Checked by
Lavitha Pereira

May 29, 2016

[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: V. Yadav

Publisher: Tata McGraw-Hill

Edition: 1

Year: 2009

ISBN: 978-0-07-008569-5

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

Quantum Mechanics

Scilab code Exa 1.1 Energy of the particle from de Broglie wavelength

```
// Scilab Code Ex1.1:Page-1.5 (2009)
clc; clear;
lambda = 2.1e-010;  // de Broglie wavelength of the
    particle, m
m = 1.67e-027;      // Mass of the particle, kg
h = 6.626e-034;     // Planck's constant, Js
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
// From de Broglie relation, lambda = h/sqrt(2*m*E),
     solving for E
E = h^2/(2*m*lambda^2*e);      // Energy of the
    particle, eV
printf("\nThe energy of the particle from de Broglie
     wavelength = %5.3e eV", E);

// Result
// The energy of the particle from de Broglie
    wavelength = 1.863e-002 eV
```

Scilab code Exa 1.2 de Broglie wavelength of the particle

```
// Scilab Code Ex1.2: Page-1.5 (2009)
clc; clear;
m = 1.67e-027;        // Mass of the particle, kg
h = 6.626e-034;       // Planck's constant, Js
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
E = 1e+011*e;     // Energy of the particle, J
lambda = h/sqrt(2*m*E);      // de Broglie wavelength of the particle, m
printf("\nThe de Broglie wavelength of the particle = %4.2e m", lambda);

// Result
// The de Broglie wavelength of the particle = 9.06e-017 m
```

Scilab code Exa 1.3 de Broglie wavelength of an accelerated electron

```
// Scilab Code Ex1.3: Page-1.5 (2009)
clc; clear;
V = 20e+03;      // Accelerating voltage of electron, V
lambda = 12.25/sqrt(V);      // de Broglie wavelength of the accelerated electron, m
printf("\nThe de Broglie wavelength of the electron = %6.4f angstrom", lambda);

// Result
// The de Broglie wavelength of the electron = 0.0866 angstrom
```

Scilab code Exa 1.4 Energy of the electron from de Broglie wavelength

```
// Scilab Code Ex1.4:  Page-1.6 (2009)
```

```
clc; clear;
lambda = 5.2e-03;   // de Broglie wavelength of the electron, m
m = 9.1e-031;       // Mass of the electron, kg
h = 6.626e-034;      // Planck's constant, Js
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
// From de Broglie relation, lambda = h/sqrt(2*m*E), solving for E
E = h^2/(2*m*lambda^2*e);      // Energy of the electron, eV
printf("\nThe energy of the electron from de Broglie wavelength = %5.3e eV", E);

// Result
// The energy of the electron from de Broglie wavelength = 5.576e-014 eV
```

Scilab code Exa 1.5 Velocity and de Broglie wavelength of a neutron

```
// Scilab Code Ex1.5: Page-1.6 (2009)
clc; clear;
m = 1.67e-027;       // Mass of the neutron, kg
h = 6.626e-034;      // Planck's constant, Js
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
E = 1e+04*e;    // Energy of the neutron, J
// As E = 1/2*m*v^2, solving for v
v = sqrt(2*E/m);    // Velocity of the neutron, m/s
lambda = h/(m*v);  // de Broglie wavelength of the neutron, m
printf("\nThe velocity of the neutron = %4.2e m/s", v);
printf("\nThe de Broglie wavelength of the neutron = %4.2e m", lambda);

// Result
```

```
// The velocity of the neutron = 1.38e+006 m/s
// The de Broglie wavelength of the neutron = 2.87e
    -013 m
```

Scilab code Exa 1.6 Wavelength of thermal neutron at room temperature

```
// Scilab Code Ex1.6: Page-1.6 (2009)
clc; clear;
m = 1.67e-027;        // Mass of the neutron, kg
k = 1.38e-023;        // Boltzmann constant, J/mol/K
T = 27+273;           // Room temperature, K
h = 6.626e-034;       // Planck's constant, Js
e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
v = sqrt(3*k*T/m);        // Velocity of the neutron,
   m/s
lambda = h/(m*v);   // de Broglie wavelength of the
    neutron, m
printf("\nThe de Broglie wavelength of the thermal
    neutrons = %4.2f angstrom", lambda/1e-010);

// Result
// The de Broglie wavelength of the thermal neutrons
    = 1.45 angstrom
```

Scilab code Exa 1.7 Angle of deviation for first order diffraction maxima

```
// Scilab Code Ex1.7: Page-1.6 (2009)
clc; clear;
m = 9.1e-031;        // Mass of the electron, kg
e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
h = 6.626e-034;       // Planck's constant, Js
E = 20e+03*e;       // Energy of the electron, J
// As 1/2*m*v^2 = E, solving for v
```

```
v = sqrt(2*E/m);          // Velocity of the electron , m/s
lambda = h/(m*v);   // de Broglie wavelength of the electron , m
n = 1;   // First order diffraction
d = 9.8e-011;         // Atomic spacing for thin gold foil , m
// Using Bragg's equation , 2*d*sin(theta) = n*lambda and solving for theta
theta = asind(n*lambda/(2*d));        // Angle of deviation for first order diffraction maxima, degree
printf("\nThe angle of deviation for first order diffraction maxima = %4.2f degrees", theta);

// Result
// The angle of deviation for first order diffraction maxima = 2.54 degrees
```

Scilab code Exa 1.8 de Broglie wavelength of a moving electron

```
// Scilab Code Ex1.8:  Page-1.7 (2009)
clc; clear;
m = 9.1e-031;        // Mass of the electron , kg
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
h = 6.626e-034;       // Planck's constant , Js
E = 5*e;       // Energy of the electron , J
// As 1/2*m*v^2 = E, solving for v
v = sqrt(2*E/m);          // Velocity of the electron , m/s
lambda = h/(m*v);   // de Broglie wavelength of the electron , m
printf("\nThe de Broglie wavelength of the electron = %3.1f angstrom", lambda/1e-010);

```

```
// Result
// The de Broglie wavelength of the electron = 5.5 angstrom
```

Scilab code Exa 1.9 de Broglie wavelength of a neutron of given kinetic energy

```
// Scilab Code Ex1.9: Page-1.7 (2009)
clc; clear;
m = 1.67e-027;       // Mass of the neutron, kg
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
h = 6.626e-034;      // Planck's constant, Js
E = 1*e;      // Energy of the electron, J
lambda = h/sqrt(2*m*E);   // de Broglie wavelength of the neutron, m
printf("\nThe de Broglie wavelength of the neutron = %4.2f angstrom", lambda/1e-010);

// Result
// The de Broglie wavelength of the neutron = 0.29 angstrom
```

Scilab code Exa 1.10 de Broglie wavelength associated with moving proton

```
// Scilab Code Ex1.10:  Page-1.8 (2009)
clc; clear;
m = 1.67e-027;       // Mass of the proton, kg
c = 3e+08;       // Speed of light, m/s
v = 1/20*c;          // Velocity of the proton, m/s
h = 6.626e-034;      // Planck's constant, Js
lambda = h/(m*v);   // de Broglie wavelength of the neutron, m
```

```
printf("\nThe de Broglie wavelength associated with moving proton = %5.3e m", lambda);

// Result
// The de Broglie wavelength associated with moving proton = 2.645e-14 m
```

Scilab code Exa 1.11 Wavelength of matter wave associated with moving proton

```
// Scilab Code Ex1.11: Page-1.8 (2009)
clc; clear;
m = 1.67e-027;       // Mass of the proton, kg
v = 2e+08;       // Velocity of the proton, m/s
h = 6.626e-034;      // Planck's constant, Js
lambda = h/(m*v);  // de Broglie wavelength of the neutron, m
printf("\nThe wavelength of matter wave associated with moving proton = %5.3e m", lambda);

// Result
// The wavelength of matter wave associated with moving proton = 1.984e-15 m
```

Scilab code Exa 1.12 de Broglie wavelength of an electron accelerated through a given potential

```
// Scilab Code Ex1.12: Page-1.17 (2009)
clc; clear;
m = 9.1e-031;       // Mass of the electron, kg
q = 1.6e-019;        // Charge on an electron, C
V = 50;      // Accelearting potential, V
E = q*V;     // Energy gained by the electron, J
```

```
h = 6.626e-034;      // Planck's constant, Js
lambda = h/sqrt(2*m*E);  // de Broglie wavelength of the electron, m
printf("\nThe de Broglie wavelength of the electron accelearted through a given potential = %5.3e m", lambda);

// Result
// The de Broglie wavelength of the electron accelearted through a given potential = 1.736e-10 m
```

Scilab code Exa 1.13 Interplanar spacing of the crystal

```
// Scilab Code Ex1.13: Page-1.17 (2009)
clc; clear;
theta = 45;     // Diffraction angle, degrees
h = 6.626e-034;     // Planck's constant
m = 1.67e-027;      // Mass of a neutron, kg
n = 1;     // Order of diffraction
k = 1.38e-023;   // Boltzmann constant, J/mol/K
T = 27+273;     // Absolute room temperature, K
E = 3/2*k*T;    // Energy of the neutron, J
lambda = h/sqrt(2*m*E); // de-Broglie wavelength of neutrons, m
// From Bragg's law, 2*d*sin(theta) = n*lambda, solving for d
d = n*lambda/(2*sind(theta));
printf("\nThe interplanar spacing of the crystal = %4.2f angstrom", d/1e-010);

// Result
// The interplanar spacing of the crystal = 1.03 angstrom
```

Scilab code Exa 1.14 Interplanar spacing using Bragg law

```
// Scilab Code Ex1.14: Page-1.18 (2009)
clc; clear;
theta = 70;     // Glancing angle at which reflection occurs, degrees
h = 6.626e-034;     // Planck's constant
m = 9.1e-031;      // Mass of a electron, kg
e = 1.6e-019;      // Electronic charge, C
V = 1000;   // Accelerating potential, V
n = 1;      // Order of diffraction
E = e*V;    // Energy of the electron, J
lambda = h/sqrt(2*m*E); // de-Broglie wavelength of electron, m
// From Bragg's law, 2*d*sin(theta) = n*lambda, solving for d
d = n*lambda/(2*sind(theta));   // Interplanar spacing, m
printf("\nThe interplanar spacing of the crystal = %6.4e m", d);

// Result
// The interplanar spacing of the crystal = 2.0660e-11 m
```

Scilab code Exa 1.15 de Broglie wavelength of electron accelerated at V volts

```
// Scilab Code Ex1.15: Page-1.18 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant
m = 9.1e-031;      // Mass of a electron, kg
```

```
e = 1.6e-019;         // Electronic charge, C
V = 1;    // For simplicity the accelerating potential is assumed to be unity, V
E = e*V;    // Energy of the electron, J
lambda = h/sqrt(2*m*E); // de-Broglie wavelength of electron, m
printf("\nde-Broglie wavelength of electron accelerated at V volts = %5.2f/sqrt(V) angstrom", lambda/1e-010);

// Result
// de-Broglie wavelength of electron accelerated at V volts = 12.23/sqrt(V) angstrom
```

Scilab code Exa 1.16 de Broglie wavelength of electron accelerated from rest

```
// Scilab Code Ex1.16: Page-1.18 (2009)
clc; clear;
h = 6.6e-034;       // Planck's constant
m = 9.1e-031;       // Mass of a electron, kg
e = 1.6e-019;        // Electronic charge, C
V = 100;    // Accelerating potential for electron, V
E = e*V;    // Energy of the electron, J
lambda = h/sqrt(2*m*E); // de-Broglie wavelength of electron, m
printf("\nde-Broglie wavelength of electron accelerated at %d volts = %6.4e m", V, lambda);

// Result
// de-Broglie wavelength of electron accelerated at 100 volts = 1.2231e-10 m
```

Scilab code Exa 1.17 The wavelength associated with moving mass

```
// Scilab Code Ex1.17:  Page-1.19 (2009)
clc; clear;
m = 10e-03;     // Mass of the body, kg
v = 110;    // Velocity of the mass, m/s
h = 6.6e-034;     // Planck's constant
lambda = h/(m*v); // de-Broglie wavelength of electron, m
printf("\nThe wavelength associated with mass moving with velocity %d m/s = %1.0e m", v, lambda);

// Result
// The wavelength associated with mass moving with velocity 110 m/s = 6e-34 m
```

Scilab code Exa 1.18 Wavelength of an electron from its kinetic energy

```
// Scilab Code Ex1.18:  Page-1.19 (2009)
clc; clear;
m = 9.1e-031;     // Mass of the electron, kg
Ek = 1.27e-017;     // Kinetic energy of electron, J
h = 6.6e-034;     // Planck's constant
lambda = h/sqrt(2*m*Ek); // de-Broglie wavelength of electron, m
printf("\nThe wavelength associated with moving electron = %4.2f angstrom", lambda/1e-010);

// Result
// The wavelength associated with moving electron = 1.37 angstrom
```

Scilab code Exa 1.19 Kinetic energy of electron

```
// Scilab Code Ex1.19: Page-1.19 (2009)
clc; clear;
m = 9.1e-031;      // Mass of the electron, kg
h = 6.6e-034;      // Planck's constant
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
lambda = 9.1e-012; // de-Broglie wavelength of electron, m
// We have lambda = h/(m*v), solving for v
v = h/(m*lambda);    // Velocity of the electron, m/s
K = 1/2*m*v^2;      // Kinetic energy of electron, J
printf("\nThe kinetic energy of electron having wavelength %3.1e m = %4.2e eV", lambda, K/e);

// Result
// The kinetic energy of electron having wavelength 9.1e-12 m = 1.81e+04 eV
```

Scilab code Exa 1.20 Speed of proton for an equivalent wavelength of that of electron

```
// Scilab Code Ex1.20: : Page-1.19 (2009)
clc; clear;
m_e = 9.1e-031;      // Mass of the electron, kg
m_p = 1.67e-027;     // Mass of the proton, kg
v_e = 1;     // For simplicity assume velocity of electron to be unity, m/s
// From de-Broglie relation,
// lambda_p = lambda_e = h(m*v_p), solving for v_p
v_p = m_e*v_e/m_p;  // Velocity of the proton, m/s
// As lambda_e = h/sqrt(2*m_e*K_e) and lambda_p = h/sqrt(2*m_p*K_p), solving for K_e/K_p
K_ratio = m_p/m_e;       // Ratio of kinetic energies of electron and proton

printf("\nThe speed of proton for an equivalent
```

```
    wavelength of that of electron = %3.1e ve", v_p);
printf("\nRatio of kinetic energies of electron and
    proton = %3.1e, therefore Ke > Kp", K_ratio);

// Result
// The speed of proton for an equivalent wavelength
    of that of electron = 5.4e-04 ve
// Ratio of kinetic energies of electron and proton
    = 1.8e+03, therefore Ke > Kp
```

Scilab code Exa 1.21 de Broglie wavelength of the electron

```
// Scilab Code Ex1.21: Page-1.20 (2009)
clc; clear;
V = 50; // Potential difference, V
m = 9.1e-031;      // Mass of the electron, kg
e = 1.6e-019;      // Electronic charge, C
h = 6.6e-034;     // Planck's constant, Js
lambda = h/sqrt(2*m*e*V);     // From de-Broglie
    relation,
printf("\nde-Broglie wavelength of the electron = %4
    .2f angstrom", lambda/1e-010);

// Result
// de-Broglie wavelength of the electron = 1.73
    angstrom
```

Scilab code Exa 1.23 Minimum accuracy to locate the position of an electron

```
// Scilab Code Ex1.23:: Page-1.31 (2009)
clc; clear;
v = 740;     // Speed of the electron, m/s
```

```
m = 9.1e-031;      // Mass of the electron, kg
h = 6.6e-034;     // Planck's constant, Js
p = m*v;     // Momentum of the electron, kg-m/s
frac_v = 0.05/100;   // Correctness in the speed
delta_p = p*frac_v;     // Uncertainty in momentum, kg-m/s
delta_x = h/(4*%pi)*1/delta_p;   // Uncertainty in position, m

printf("\nThe minimum accuracy to locate the position of an electron = %4.2e m",delta_x);

// Result
// The minimum accuracy to locate the position of an electron = 1.56e-04 m
```

Scilab code Exa 1.24 Uncertainty in energy of an emitted photon

```
// Scilab Code Ex1.24: : Page-1.31 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
h_cross = h/(2*%pi);     // Reduced Planck's constant, Js
delta_t = 1e-010;    // Uncertainty in time, s
// From Energy-time uncertainty,
// delta_E*delta_t = h_cross/2, solving for delta_E
delta_E = h_cross/(2*delta_t);   // Uncertainty in energy of an emitted photon, J

printf("\nThe uncertainty in energy of an emitted photon = %5.3e eV", delta_E/1.6e-019);

// Result
// The uncertainty in energy of an emitted photon = 3.283e-06 eV
```

Scilab code Exa 1.25 Minimum uncertainty in velocity of electron

```
// Scilab Code Ex1.25: : Page-1.31 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
delta_x_max = 1e-007;    // Uncertainty in length, m
m = 9.1e-031;    // Mass of an electron, kg
// From Position-momentum uncertainty,
// delta_p_min = m*delta_v_min = h/delta_x_max, solving for delta_v_min
delta_v_min = h/(delta_x_max*m);     // Minimum uncertainty in velocity of electron, m/s

printf("\nThe minimum uncertainty in velocity of electron = %4.2e m/s", delta_v_min);

// Result
// The minimum uncertainty in velocity of electron = 7.25e+03 m/s
```

Scilab code Exa 1.26 Minimum uncertainty in momentum and minimum kinetic energy of proton

```
// Scilab Code Ex1.26:  Page-1.32 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
delta_x_max = 8.5e-014;    // Uncertainty in length, m
m = 1.67e-027;  // Mass of proton, kg
// From Position-momentum uncertainty,
// delta_p_min*delta_x_max = h, solving for delta_p_min
```

```
delta_p_min = h/delta_x_max;      // Minimum uncertainty in momentum of electron, kg-m/s
p_min = delta_p_min;      // Minimum momentum of the proton, kg.m/s
delta_E = p_min^2/(2*m);

printf("\nThe minimum uncertainty in momemtum of proton = %4.2e kg-m/s", p_min);
printf("\nThe kinetic energy of proton = %6.3e eV", delta_E/1.6e-019);

// Result
// The minimum uncertainty in momemtum of proton = 7.76e-21 kg-m/s
// The kinetic energy of proton = 1.128e+05 eV
```

Scilab code Exa 1.27 Uncertainty in momentum of electron

```
// Scilab Code Ex1.27:: Page-1.32 (2009)
clc; clear;
h = 6.6e-034;      // Planck's constant, Js
e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
E = 0.15*1e+03*e;     // Energy of the electron, J
m = 9.1e-031;    // Mass of electron, kg
delta_x = 0.5e-010; // Position uncertainty of electron, m
p = (2*m*E)^(1/2);   // Momentum of the electron, kg-m/s
// delta_x*delta_p = h/(4*%pi), solving for delta_p
delta_p = h/(4*%pi*delta_x);      // Uncertainty in momentum of electron, kg-m/s
frac_p = delta_p/p*100;       // Percentage uncertainty in momentum of electron, kg-m/s

printf("\nThe percentage uncertainty in momentum of
```

```
electron = %2d percent", frac_p);

// Result
// The percentage uncertainty in momentum of electron = 15 percent
```

Scilab code Exa 1.28 Uncertainty in position of the particle

```
// Scilab Code Ex1.28:: Page-1.33 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
delta_v = 7.54e-015;     // Uncertainty in velocity of the particle, m/s
m = 0.25e-06;   // Mass of particle, kg
// delta_x*delta_p = h/(4*%pi), solving for delta_x
delta_x = h/(4*%pi*m*delta_v); // Position uncertainty of particle, m

printf("\nThe position uncertainty of particle = %4.2e m", delta_x);

// Result
// The position uncertainty of particle = 2.79e-14 m
```

Scilab code Exa 1.29 Uncertainty in position of the moving electron

```
// Scilab Code Ex1.29:: Page-1.33 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
v = 450;         // Velocity of the electron, m/s
```

```
delta_v = v*0.05/100;     // Uncertainty in velocity of the particle, m/s
m = 9.1e-031;  // Mass of electron, kg
// delta_x*delta_p = h/(4*%pi), solving for delta_x
delta_x = h/(4*%pi*m*delta_v); // Position uncertainty of particle, m

printf("\nThe position uncertainty of moving electron = %4.2e m", delta_x);

// Result
// The position uncertainty of moving electron = 2.57e-04 m
```

Scilab code Exa 1.30 Smallest possible uncertainty in position of the electron

```
// Scilab Code Ex1.30:: Page-1.33 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
c = 3e+08;       // Speed of light, m/s
v = 3e+07;         // Velocity of the electron, m/s
m0 = 9.1e-031;  // Rest mass of electron, kg
m = m0/sqrt(1-v^2/c^2); // Mass of moving electron, kg
delta_p_max = m*v;     // Maximum uncertainty in momentum of the particle, m/s
// delta_x_min*delta_p_max = h/(4*%pi), solving for delta_x_min
delta_x_min = h/(4*%pi*delta_p_max); // Minimum position uncertainty of particle, m

printf("\nThe smallest possible uncertainty in position of the electron = %5.3f angstrom",
```

```
    delta_x_min/1e-010);

// Result
// The smallest possible uncertainty in position of
    the electron = 0.019 angstrom
```

Scilab code Exa 1.31 Difference in the energy between the neighboring levels of Na at the highest state

```
// Scilab Code Ex1.31: : Page-1.44 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
m = 9.1e-031;    // Electronic mass, kg
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
l = 2e-002;      // Length of the side of the cube, m
E_F = 9*e;  // Fermi energy, J
// As E_F = h^2/(8*m*l^2)*(nx^2 + ny^2 + nz^2) and
    nx = ny = nz for a cube, solving for nx
nx = sqrt(E_F*(8*m*l^2)/(3*h^2));   // Value of
    integer for a cube
E = h^2/(8*m*l^2)*3*nx^2; // Fermi energy, J
E1 = h^2/(8*m*l^2)*((nx-1)^2 + nx^2 + nx^2);    //
    Energy of the level just below the fermi level, J
delta_E = E - E1;    // Difference in the energy
    between the neighbouring levels of Na at the
    highest state, J

printf("\nThe energy difference between the
    neighbouring levels of Na at the highest state =
    %4.2e eV", delta_E/e);

// Result
// The energy difference between the neighbouring
    levels of Na at the highest state = 1.06e-07 eV
```

Scilab code Exa 1.32 Energy of the neutron confined in a nucleus

```
// Scilab Code Ex1.32:: Page-1.45 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
m = 1.67e-027;    // Electronic mass, kg
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
nx = 1, ny = 1, nz = 1; // Principle quantum numbers
    in 3D corresponding to the longest energy state
lx = 1e-014, ly = 1e-014, lz = 1e-014;    //
   Dimensions of the box to which the neutron is
   confined, m
E = h^2/(8*m)*(nx^2/lx^2+ny^2/ly^2+nz^2/lz^2);    //
    Energy of the neutron confined in the nucleus, J

printf("\nThe energy of the neutron confined in a
   nucleus = %4.2e eV", E/e);

// Result
// The energy of the neutron confined in a nucleus =
    6.11e+06 eV
```

Scilab code Exa 1.33 Energy of an electron moving in one dimensional infinitely high potential box

```
// Scilab Code Ex1.33:: Page-1.46 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
m = 9.1e-031;    // Electronic mass, kg
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
n = 1;     // For simplicity assume principle
   quantum number to be unity
```

```
l = 2.1e-010;    // Length of one dimensional potential box, m
E = h^2*n^2/(8*m*l^2);    // Energy of the electron, J

printf("\nThe energy of the electron moving in one dimensional infinitely high potential box = %4.2f n^2 eV", E/e);

// Result
// The energy of the electron moving in one dimensional infinitely high potential box = 8.48 n^2 eV
```

Scilab code Exa 1.34 Lowest energy of an electron in a one dimensional force free region

```
// Scilab Code Ex1.34:: Page-1.46 (2009)
clc; clear;
h = 6.6e-034;    // Planck's constant, Js
m = 9.1e-031;    // Electronic mass, kg
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
n = 1;    // The lowest energy state of electron
l = 3.5e-010;    // Length of one dimensional potential box, m
E = h^2*n^2/(8*m*l^2);    // Energy of the electron in the lowest state, J

printf("\nThe lowest energy of the electron in a one dimensional force free region = %1d eV", E/e);

// Result
// The lowest energy of an electron in a one dimensional force free region = 3 eV
```

Scilab code Exa 1.35 First three energy levels of an electron in one dimensional box

```
// Scilab Code Ex1.35:: Page-1.46 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
m = 9.1e-031;    // Electronic mass, kg
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
l = 9.5e-010;    // Length of one dimensional potential box, m

// First energy level
n = 1;      // The first energy state of electron
E1 = h^2*n^2/(8*m*l^2);     // Energy of the electron in first state, J

// Second energy level
n = 2;      // The second energy state of electron
E2 = h^2*n^2/(8*m*l^2);     // Energy of the electron in second state, J

// Third energy level
n = 3;      // The third energy state of electron
E3 = h^2*n^2/(8*m*l^2);     // Energy of the electron in third state, J

printf("\nThe energy of the electron in first state = %4.1e J", E1);
printf("\nThe energy of the electron in second state = %4.1e J", E2);
printf("\nThe energy of the electron in third state = %4.1e J", E3);

// Result
```

```
// The energy of the electron in first state  = 6.6e-20 J
// The energy of the electron in second state  = 2.7e-19 J
// The energy of the electron in third state  = 6.0e-19 J
```

Scilab code Exa 1.36 Lowest two permitted energy values of the electron in a 1D box

```
// Scilab Code Ex1.36:: Page-1.47 (2009)
clc; clear;
h = 6.6e-034;    // Planck's constant, Js
m = 9.1e-031;   // Electronic mass, kg
e = 1.6e-019;   // Energy equivalent of 1 eV, J/eV
l = 2.5e-010;   // Length of one dimensional potential box, m

// First energy level
n = 1;      // The lowest energy state of electron
E1 = h^2*n^2/(8*m*l^2);    // Energy of the electron in first state, J

// Second energy level
n = 2;      // The second energy state of electron
E2 = h^2*n^2/(8*m*l^2);    // Energy of the electron in second state, J

printf("\nThe energy of the electron in lowest state = %5.2f eV", E1/e);
printf("\nThe energy of the electron in second state = %5.2f eV", E2/e);


// Result
```

```
// The energy of the electron in lowest state = 5.98 eV
// The energy of the electron in second state = 23.93 eV
```

Scilab code Exa 1.37 Lowest energy of the neutron confined to the nucleus

```
// Scilab Code Ex1.37:: Page-1.47 (2009)
clc; clear;
h = 6.6e-034;      // Planck's constant, Js
m = 1.67e-027;     // Electronic mass, kg
e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
l = 2.5e-010;     // Length of one dimensional potential box, m
delta_x = 1e-014;    // Uncertainty in position of neutron, m
// From uncertainty principle,
// delta_x*delta_p = h/(4*%pi), solving for delta_p
delta_p = h/(4*%pi*delta_x);     // Uncertainty in momentum of neutron, kg-m/s
p = delta_p;     // Momemtum of neutron in the box, kg-m/s
KE = p^2/(2*m); // Kinetic energy of neutron in the box, J

printf("\nThe lowest energy of the neutron confined to the nucleus = %4.2f MeV", KE/(e*1e+06));

// Result
// The lowest energy of the neutron confined to the nucleus = 0.05 MeV
```

Scilab code Exa 1.38 X ray scattering

```
// Scilab Code Ex1.38: : Page-1.56 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
m0 = 9.1e-031;    // Electronic mass, kg
c = 3e+08;        // Speed of light, m/s
e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
phi = 45;      // Scattering angle of X-rays, degrees
E = 75;      // Incident energy of X-rays, keV
// As from Compton shift formula
// 1/E_prime - 1/E = 1/(m0*c^2)*(1-cosd(phi))
// Solving for E_prime
E_prime = 1/((1/(m0*c^2/(e*1e+03)))*(1-cosd(phi))+1/
    E); // Energy of scattered photon, keV
E_recoil = E - E_prime;      // Energy of recoil
    electron, keV

printf("\nThe energy of scattered X-ray = %4.1f keV"
    , E_prime);
printf("\nThe energy of recoil electron = %3.1f keV"
    , E_recoil);

// Result
// The energy of scattered X-ray = 71.9 keV
// The energy of recoil electron = 3.1 keV
```

Scilab code Exa 1.39 Wavelength of scattered Xray

```
// Scilab Code Ex1.39: : Page-1.57 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
m0 = 9.1e-031;    // Electronic mass, kg
c = 3e+08;        // Speed of light, m/s
e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
```

```
phi = 60;       // Scattering angle of X-rays, degrees
E = 75;       // Incident energy of X-rays, keV
// As from Compton shift formula
delta_L = h/(m0*c)*(1-cosd(phi));    // Change in
    photon wavelength, m
lambda = 0.198e-010;     // Wavelength of incident
    photon, m
lambda_prime = (lambda+delta_L)/1e-010;   //
    Wavelength of scattered X-ray, angstrom

printf("\nThe wavelength of scattered X-ray = %6.4f
    angstrom", lambda_prime);

// Result
// The wavelength of scattered X-ray = 0.2101
    angstrom
```

Scilab code Exa 1.40 Wavelength of scattered radiation with changed angle of view

```
// Scilab Code Ex1.40:: Page-1.57 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
m0 = 9.1e-031;    // Electronic mass, kg
c = 3e+08;       // Speed of light, m/s
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
phi = 180;      // Scattering angle of X-rays,
    degrees
lambda = 1.78;     // Wavelength of incident photon,
    m
lambda_prime = 1.798;   // Wavelength of scattered X-
    ray, angstrom
// As from Compton shift formula
// lambda_prime - lambda = h/(m0*c)*(1-cosd(phi)),
    Change in photon wavelength, m
```

```
// Or we may write, lambda_prime - lambda = k*(1-cosd(phi))
// solving for k
k = (lambda_prime - lambda)/(1-cosd(phi));  // k = h/(m0*c) value, angstrom

// For phi = 60
phi = 60;         // New angle of scattering, degrees
lambda_prime = lambda + k*(1-cosd(phi)); // Wavelength of scattered radiation at 60 degree angle, angstrom
printf("\nThe wavelength of scattered X-ray at %d degrees view = %6.4f angstrom", phi, lambda_prime);
// Recoil energy of electron
E = h*c*(1/lambda - 1/lambda_prime)*1e+010;    // Recoil energy of electron, joule
printf("\nThe recoil energy of electron scattered through %d degrees = %4.1f eV", phi, E/e);

// Result
// The wavelength of scattered X-ray at 60 degrees view = 1.7845 angstrom
// The recoil energy of electron scattered through 60 degrees = 17.5 eV
```

Scilab code Exa 1.41 Compton scattering through aluminium foil

```
// Scilab Code Ex1.41:: Page-1.58 (2009)
clc; clear;
h = 6.6e-034;     // Planck's constant, Js
m0 = 9.1e-031;    // Electronic mass, kg
c = 3e+08;        // Speed of light, m/s
e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
phi = 90;      // Scattering angle of X-rays, degrees
```

```
E = 510*1e+03*e;    // Energy of incident photon, J
// As E = h*c/lambda, solving for lambda
lambda = h*c/E;    // Wavelength of incident photon, m
// As from Compton shift formula
// lambda_prime - lambda = h/(m0*c)*(1-cosd(phi)), solving for lambda_prime
lambda_prime = lambda + h/(m0*c)*(1-cosd(phi));  // Wavelength of scattered X-ray, m
printf("\nThe wavelength of scattered X-ray as viewed at %d degrees = %4.2e m", phi, lambda_prime);

// Recoil energy of electron
E = h*c*(1/lambda - 1/lambda_prime);    // Recoil energy of electron, joule
printf("\nThe recoil energy of electron scattered through %d degrees = %4.2e eV", phi, E/e);

// Direction of recoil electron
theta = atand(lambda*sind(phi)/(lambda_prime-lambda*cosd(phi)));  // Direction of recoil electron, degrees
printf("\nThe direction of emission of recoil electron = %5.2f degrees", theta);


// Result
// The wavelength of scattered X-ray as viewed at 90 degrees = 4.84e-12 m
// The recoil energy of electron scattered through 90 degrees = 2.55e+05 eV
// The direction of emission of recoil electron = 26.61 degrees
```

Scilab code Exa 1.42 Energetic electrons in the Xray tube

```
// Scilab Code Ex1.42: : Page-1.59 (2009)
clc; clear;
m = 9.1e-031;    // Electronic mass, kg
c = 3e+08;       // Speed of light, m/s
e = 1.6e-019;    // Charge on the electron, C
V = 12.4e+03;    // Potential diffeence applied across the X-ray tube, V
i = 2e-03;   // Current through the X-ray tube, A
t = 1;   // Time for which the electrons strike the target material, s
N = i*t/e;     // Number of electrons striking the target per sec, per sec
v_max = sqrt(2*e*V/m);   // Maximum speed of the electrons, m/s

printf("\nThe number of electrons striking the target per sec = %4.2e electrons/sec", N);
printf("\nThe maximum speed of the electrons when they strike = %3.1e m/s", v_max);


// Result
// The number of electrons striking the target per sec = 1.25e+16 electrons/sec
// The maximum speed of the electrons when they strike = 6.6e+07 m/s
```

Chapter 2

Interference

Scilab code Exa 2.1 Slit separation in Double Slit experiment

```
// Scilab Code Ex2.1:: Page-2.9 (2009)
clc; clear;
lambda = 5893e-008;  // Wavelength of light used, m
D = 200;    // Distance of the source from the screen, m
b = 0.2;    // Fringe separation, cm
d = lambda*D/b; // Separation between the slits, cm

printf("\nThe separation between the slits = %3.1e cm", d);

// Result
// The separation between the slits = 5.9e-002 cm
```

Scilab code Exa 2.2 Wavelength of light in Young Double Slit experiment

```
// Scilab Code Ex2.2:: Page-2.10 (2009)
clc; clear;
```

```
d = 0.2; // Separation between the slits, cm
D = 100;    // Distance of the source from the
    screen, m
b = 0.35e-01;     // Fringe separation, cm
lambda = b*d/D;  // Wavelength of light used, m
printf("\nThe wavelength of the light = %3.1e cm",
    lambda);

// Result
// The wavelength of the light = 7.0e-005 cm
```

Scilab code Exa 2.3 Ratio of maximum intensity to minimum intensity of interference fringes

```
// Scilab Code Ex2.3:: Page-2.10 (2009)
clc; clear;
I2 = 1;      // For simplicity assume intensity from
    slit 2 to be unity, W/sq-m
I1 = I2*25; // Intensity from slit 1, W/sq-m
I_ratio = I1/I2;    // Intensity ratio
a_ratio = sqrt(I_ratio);    // Amplitude ratio
a2 = 1;      // For simplicity assume amplitude from
    slit 2 to be unity, m
a1 = a_ratio*a2;    // Amplitude from slit 1, m
I_max = (a1 + a2)^2;   // Maximum intensity of wave
    during interference, W/sq-m
I_min = (a1 - a2)^2;   // Minimum intensity of wave
    during interference, W/sq-m
cf = 4; // Common factor
printf("\nThe ratio of maximum intentisy to minimum
    intensity of interference fringes = %d/%d", I_max
    /cf, I_min/cf);

// Result
// The ratio of maximum intentisy to minimum
```

```
intensity of interference fringes = 9/4
```

Scilab code Exa 2.4 Wavelength of light from monochromatic coherent sources

```
// Scilab Code Ex2.4:: Page-2.10 (2009)
clc; clear;
d = 0.02; // Separation between the slits, cm
D = 100;    // Distance of the source from the screen, m
n = 6;  // No. of bright fringe from the centre
x = 1.22;   // Position of 6th bright fringe, cm
lambda = x*d/(n*D);  // Wavelength of light used, m
printf("\nThe wavelength of the light from coherent sources = %5.3e cm", lambda);

// Result
// The wavelength of the light from coherent sources = 4.067e-005 cm
```

Scilab code Exa 2.5 Separation between fourth order dark fringes

```
// Scilab Code Ex2.5:: Page-2.10 (2009)
clc; clear;
lambda1 = 5890e-008;    // Wavelength of D1 line of sodium, cm
lambda2 = 5896e-008;    // Wavelength of D2 line of sodium, cm
D = 120;    // Distance between source and the screen, cm
d = 0.025;  // Separation between the slits, cm
n = 4;  // Order of dark fringe
```

```
x1 = (2*n+1)*lambda1*D/(2*d);    // Position of 4th dark fringe due to D1 line, cm
x2 = (2*n+1)*lambda2*D/(2*d);    // Position of 4th dark fringe due to D2 line, cm
delta_x = x2-x1;    // Fringe separation, cm

printf("\nThe separation between fourth order dark fringes = %4.2e cm", x2-x1);

// Result
// The separation between fourth order dark fringes = 1.30e-03 cm
```

Scilab code Exa 2.6 Distance between two coherent sources

```
// Scilab Code Ex2.6:: Page-2.11 (2009)
clc; clear;
lambda = 5500e-008;    // Wavelength of light used, cm
Y1 = 10;    // Distance of biprism from the source, cm
Y2 = 90;    // Distance of biprism from the screen, cm
D = Y1 + Y2;    // Distance between slits and the screen, cm
b = 8.526e-02;  // Fringe width, cm
d = lambda*D/b;  // Separation between the slits, cm

printf("\nThe distance between two coherent sources = %4.2e cm", d);

// Result
// The distance between two coherent sources = 6.45e-02 cm
```

Scilab code Exa 2.7 Fringe width of the interference pattern due to biprism

```
// Scilab Code Ex2.7:: Page-2.11 (2009)
clc; clear;
alpha = %pi/180;     // Acute angle of biprism,
    radian
mu = 1.5;    // Refractive index of biprism
lambda = 5500e-008;     // Wavelength of light used,
    cm
y1 = 5;     // Distance of biprism from the source,
    cm
y2 = 75;     // Distance of biprism from the screen,
    cm
D = y1 + y2;     // Distance between slits and the
    screen, cm
d = 2*(mu-1)*alpha*y1;  // Separation between the
    slits, cm
b = lambda*D/d;  // Fringe width of the interfernce
    pattern due to biprism, cm

printf("\nThe fringe width of the interfernce
    pattern due to biprism = %4.2e cm", b);

// Result
// The fringe width of the interfernce pattern due
    to biprism = 5.04e-02 cm
```

Scilab code Exa 2.8 Angle of vertex of the biprism

```
// Scilab Code Ex2.8:: Page-2.11 (2009)
clc; clear;
mu = 1.5;    // Refractive index of biprism
```

```
lambda = 5500e-008;     // Wavelength of light used,
    cm
y1 = 5;     // Distance of biprism from the source,
    cm
y2 = 95;     // Distance of biprism from the screen,
    cm
D = y1 + y2;     // Distance between slits and the
    screen, cm
b = 0.025;  // Fringe width of the interfernce
    pattern due to biprism, cm
// As d = 2*(mu-1)*alpha*y1, solving for alpha
alpha = lambda*D/(b*2*(mu-1)*y1)     // Angle of
    vertex of the biprism, radian

printf("\nThe angle of vertex of the biprism = %3.1e
     rad", alpha);

// Result
// The angle of vertex of the biprism = 4.4e-02 rad
```

Scilab code Exa 2.9 Number of interference fringes for changed wavelength

```
// Scilab Code Ex2.9:: Page-2.12 (2009)
clc; clear;
n1 = 69;     // Number of interference fringes
    obtained with yellow wavelength
lambda1 = 5893e-008;     // Wavelength of yellow
    light used, cm
lambda2 = 5461e-008;     // Wavelength of green light
     used, cm
// As n*lambda = l*d/D = constant, therefore
n2 = n1*lambda1/lambda2;     // Number of
    interference fringes for green wavelength

printf("\nThe number of interference fringes for
```

```
    changed wavelength = %2d", ceil(n2));

// Result
// The number of interference fringes for changed
    wavelength = 75
```

Scilab code Exa 2.10 Wavelength of light in a biprism experiment

```
// Scilab Code Ex2.10:: Page-2.12 (2009)
clc; clear;
D = 100;     // Distance between slits and the screen
    , cm
d = 0.08;   // Separation between the slits, cm
b = 2.121/25;   // Fringe width of the interfernce
    pattern due to biprism, cm
lambda = b*d/D;      // Wavelength of light in a
    biprism experiment, cm

printf("\nThe wavelength of light in a biprism
    experiment = %5.0f angstrom", lambda/1e-008);

// Result
// The wavelength of light in a biprism experiment =
    6787 angstrom
```

Scilab code Exa 2.11 Fringe width at a certain distance from biprism

```
// Scilab Code Ex2.11:: Page-2.13 (2009)
clc; clear;
alpha = %pi/180;      // Acute angle of biprism,
    radian
mu = 1.5;    // Refractive index of biprism
```

```
lambda = 5900e-008;     // Wavelength of light used, cm
y1 = 10;     // Distance of biprism from the source, cm
y2 = 100;     // Distance of biprism from the screen, cm
D = y1 + y2;     // Distance between slits and the screen, cm
d = 2*(mu-1)*alpha*y1;  // Separation between the slits, cm
b = lambda*D/d;  // Fringe width of the interfernce pattern due to biprism, cm

printf("\nThe fringe width at a distance of %d cm from biprism = %4.2e cm", y2, b);

// Result
// The fringe width at a distance of 100 cm from biprism = 3.72e-02 cm
```

Scilab code Exa 2.12 Distance between coherent sources in biprism experiment

```
// Scilab Code Ex2.12:: Page-2.13 (2009)
clc; clear;
lambda = 5893e-008;     // Wavelength of light used, cm
y1 = 10;     // Distance of biprism from the source, cm
y2 = 100;     // Distance of biprism from the screen, cm
D = y1 + y2;     // Distance between slits and the screen, cm
b = 3.5e-02;  // Fringe width of the interfernce pattern due to biprism, cm
```

```
d = lambda*D/b;   // Distance between coherent sources , cm

printf("\nThe distance between coherent sources = %5.3f cm", d);

// Result
// The distance between coherent sources = 0.185 cm
```

Scilab code Exa 2.13 Effect of slit separation on fringe width

```
// Scilab Code Ex2.13:: Page-2.13 (2009)
clc; clear;
b = 0.125;  // Fringe width of the interfernce pattern due to biprism , cm
d = 1;  // For simplicity assume distance between sources to be unity , cm
d_prime = 3/4*d;     // New distance between sources , cm
// As b is proportional to 1/d, so
b_prime = b*d/d_prime;   // New fringe width of the interfernce pattern due to biprism , cm

printf("\nThe new value of fringe width due to reduced slit separation = %5.3f cm", b_prime);

// Result
// The new value of fringe width due to reduced slit separation = 0.167 cm
```

Scilab code Exa 2.14 Effect of slit biprism separation on fringe width

```
// Scilab Code Ex2.14:: Page-2.13 (2009)
```

```
clc; clear;
b = 0.187;  // Fringe width of the interfernce
    pattern due to biprism, cm
y1 = 1;  // For simplicity assume distance between
    slit and biprism to be unity, cm
y1_prime = 1.25*y1;    // New distance between slit
    and biprism, cm
// As d is directly proportional to y1 and b is
    directly proportional to d, so
// b is inversely proportional to y1
b_prime = b*y1/y1_prime;  // New fringe width of the
    interfernce pattern due to biprism, cm

printf("\nThe new value of fringe width due to
    increased slit-biprism separation = %5.3f cm",
    b_prime);

// Result
// The new value of fringe width due to increased
    slit-biprism separation = 0.150 cm
```

Scilab code Exa 2.15 Distance between interference bands

```
// Scilab Code Ex2.15:: Page-2.14 (2009)
clc; clear;
d1 = 5e-01; // First distance between images of the
    slit, cm
d2 = 2.25e-01; // Second distance between images of
    the slit, cm
lambda = 5896e-008; // Wavelength of the light used,
    cm
D = 120;    // Distance between screen and the slits
    , cm
d = sqrt(d1*d2);    // Geometric mean of distance
    between the two slits, cm
```

```
b = lambda*D/d;      // Distance between interference bands, cm

printf("\nThe distance between interference bands = %5.3e cm", b);

// Result
// The distance between interference bands = 2.109e-02 cm
```

Scilab code Exa 2.16 Angle of vertex of Fresnel biprism

```
// Scilab Code Ex2.16:: Page-2.14 (2009)
clc; clear;
mu = 1.5;    // Refractive index of biprism
lambda = 5500e-008;     // Wavelength of light used, cm
y1 = 25;     // Distance of biprism from the source, cm
y2 = 150;     // Distance of biprism from the screen, cm
D = y1 + y2;     // Distance between slits and the screen, cm
b = 0.05;   // Fringe width of the interfernce pattern due to biprism, cm
// As d = 2*(mu-1)*alpha*y1, solving for alpha
alpha = lambda*D/(b*2*(mu-1)*y1)     // Angle of vertex of the biprism, radian

printf("\nThe angle of vertex of the biprism = %6.4f rad", alpha);

// Result
// The angle of vertex of the biprism = 0.0077 rad
```

Scilab code Exa 2.17 Wavelength of light used in biprism experiment to illuminate slits

```
// Scilab Code Ex2.17:: Page-2.15 (2009)
clc; clear;
theta = 178;    // Vertex angle of biprism, degrees
alpha = (180-theta)/2*%pi/180;    // Acute angle of biprism, radian
mu = 1.5;   // Refractive index of biprism
y1 = 20;    // Distance of biprism from the source, cm
y2 = 125;    // Distance of biprism from the screen, cm
D = y1 + y2;    // Distance between slits and the screen, cm
d = 2*(mu-1)*alpha*y1;  // Separation between the slits, cm
b = 0.025;  // Fringe width of the interfernce pattern due to biprism, cm
lambda = b*d/D;    // Wavelength of light used, cm

printf("\nThe wavelength of light used to illuminate slits = %4d angstrom", lambda/1e-08);

// Result
// The wavelength of light used to illuminate slits = 6018 angstrom
```

Scilab code Exa 2.18 Vertex angle of Fresnel biprism

```
// Scilab Code Ex2.18:: Page-2.15 (2009)
clc; clear;
```

```
mu = 1.5;    // Refractive index of biprism
lambda = 6600e-008;    // Wavelength of light used, cm
y1 = 40;    // Distance of biprism from the source, cm
y2 = 175;    // Distance of biprism from the screen, cm
D = y1 + y2;    // Distance between slits and the screen, cm
b = 0.04;   // Fringe width of the interfernce pattern due to biprism, cm
// As d = 2*(mu-1)*alpha*y1, solving for alpha
alpha = lambda*D/(b*2*(mu-1)*y1)    // Acute angle of the biprism, radian
theta = (%pi-2*alpha);  // Vertex angle of the biprism, radian

printf("\nThe vertex angle of the biprism = %6.2f degrees", theta*180/%pi);

// Result
// The vertex angle of the biprism = 178.98 degrees
```

Scilab code Exa 2.19 Order of visible fringe for changed wavelength of light

```
// Scilab Code Ex2.19: : Page-2.16 (2009)
clc; clear;
lambda1 = 7000e-008;    // Original wavelength of light, cm
lambda2 = 5000e-008;    // New wavelength of light, cm
n1 = 10;    // Order of the fringes with original wavelength
// As x = n*lambda*D/d, so n*lambda = constant
```

```
// n1*lambda1 = n2*lambda2, solving for n2
n2 = n1*lambda1/lambda2;    // Order of visible fringe for changed wavelength of light

printf("\nThe order of visible fringe for changed wavelength of light = %2d", ceil(n2));

// Result
// The order of visible fringe for changed wavelength of light = 14
```

Scilab code Exa 2.20 Angle of vertex of biprism

```
// Scilab Code Ex1.20:: Page-2.16 (2009)
clc; clear;
y1 = 40;    // Distance between biprism from the slit, cm
D = 160;    // Distance between slit and the screen, cm
mu = 1.52;  // Refractive index of material of the prism
lambda = 5893e-008; // Wavelength of light used, cm
b = 0.01;   // Fringe width, cm
// As b = lambda*D/d, solving for d
d = lambda*D/b;     // Distance between virtual sources, cm
// But d = 2*y1*(mu-1)*alpha, solving for alpha
alpha = d/(2*y1*(mu-1))*180/%pi;    // Angle of biprism, degrees
theta = 180-2*alpha;    // Angle of vertex of biprism, degrees

printf("\nThe angle of vertex of biprism = %5.1f degree", theta);

```

```
// Result
// The angle of vertex of biprism = 177.4 degree
```

Scilab code Exa 2.21 Separation between two coherent sources

```
// Scilab Code Ex2.21: : Page-2.16 (2009)
clc; clear;
lambda = 6000e-008;     // Wavelength of light used, cm
D = 100;     // Distance between slits and the screen, cm
b = 0.05;   // Fringe width of the interfernce pattern due to biprism, cm
d = lambda*D/b;   // Distance between coherent sources, cm

printf("\nThe distance between coherent sources = %3.1f mm", d/1e-01);

// Result
// The distance between coherent sources = 1.2 mm
```

Scilab code Exa 2.22 Refractive index of the glass sheet

```
// Scilab Code Ex2.22:: Page-2.19 (2009)
clc; clear;
t = 3.2e-04;     // Thickness of the glass sheet, cm
lambda = 5500e-008;      // Wavelength of light used, cm
n = 5;        // Order of interference fringes
// As path difference (mu - 1)*t = n*lambda
mu = n*lambda/t + 1;     // Refractive indexof the glass sheet
```

```

printf("\nThe refractive index of the glass sheet=
    %4.2f", mu);

// Result
// The refractive indexof the glass sheet= 1.86
```

Scilab code Exa 2.23 Refractive index of material of sheet

```
// Scilab Code Ex2.23:: Page-2.19 (2009)
clc; clear;
t = 2.1e-03;     // Thickness of the glass sheet, cm
lambda = 5400e-008;     // Wavelength of light used,
    cm
n = 11;     // Order of interference fringes
// As path difference, (mu - 1)*t = n*lambda
mu = n*lambda/t + 1;     // Refractive index of the
    glass sheet

printf("\nThe refractive index of the glass sheet =
    %4.2f", mu);

// Result
// The refractive index of the glass sheet= 1.28
```

Scilab code Exa 2.24 Wavelength of light used in biprism arrangement

```
// Scilab Code Ex2.24:: Page-2.19 (2009)
clc; clear;
t = 9.21e-05;     // Thickness of the mica sheet, cm
mu = 1.5;     // Refractive index of material of sheet
n = 1;     // Order of interference fringes
```

```
// As path difference, (mu - 1)*t = n*lambda, solving for lambda
lambda = (mu - 1)*t/n;    // Wavelength of light used, cm

printf("\nThe wavelength of light used = %5.3e cm", lambda);

// Result
// The wavelength of light used = 4.605e-005 cm
```

Scilab code Exa 2.25 Thickness of the transparent sheet

```
// Scilab Code Ex2.25:: Page-2.19 (2009)
clc; clear;
lambda = 5890e-008;     // Wavelength of light used, cm
mu = 1.5;    // Refractive index of material sheet
// As shift = 9*lambda*D/d = D/d*(mu - 1)*t, solving for t
t = 9*lambda/(mu - 1);    // Thickness of the glass sheet, cm
printf("\nThe thickness of the glass sheet = %4.2e cm", t);

// Result
// The thickness of the glass sheet = 1.06e-003 cm
```

Scilab code Exa 2.26 Thickness of the transparent sheet from fringe shift

```
// Scilab Code Ex2.26:: Page-2.20 (2009)
clc; clear;
```

```
lambda = 5400e-008;     // Wavelength of light used, cm
mu = 1.7;    // Refractive index of material sheet convering the first slit
mu_prime = 1.5;     // Refractive index of material sheet convering the seecond slit
// As shift, S = D/d*(mu - mu_prime)*t = b/lambda*(mu - mu_prime)*t, solving for t
t = 8*lambda/(mu-mu_prime)     // Thickness of the glass sheet, cm

printf("\nThe thickness of the glass sheet = %4.2e cm", t);

// Result
// The thickness of the glass sheet = 2.16e-003 cm
```

Scilab code Exa 2.27 Refractive index of thin mica sheet

```
// Scilab Code Ex2.27:: Page-2.20 (2009)
clc; clear;
t = 21.5e-05;     // Thickness of the glass sheet, cm
lambda = 5890e-008;     // Wavelength of light used, cm
n = 1;       // Order of interference fringes
// As path difference, (mu - 1)*t = n*lambda
mu = n*lambda/t + 1;     // Refractive indexof the glass sheet

printf("\nThe refractive index of the glass sheet = %5.3f", mu);

// Result
// The refractive index of the glass sheet = 1.274
```

Scilab code Exa 2.28 Wavelength of light used in double slit experiment

```
// Scilab Code Ex2.28:: Page-2.20 (2009)
clc; clear;
D = 1;   // For simplicity assume distance between
    source and slits to be unity, unit
d = 1;   // For simplicity assume slit separation to
     be unity, unit
t = 2.964e-06;    // Thickness of the mica sheet, cm
mu = 1.5;   // Refractive index of material of shee
L = poly(0, 'L');
// As b = b_prime or 2.25*D*L/d = D/d*(mu-1)*t, or
    we may write
L = roots(2.25*D*L/d-D/d*(mu-1)*t);      //
    Wavelength of the light used, m

printf("\nThe wavelength of the light used = %4.0f
    angstrom", L/1e-010);

// Result
// The wavelength of the light used = 6587 angstrom
```

Scilab code Exa 2.29 Thickness of mica sheet from central fringe shift

```
// Scilab Code Ex2.29:: Page-2.21 (2009)
clc; clear;
lambda = 5890e-008;      // Wavelength of light used,
     cm
n = 5;      // Order of interference fringes
mu = 1.5;   // Refractive index of the mica sheet
// As path difference, (mu - 1)*t = n*lambda,
    solving for t
```

```
t = n*lambda/(mu-1);     // Thickness of the mica sheet, cm

printf("\nThe thickness of the mica sheet = %4.2e cm", t);

// Result
// The thickness of the mica sheet = 5.89e-004 cm
```

Scilab code Exa 2.30 Refractive index of material from shifting fringe pattern

```
// Scilab Code Ex2.30:: Page-2.21 (2009)
clc; clear;
b = 1;      // For simplicity assume fringe width to be unity, cm
S = 30*b;   // Fringe shift, cm
lambda = 6600e-008;     // Wavelength of light used, cm
t = 4.9e-003;       // Thickness of the film, cm
// As S = b/lambda*(mu-1)*t, solving for mu
mu = S*lambda/t + 1;        // Refractive index of material from shifting fringe pattern

printf("\nThe refractive index of material from shifting fringe pattern = %3.1f", mu);

// Result
// The refractive index of material from shifting fringe pattern = 1.4
```

Scilab code Exa 2.31 Fringe width and optical path change during interference of waves through mica and glass

```
// Scilab Code Ex2.31:: Page-2.22 (2009)
clc; clear;
mu1 = 1.55;     // Refractive index of mica
mu2 = 1.52;     // Refractive index of glass
t = 0.75e-003;  // Thickness of the sheets, m
d = 0.25e-02;   // Separation between the slits, m
lambda = 5896e-010; // Wavelength of light used, m
D = 1.5;    // Distance between the source ans the slits, m
// Fringe width
b = lambda*D/d;     // Fringe width, m
// Optical path difference
delta_x = (mu1-1)*t-(mu2-1)*t;  // Optical path change, m

printf("\nThe fringe width = %3.1e m", b);
printf("\nThe optical path change = %5.3e m", delta_x);

// Result
// The fringe width = 3.5e-004 m
// The optical path change = 2.250e-005 m
```

Scilab code Exa 2.32 Thickness of mica sheet from Fresnel biprism experiment

```
// Scilab Code Ex2.32:: Page-2.22 (2009)
clc; clear;
b = 1;      // For simplicity assume fringe width to be unity, cm
S = 3*b;    // Fringe shift, cm
lambda = 5890e-008;     // Wavelength of light used, cm
mu = 1.6;    // Refractive index of the mica sheet
// As S = b/lambda*(mu-1)*t, solving for t
```

```
t = S*lambda/(mu-1);        // Thickness of the mica
    sheet , cm

printf("\nThe thickness of the mica sheet = %3.1e m"
    , t/1e+02);

// Result
// The thickness of the mica sheet = 2.9e-006 m
```

Scilab code Exa 2.33 Smallest thickness of glass plate for a fringe of minimum intensity

```
// Scilab Code Ex2.33: : Page-2.26 (2009)
clc; clear;
mu = 1.5;         // Refractive index of glass
lambda = 5100e-008; // Wavelength of light used, cm
i = 30;       // Angle of incidence, degrees
n = 1;   // Order of interference fringes
// From Snell's law, mu = sind(i)/sind(r), solving
    for r
r = asind(sind(i)/mu);   // Angle of refraction,
    degrees
// For a dark fringe in reflection, 2*mu*t*cosd(r) =
     n*lambda, solving for t
t = n*lambda/(2*mu*cosd(r));        // Smallest
    thickness of glass plate for a fringe of minimum
    intensity, cm
printf("\nThe smallest thickness of glass plate for
    a fringe of minimum intensity = %4.2e cm", t);

// Result
// The smallest thickness of glass plate for a
    fringe of minimum intensity = 1.80e-005 cm
```

Scilab code Exa 2.34 The wavelength reflected strongly from the soap film

```
// Scilab Code Ex2.34:: Page-2.26 (2009)
clc; clear;
t = 3.1e-05;     // Thickness of the soap film, cm
mu = 1.33;       // Refractive index of the soap film
r = 0;       // Angle of refraction of the light ray
   on the soap film, degrees
// For bright fringe in reflected pattern,
// 2*mu*t*cosd(r) = (2*n+1)*lambda/2
lambda = zeros(3);
for n = 1:1:3
    lambda(n) = 4*mu*t*cosd(r)/(2*(n-1)+1);      //
       Wavelengths for n = 1, 2 and 3
    if lambda(n) > 4000e-008 & lambda(n) < 7500e-008
        then
        lambda_reflected = lambda(n);
    end
end

printf("\nThe wavelength reflected strongly from the
    soap film = %5.3e cm", lambda_reflected);

// Result
// The wavelength reflected strongly from the soap
   film = 5.497e-05 cm
```

Scilab code Exa 2.35 Order of interference of the dark band

```
// Scilab Code Ex2.35:: Page-2.27 (2009)
clc; clear;
```

```
t = 3.8e-05;     // Thickness of the transparent film, cm
mu = 1.5;       // Refractive index of the transparent film
i = 45;      // Angle of incidence of the light ray on the transparent film, degrees
lambda = 5700e-008;      // Wavelength of light, cm
// As mu = sind(i)/sind(r), solving for r
r = asind(sind(i)/mu);
// For dark fringe in reflected pattern,
// 2*mu*t*cosd(r) = 2*n*lambda, solving for n
n = 2*mu*t*cosd(r)/lambda;       // Order of interference of dark band

printf("\nThe order of interference of dark band = %d", ceil(n));

// Result
// The order of interference of dark band = 2 velength reflected strongly from the soap film = 5.497e-05 cm
```

Scilab code Exa 2.36 Absent wavelength of reflected light in the visible spectrum

```
// Scilab Code Ex2.36:: Page-2.27 (2009)
clc; clear;
t = 4.5e-05;     // Thickness of the soap film, cm
mu = 1.33;       // Refractive index of the soap film
i = 45;      // Angle of incidence of the light ray on the soap film, degrees
// As mu = sind(i)/sind(r), solving for r
r = asind(sind(i)/mu);
// For dark fringe in reflected pattern,
// 2*mu*t*cosd(r) = n*lambda, solving for lambda for
```

```
    different n's
lambda = zeros(4);
for n = 1:1:4
    lambda(n) = 2*mu*t*cosd(r)/n;     // Wavelengths for n = 1, 2, 3 and 4
    if lambda(n) > 4000e-008 & lambda(n) < 7500e-008 then
        lambda_absent = lambda(n);
    end
end
printf("\nThe absent wavelength of reflected light in the visible spectrum = %4.2e", lambda_absent);

// Result
// The absent wavelength of reflected light in the visible spectrum = 5.07e-05
```

Scilab code Exa 2.37 Minimum thickness of the plate that will appear dark in the reflection pattern

```
// Scilab Code Ex2.37:: Page-2.28 (2009)
clc; clear;
mu = 1.6;       // Refractive index of the mica plate
r = 60;      // Angle of refraction of the light ray on the mica plate, degrees
lambda = 5500e-008;      // Wavelength of light used, cm
n = 1;  // Order of interference for minimum thickness
// For dark fringe in reflected pattern,
// 2*mu*t*cosd(r) = 2*n*lambda, solving for t
t = n*lambda/(2*mu*cosd(r));     // Minimum thickness of the plate that will appear dark in the reflection pattern

```

```
printf("\nThe minimum thickness of the plate that will appear dark in the reflection pattern = %4.2e cm", t);

// Result
// The minimum thickness of the plate that will appear dark in the reflection pattern = 3.44e-05 cm
```

Scilab code Exa 2.38 Thickness of the thin soap film

```
// Scilab Code Ex2.38:: Page-2.28 (2009)
clc; clear;
mu = 1.33;        // Refractive index of the thin soap film
lambda1 = 5500e-008;     // Wavelength of the first dark fringe, cm
lambda2 = 5400e-008;     // Wavelength of the consecutive dark fringe, cm
i = 30;      // Angle of incidence of the light ray on the soap film, degrees
// For overlapping fringes,
// n*lambda1 = (n+1)*lambda2, solving for n
n = lambda2/(lambda1-lambda2);   // Order of interference fringes
// As mu = sind(i)/sind(r), solving for r
r = asind(sind(i)/mu);
// For dark fringe in reflected pattern,
// 2*mu*t*cosd(r) = 2*n*lambda1, solving for t
t = n*lambda1/(2*mu*cosd(r));     // Thickness of the thin soap film

printf("\nThe thickness of the thin soap film = %5.3e cm", t);

```

```
// Result
// The thickness of the thin soap film = 1.205e−03
   cm
```

Scilab code Exa 2.39 Order of interference for which light is strongly reflected

```
// Scilab Code Ex2.39:: Page−2.29 (2009)
clc; clear;
t = 0.75e-06;   // Thickness of the glass plate, m
mu = 1.5;       // Refractive index of the glass
   plate
lambda1 = 4000e-010;    // First wavelength of
   visible range, cm
lambda2 = 7000e-010;    // Last wavelength of
   visible range, cm
r = 0;      // Angle of refraction for normal
   incidence, degrees
n = zeros(2);
// For bright fringe in reflected pattern,
// 2*mu*t*cosd(r) = (2*n+1)*lambda/2, solving for n
// For lambda1
n(1) = (4*mu*t*cosd(r)/lambda1-1)/2;
// For lambda2
n(2) = (4*mu*t*cosd(r)/lambda2-1)/2;

printf("\nFor n = %d and n = %d the light is
   strongly reflected.", n(1), ceil(n(2)));

// Result
// For n = 5 and n = 3 the light is strongly
   reflected.
```

Scilab code Exa 2.40 Minimum thickness of the film for which light is strongly reflected

```
// Scilab Code Ex2.40:: Page-2.30 (2009)
clc; clear;
mu = 1.45;       // Refractive index of the film
lambda = 5500e-010;     // First wavelength of visible range, cm
r = 0;      // Angle of refraction for normal incidence, degrees
n = 0;       // Order of interference is zero for minimum thickness
// For bright fringe in reflected pattern,
// 2*mu*t*cosd(r) = (2*n+1)*lambda/2, solving for t
t = (2*n+1)*lambda/(4*mu*cosd(r));     // Minimum thickness of the film for which light is strongly reflected

printf("\nThe minimum thickness of the film for which light is strongly reflected = %4.2e cm", t);

// Result
// The minimum thickness of the film for which light is strongly reflected = 9.48e-08 cm
```

Scilab code Exa 2.41 Thickness of the soap film for dark fringe in reflected pattern

```
// Scilab Code Ex2.41:: Page-2.30 (2009)
clc; clear;
mu = 5/4;       // Refractive index of the film
lambda = 5890e-010;     // Wavelength of visible light, cm
i = 45;      // Angle of incidence, degrees
```

```
n = 1;       // Order of interference is unity for minimum thickness in dark reflected pattern
// As mu = sind(i)/sind(r), solving for r
r = asind(sind(i)/mu);
// For dark fringe in reflected pattern,
// 2*mu*t*cosd(r) = n*lambda, solving for t
t = n*lambda/(2*mu*cosd(r));    // Thickness of the soap film for dark fringe in reflected pattern

printf("\nThe thickness of the soap film for dark fringe in reflected pattern = %5.3e cm", t);

// Result
// The thickness of the soap film for dark fringe in reflected pattern = 2.857e-07 cm
```

Scilab code Exa 2.42 Wavelength in the visible range which is intensified in the reflected beam

```
// Scilab Code Ex2.42:: Page-2.30 (2009)
clc; clear;
mu = 1.5;      // Refractive index of the plate
t = 0.5e-006;    // Thickness of the plate, m
r = 0;       // Angle of refraction for normal incidence, degrees
// For bright fringe in reflected pattern,
// 2*mu*t*cosd(r) = (2*n+1)*lambda/2, solving for lambda for different n's
lambda = zeros(4);
for n = 0:1:3
    lambda(n+1) = 4*mu*t*cosd(r)/(2*n+1);     // Wavelengths for n = 0, 1, 2 and 3
    lambda_strong = lambda(n+1);
    if lambda(n+1) >= 4000e-010 & lambda(n+1) <= 7500e-010 then
```

```
            if lambda_strong > lambda(n+1)  then  //
               Search for the stronger wavelength
              lambda_strong = lambda(n+1);
            end
      end
end

printf("\nFor n = %d, %4.0f angstrom will be
   reflected strongly", n, lambda_strong/1e-010);

// Result
// For n = 3, 4286 angstrom will be reflected
   strongly
```

Scilab code Exa 2.43 Thickness of the film with incident white light

```
// Scilab Code Ex2.43:: Page-2.31(2009)
clc; clear;
mu = 1.33;       // Refractive index of the film
i = asind(0.8);       // Angle of refraction for
   normal incidence, degrees
// As mu = sind(i)/sind(r), solving for r
r = asind(sind(i)/mu);
lambda1 = 6100e-010;    // First wavelength of dark
   band, m
lambda2 = 6000e-010;    // Second wavelength of dark
    band, m
// For consecutive overlapping wavelenghts
// n*lambda1 = (n+1)*lambda2, solving for n
n = lambda2/(lambda1-lambda2);
// For dark fringe in reflected pattern,
// 2*mu*t*cosd(r) = n*lambda1, solving for t
t = n*lambda1/(2*mu*cosd(r));   // Thickness of the
   film with incident white light. m
printf("\nThickness of the film with incident white
```

```
    light = %3.1e m", t);

// Result
// Thickness of the film with incident white light =
    1.7e-05 m
```

Scilab code Exa 2.44 Thickness of the film with parallel beam of yellow light

```
// Scilab Code Ex2.44:: Page-2.31(2009)
clc; clear;
mu = 1.5;      // Refractive index of the film
i = 45;      // Angle of incidence, degrees
// As mu = sind(i)/sind(r), solving for r
r = asind(sind(i)/mu);
lambda = 5500e-010;     // Wavelength of parallel
    beam of light, m
n = 15;     // Order of dark band
// For dark fringe in reflected pattern,
// 2*mu*t*cosd(r) = n*lambda, solving for t
t = n*lambda/(2*mu*cosd(r));   // Thickness of the
    film with incident parallel beam of light. m

printf("\nThe thickness of the film with paralle
    beam of yellow light = %4.2e m", t);

// Result
// The thickness of the film with paralle beam of
    yellow light = 3.12e-06 m
```

Scilab code Exa 2.46 Refractive index of oil

```
// Scilab Code Ex2.46:: Page-2.33(2009)
```

```
clc; clear;
V = 0.58e-006;    // Volume of oil, metre cube
A = 2.5;          // Area of water surface, metre square
t = V/A;     // Thickness of film, m
r = 0;       // Angle of refraction for normal incidence, degrees
n = 1;       // Order of interference for minimum thickness
lambda = 4700e-010;      // Wavelength of light used, m
// For dark fringe in reflected pattern,
// 2*mu*t*cosd(r) = n*lambda, solving for mu
mu = n*lambda/(2*t*cosd(r));     // Refractive index of oil

printf("\nThe refractive index of oil = %5.3f", mu);

// Result
// The refractive index of oil = 1.013
```

Scilab code Exa 2.47 Thickness of the soap film to produce constructive interference during reflection

```
// Scilab Code Ex2.47:: Page-2.33(2009)
clc; clear;
mu = 1.46;        // Refractive index of the soap film
lambda = 6000e-010;      // Wavelength of light used, m
r = 0;       // Angle of refraction for normal incidence, degrees
n = 0;       // Order of interference for minimum thickness
// For bright fringe in reflected pattern,
// 2*mu*t*cosd(r) = (2*n+1)*lambda/2, solving for mu
```

```
t = (2*n+1)*lambda/(4*mu*cosd(r));   // Thickness of
    soap film , m

printf("\nThe thickness of soap film = %5.3e m", t);

// Result
// The thickness of soap film = 1.027e-07 m
```

Scilab code Exa 2.48 Wavelength of light falling on wedge shaped film

```
// Scilab Code Ex2.48: : Page-2.35(2009)
clc; clear;
mu = 1.4;        // Refractive index of the film
alpha = 1.07e-004;   // Acute angle of the wedge,
    radian
b = 0.2;     // Fringe width, cm
// As b = lambda/(2*mu*alpha), solving for lambda
lambda = 2*mu*alpha*b;   // Wavelength of light
    falling on wedge shaped film, m

printf("\nThe wavelength of light falling on wedge
    shaped film = %4d ansgtrom", lambda/1e-008);

// Result
// The wavelength of light falling on wedge shaped
    film = 5991 ansgtrom
```

Scilab code Exa 2.49 Difference between the thicknesses of the films

```
// Scilab Code Ex2.49:: Page-2.35(2009)
clc; clear;
mu = 1.4;        // Refractive index of the film
lambda = 5500e-008; // Wavelength of the light, cm
```

```
// As alpha = (delta_t)/x and x = 10*b; b = lambda/(2*mu*alpha), solving for dt
delta_t = 10*lambda/(2*mu);      // Difference between the thicknesses of the films, cm

printf("\nDifference between the thicknesses of the films = %4.2e cm", delta_t);

// Result
// Difference between the thicknesses of the films = 1.96e-04 cm
```

Scilab code Exa 2.50 Angle of thin wedge shaped film

```
// Scilab Code Ex2.50:: Page-2.36(2009)
clc; clear;
mu = 1.6;           // Refractive index of the film
lambda = 5500e-008; // Wavelength of the light, cm
b = 0.1;            // Fringe width, cm
// As b = lambda/(2*mu*alpha), solving for alpha
alpha = lambda/(2*mu*b);     // Angle of thin wedge shaped film, radian
printf("\nAngle of thin wedge shaped film = %3.1e radian", alpha);

// Result
// Angle of thin wedge shaped film = 1.7e-04 radian
```

Scilab code Exa 2.51 Wavelength of light used to illuminate a wedge shaped film

```
// Scilab Code Ex2.51:: Page-2.36(2009)
clc; clear;
```

```
mu = 1.5;        // Refractive index of the film
b = 0.20;        // Fringe width, cm
theta = 25/(60*60)*%pi/180;      // Angle of the wedge, radian
// As b = lambda/(2*mu*theta), solving for lambda
lambda = 2*mu*b*theta;     // Wavelength of light used to illuminate a wedge shaped film, cm

printf("\nThe wavelength of light used to illuminate a wedge shaped film = %4d angstrom", lambda/1e-008);

// Result
// The wavelength of light used to illuminate a wedge shaped film = 7272 angstrom
// The answer is given wrong in the textbook
```

Scilab code Exa 2.52 Thickness of the wire separating two glass surfaces

```
// Scilab Code Ex2.52:: Page-2.36(2009)
clc; clear;
lambda = 5893e-010;      // Wavelength of light used, m
mu = 1;       // Refractive index of the glass
b = 1;        // Assume fringe width to be unity, cm
// As b = l/20, solving for l
l = b*20;        // Length of the film, m
// As b = lambda/(2*mu*theta) and theta = t/l, solving for t
t = lambda*l/(2*mu);     // Thickness of the wire separating two glass surfaces, m

printf("\nThe thickness of the wire separating two glass surfaces = %4.2e m", t);

```

```
// Result
// The thickness of the wire separating two glass surfaces = 5.89e-06 m
```

Scilab code Exa 2.53 Angle of the wedge shaped air film

```
// Scilab Code Ex2.53:: Page-2.37(2009)
clc; clear;
mu = 1;        // Refractive index of the air film
b = 1.5/25;        // Fringe width, cm
lambda = 5893e-008;     // Wavelength of light used to illuminate a wedge shaped film, cm
// As b = lambda/(2*mu*theta), solving for theta
theta = lambda/(2*mu*b);     // Angle of the wedge, radian

printf("\nThe angle of the wedge shaped air film = %5.3f degrees", theta*180/%pi);

// Result
// The angle of the wedge shaped air film = 0.028 degrees
```

Scilab code Exa 2.54 Acute angle of the wedge shaped film

```
// Scilab Code Ex2.54:: Page-2.37(2009)
clc; clear;
mu = 1.45;        // Refractive index of the film
b = 1/10;        // Fringe width, cm
lambda = 6600e-008;     // Wavelength of light used to illuminate a wedge shaped film, cm
// As b = lambda/(2*mu*theta), solving for theta
```

```
theta = lambda/(2*mu*b);     // Angle of the wedge, radian

printf("\nThe acute angle of the wedge shaped film = %6.4f degrees", theta*180/%pi);

// Result
// The acute angle of the wedge shaped film = 0.0130 degrees
```

Scilab code Exa 2.55 Diameter of nth dark ring due to first wavelength

```
// Scilab Code Ex2.55:: Page-2.46(2009)
clc; clear;
lambda1 = 6000e-008;     // First visible wavelength, cm
lambda2 = 4500e-008;     // Second visible wavelength, cm
R = 100;          // Radius of curvature of the lens, cm
// As diameter of nth dark ring due to lambda1 is
// D_n^2 = 4*n*R*lambda1 and D_nplus1^ = 4*(n+1)*R*lambda2, so that D_n^2 = D_nplus1^2 gives
n = lambda2/(lambda1-lambda2);       // Order of interference for dark fringes
D_n = sqrt(4*n*R*lambda1);       // Diameter of nth dark ring due to lambda1

printf("\nThe diameter of nth dark ring due to wavelength of %4d angstrom = %4.2f cm", lambda1/1e-008, D_n);

// Result
// The diameter of nth dark ring due to wavelength of 6000 angstrom = 0.27 cm
```

Scilab code Exa 2.56 Diameter of fifteenth dark ring

```
// Scilab Code Ex2.56:: Page-2.46(2009)
clc; clear;
R = 1;        // For simplicity assume radius of
    curvature of the lens to be unity, cm
D_n = 0.251;      // Diameter of 3rd dark ring, cm
D_nplusp = 0.548;      // Diameter of 9th dark ring,
    cm
n = 3;        // Order of 3rd Newton ring
p = 9 - n;      // Order of 6th Newton ring from 3rd
    ring
// As D_nplusp^2 - D_n^2 = 4*p*R*lambda, solving for
    lambda
lambda = (D_nplusp^2 - D_n^2)/(4*p*R);       //
    Wavelength of light used
D_15 = sqrt(D_n^2+4*(15-n)*lambda*R);        //
    Diameter of 15th dark ring, cm

printf("\nThe diameter of 15th dark ring = %5.3f cm"
    , D_15);

// Result
// The diameter of 15th dark ring = 0.733 cm
```

Scilab code Exa 2.57 Order of a dark ring having thrice the diameter of the thirtieth ring

```
// Scilab Code Ex2.57: : Page-2.47(2009)
clc; clear;
R = 1;        // For simplicity assume radius of
    curvature of the lens to be unity, cm
```

```
n = 30;        // Order of 3rd Newton ring
D_30 = 1;    // Assume diameter of thirtieth ring to be unity, cm
// As D_30^2 = 4*n*R*lambda, solving for lambda
lambda = D_30^2/(4*n*R);     // Wavelength of light used, cm
D_n = 3*D_30;         // Diameter of nth dark ring having thrice the diameter of the thirtieth ring, cm
n = D_n^2/(4*R*lambda);      // Order of a dark ring having thrice the diameter of the thirtieth ring

printf("\nThe order of the dark ring having thrice the diameter of the thirtieth ring = %3d", n);

// Result
// The order of the dark ring having thrice the diameter of the thirtieth ring = 270
```

Scilab code Exa 2.58 Radius of curvature of lens and thickness of air film

```
// Scilab Code Ex2.58:: Page-2.47(2009)
clc; clear;
n = 15;       // Order of 15rd Newton ring
D_15 = 0.75;    // Diameter of fifteenth dark ring, cm
lambda = 5890e-008;      // Wavelength of light used, cm
// As D_15^2 = 4*15*R*lambda, solving for R
R = D_15^2/(4*15*lambda);     // Radius of curvature of lens, cm
// For dark ring, 2*t = n*lambda, solving for t
t = n*lambda/2;      // Thickness of air film, cm

printf("\nThe radius of curvature of lens = %5.1f cm
```

```
", R);
printf("\nThe thickness of air film = %3.1e cm", t);

// Result
// The radius of curvature of lens = 159.2 cm
// The thickness of air film = 4.4e-004 cm
```

Scilab code Exa 2.59 Refractive index of the liquid

```
// Scilab Code Ex2.59:: Page-2.47(2009)
clc; clear;
D_15 = 1.62;          // Diameter of 15th dark ring with air film, cm
D_15_prime = 1.47;          // Diameter of 15th dark ring with liquid, cm
R = 1;        // For simplicity assume radius of curvature to be unity, cm
n = 15;       // Order of 15rd Newton ring
// As for ring with air film, D_15^2 = 4*15*R*lambda, solving for lambda
lambda = D_15^2/(4*15*R);     // Wavelength of light used, cm
// As for ring with liquid, D_15_prime^2 = 4*15*R*lambda/mu, solving for mu
mu = 4*15*R*lambda/D_15_prime^2;          // Refractive index of the liquid
printf("\nThe refractive index of the liquid = %4.2f", mu)

// Result
// The refractive index of the liquid = 1.21
```

Scilab code Exa 2.60 Wavelength of light used in Newton rings experiment

```
// Scilab Code Ex2.60:: Page-2.48(2009)
clc; clear;
D_10 = 0.48;          // Diameter of 10th dark ring with air film, cm
D_3 = 0.291;          // Diameter of 3rd dark ring with air film, cm
p = 7;        // Order of the 10th ring next to the 3rd ring
R = 90;       // Radius of curvature of the lens, cm
lambda = (D_10^2-D_3^2)/(4*p*R);       // Wavelength of light used in Newton rings experiment

printf("\nThe wavelength of light used in Newton rings experiment = %4d angstrom", lambda/1e-008);

// Result
// The wavelength of light used in Newton rings experiment = 5782 angstrom
```

Scilab code Exa 2.61 Diameter of fifteenth bright ring

```
// Scilab Code Ex2.61:: Page-2.48(2009)
clc; clear;
R1 = 200;        // Radius of curvature of the convex surface, cm
R2 = 250;        // Radius of curvature of the concave surface, cm
lambda = 5500e-008; // Wavelength of light used, cm
n = 15;       // Order of interfernce Newton ring
// As r_n^2*(1/R1-1/R2) = (2*n-1)*lambda/2, solving for r_n
r_n = sqrt((2*n-1)*lambda/(2*(1/R1-1/R2)));      //
```

```
    Radius of nth ring, cm
D_15 = 2*r_n;        // Daimeter of 15th bright ring, cm

printf("\nThe daimeter of 15th bright ring = %4.2f cm", D_15);

// Result
// The daimeter of 15th bright ring = 1.79 cm
```

Scilab code Exa 2.62 Wavelength of light used in Newton rings experiment

```
// Scilab Code Ex2.62:: Page-2.49(2009)
clc; clear;
R = 80;        // Radius of curvature of the convex surface, cm
D5 = 0.192;     // Diameter of 5th dark ring, cm
D25 = 0.555;    // Diameter of 25th dark ring, cm
n = 5;      // Order of interfernce Newton ring
P = 25 - n;
lambda = (D25^2 - D5^2)/(4*P*R);    // Wavelength of light used, cm
printf("\nThe wavelength of light used = %5.3e cm", lambda);

// Result
// The wavelength of light used = 4.237e-005 cm
// The expression for lambda is given wrong in the textbook but solved correctly
```

Scilab code Exa 2.63 Diameter of fifteenth dark Newton ring

```
// Scilab Code Ex2.63:: Page-2.49(2009)
clc; clear;
R1 = 4;        // Radius of curvature of the convex surface, m
R2 = 5;        // Radius of curvature of the concave surface, m
lambda = 6600e-010; // Wavelength of light used, cm
n = 15;      // Order of Newton ring
// As D_n^2*(1/R1-1/R2) = 4*n*lambda, solving for D_n
D_15 = sqrt(4*n*lambda/(1/R1-1/R2));      // Diameter of 15th dark ring, cm

printf("\nThe diameter of %dth dark ring = %4.2e m", n, D_15);

// Result
// The diameter of 15th dark ring = 2.81e-002 m
// The answer is given wrong in the textbook (the square root is not solved)
```

Scilab code Exa 2.64 Diameter of fifteenth dark ring due to first wavelength

```
// Scilab Code Ex2.64:: Page-2.49(2009)
clc; clear;
lambda1 = 6000e-008;    // First visible wavelength, cm
lambda2 = 4500e-008;    // Second visible wavelength, cm
R = 120;          // Radius of curvature of the lens, cm
// As diameter of nth dark ring due to lambda1 is
// D_n^2 = 4*n*R*lambda1 and D_nplus1^ = 4*(n+1)*R*lambda2, so that D_n^2 = D_nplus1^2 gives
```

```
n = lambda2/(lambda1-lambda2);        // Order of
    interference for dark fringes
printf("\nThe value of n = %d", n);
n = 15;       // Order of interference fringe
D_n = sqrt(4*n*R*lambda1);        // Diameter of nth
    dark ring due to lambda1
printf("\nThe diameter of 15th dark ring due to
    wavelength of %4d angstrom = %4.2f cm", lambda1/1
    e-008, D_n);

// Result
// The value of n = 3
// The diameter of 15th dark ring due to wavelength
    of 6000 angstrom = 0.66 cm
```

Scilab code Exa 2.65 Refractive index of the liquid filled into container

```
// Scilab Code Ex2.65:: Page-2.49(2009)
clc; clear;
lambda = 5896e-008;     // Wavelength of light used,
    cm
R = 100;         // Radius of curvature of the lens,
    cm
D10 = 0.4;        // Diametre of 10th dark ring, cm
n = 10;       // Order of Newton ring
// As for a dark ring, 2*mu*t = n*lambda and 2*t = (
    D10/2)^2/R, solving for mu
mu = 4*n*lambda*R/D10^2;     // Refractive index of
    the liquid filled into container

printf("\nThe refractive index of the liquid filled
    into container = %4.2f", mu);

// Result
// The refractive index of the liquid filled into
```

```
container = 1.47
```

Scilab code Exa 2.67 Refractive index of the liquid

```
// Scilab Code Ex2.67:: Page-2.50(2009)
clc; clear;
Dn = 1.8;          // Diameter of 15th dark ring, cm
Dn_prime = 1.67;     // Diameter of 15th dark ring with liquid, cm
mu = (Dn/Dn_prime)^2;    // Refractive index of the liquid

printf("\nThe refractive index of the liquid = %4.2f", mu);

// Result
// The refractive index of the liquid = 1.16
```

Scilab code Exa 2.68 Diameter of eighteenth dark ring

```
// Scilab Code Ex2.68:: Page-2.51(2009)
clc; clear;
R = 1;        // For simplicity assume radius of curvature to be unity, cm
D8 = 0.45;         // Diameter of 8th dark ring, cm
D15 = 0.81;       // Diameter of 15th dark ring, cm
n = 8;            // Order of 8th Newton ring
p = 7;            // Order of 7th Newton ring after 8th ring
lambda = (D15^2-D8^2)/(4*p*R);       // Wavelength of light used, cm
// As D18^2-D15^2 = 4*p*lambda*R
p = 3;        // For 18th and 15th rings
```

```
D18 = sqrt(D15^2+4*p*lambda*R);      // Diameter of 18th ring, cm

printf("\nThe diameter of 18th dark ring = %6.4f cm", D18);

// Result
// The diameter of 18th dark ring = 0.9222 cm
```

Scilab code Exa 2.69 Wavelength of light used to illuminate plano convex lens in Newton rings experiment

```
// Scilab Code Ex2.69:: Page-2.51(2009)
clc; clear;
R = 100;        // Radius of curvature of plano-convex lens, cm
D15 = 0.590;      // Diameter of 15th dark ring, cm
D5 = 0.336;      // Diameter of 5th dark ring, cm
p = 10;           // Order of 10th Newton ring after 5th ring
lambda = (D15^2-D5^2)/(4*p*R);       // Wavelength of light used, cm

printf("\nThe wavelength of light used = %4.0f angstrom", lambda/1e-008);

// Result
// The wavelength of light used = 5880 angstrom
```

Scilab code Exa 2.70 Wavelength of monochromatic light used in Michelson Interferometer

```
// Scilab Code Ex2.70:: Page-2.57(2009)
```

```
clc; clear;
N = 250;          // Number of fringes crossing the field of view
delta_x = 0.0595e-01;    // Displacement in movable mirror, cm
// As N*lambda/2 = delta_x, solving for lambda
lambda = 2*delta_x/N;        // Wavelength of light used, cm

printf("\nThe wavelength of monochromatic light used = %4.0f ansgtrom", lambda/1e-008);

// Result
// The wavelength of monochromatic light used = 4760 ansgtrom
// Answer is given wrong in the textbook
```

Scilab code Exa 2.71 Number of fringes that passes across the cross wire of telescope

```
// Scilab Code Ex2.71:: Page-2.58(2009)
clc; clear;
delta_x = 0.02559e-01;    // Displacement in movable mirror, cm
lambda = 5890e-008;        // Wavelength of light used, cm
// As N*lambda/2 = delta_x, solving for N
N = 2*delta_x/lambda;        // Number of fringes crossing the field of view

printf("\nThe number of fringes that passes across the cross wire of telescope = %2d", ceil(N));

// Result
// The number of fringes that passes across the
```

```
cross wire of telescope = 87
```

Scilab code Exa 2.72 Distance between two successive positions of movable mirror

```
// Scilab Code Ex2.72:: Page-2.58(2009)
clc; clear;
lambda1 = 5890e-008;     // Wavelength corresponding to the D1 line, cm
lambda2 = 5896e-008;     // Wavelength corresponding to the D2 line, cm
delta_lambda = lambda2 - lambda1;    // Difference in the wavelengths, cm
// As delta_lambda = lambda1*lambda2/(2*x), solving for x
x = lambda1*lambda2/(2*(lambda2-lambda1));   // Distance between two successive positions of movable mirror

printf("\nThe distance between two successive positions of movable mirror = %3.1e cm", x);

// Result
// The distance between two successive positions of movable mirror = 2.9e-002
```

Scilab code Exa 2.73 Thickness of the transparent glass film

```
// Scilab Code Ex2.73:: Page-2.58(2009)
clc; clear;
N = 550;          // Number of fringes crossing the field of view
```

```
lambda = 5500e-008;     // Wavelength of light used, cm
mu = 1.5;        // Refractive index of the glass slab
// As 2*(mu-1)*t = N*lambda, solving for t
t = N*lambda/(2*(mu-1));        // Thickness of the transparent glass film

printf("\nThe distance between two successive positions of movable mirror = %3.1e cm", t);

// Result
// The distance between two successive positions of movable mirror = 3.0e-002 cm
```

Chapter 3

Diffraction

Scilab code Exa 3.1 Position of the screen so that light is focused on the brightest spot

```
// Scilab Code Ex3.1:: Page-3.9 (2009)
clc; clear;
lambda = 5890e-008;  // Wavelength of light used, cm
r1 = 0.2;   // Radius of first ring of zone plate,
    cm
n = 1;  // Order of zone plate
f1 = r1^2/(n*lambda);   // Position of the screen so
     that light is focused on the brightest spot, cm

printf("\nThe position of the screen so that light
    is focused on the brightest spot = %3.1e cm",
    lambda);

// Result
// The position of the screen so that light is
    focused on the brightest spot = 5.9e-005 cm
```

Scilab code Exa 3.2 Zone plate with a point source of light on the axis

```
// Scilab Code Ex3.2:: Page-3.9 (2009)
clc; clear;
v1 = 36;      // Position of the strongest image from
    the zone plate, cm
v2 = 9;      // Position of the next image from the
    zone plate, cm
lambda = 5890e-008;  // Wavelength of light used, cm
r1 = 1;    // For simplicity assume radius of first
    ring of zone plate to be unity, cm
n = 1;  // Order of zone plate
// As 1/v1-1/u = n*lambda/r1^2 = 1/3*(1/v2-1/u),
    solving for u
u = 2/(3/36-1/9);         // Distance of the zone
    plate from source, cm
// As 1/v-1/u = n*lambda/r1^2, solving for r1
r1 = sqrt(lambda/(1/v1-1/abs(u)));        // Radius
    of first zone, cm
f1 = r1^2/(n*lambda);   // Principal focal length,
    cm

printf("\nThe distance of the zone plate from source
    = %2d cm", u);
printf("\nThe radius of first zone = %3.1e cm", r1);
printf("\nThe principal focal length = %4.1f cm", f1
    );

// Result
// The distance of the zone plate from source = -72
    cm
// The radius of first zone = 6.5e-002 cm
// The principal focal length = 72.0 cm
```

Scilab code Exa 3.3 Position of the first image in a zone plate

```
// Scilab Code Ex3.3:: Page-3.10 (2009)
```

```
clc; clear;
lambda = 5500e-010;  // Wavelength of light used, cm
u = -4;      // Distance of the zone plate from source, cm
D = 3.7e-003;    // Diameter of central zone of zone plate, cm
r = D/2;     // Radius of central zone of zone plate, cm
n = 1;  // Order of zone plate
f1 = r^2/(n*lambda);         // Principal focal length, cm
v1 = 36;      // Position of the strongest image from the zone plate, cm
v2 = 9;      // Position of the next image from the zone plate, cm
// As 1/v - 1/u = 1/f, solving for v
v = 1/(1/f1+1/u);         // Position of the first image in a zone plate, cm

printf("\nThe position of the first image in a zone plate = %2d cm", floor(v));

// Result
// The position of the first image in a zone plate = -12 cm
```

Scilab code Exa 3.4 Principal focal length of zone plate

```
// Scilab Code Ex3.4:: Page-3.11 (2009)
clc; clear;
lambda = 1;      // For simplicity assume wavelength of light used to be unity, unit
R = 150;         // Radius of curvature of the curved surface, cm
r1 = sqrt(lambda*R);     // For Newton's ring, cm
```

```
f1 = r1^2/lambda;        // Principal focal length of
    zone plate, cm

printf("\nThe principal focal length of zone plate =
    %3d cm", f1);

// Result
// The principal focal length of zone plate = 150 cm
```

Scilab code Exa 3.5 Half angular width at central maximum in Fraunhoffer diffraction

```
// Scilab Code Ex3.5:: Page-3.22 (2009)
clc; clear;
lambda = 5000e-008;      // Wavelength of light used,
    cm
a = 15e-005;     // Width of the slit, cm
n = 1;       // Order of diffraction
// For a single slit Fraunhofer diffraction, a*sin(
    theta) = n*lambda, solving for theta
theta = asin(n*lambda/a);  // Half angular width at
    central maximum in Fraunhoffer diffraction,
    radian

printf("\nThe half angular width at central maximum
    in Fraunhoffer diffraction = %5.3f rad", theta);

// Result
// The half angular width at central maximum in
    Fraunhoffer diffraction= 0.340 rad
```

Scilab code Exa 3.6 Width of the slit

```
// Scilab Code Ex3.6:: Page-3.23 (2009)
clc; clear;
lambda = 5000e-010;      // Wavelength of light used,
    cm
n = 1;       // Order of diffraction
x = 5e-003;      // Position of first minima on
   either sides of central maximum, m
D = 2.5;     // Distance of screen from the narrow
   slir , m
sin_theta = x/sqrt(x^2+D^2);     // Sine of angle
   theta, rad
// For a single slit Fraunhofer diffraction, a*sin(
   theta) = n*lambda, solving for a
a = n*lambda/sin_theta;      // Width of the slit, m

printf("\nThe Width of the slit = %3.1e m", a);

// Result
// The Width of the slit = 2.5e-004 m
```

Scilab code Exa 3.7 Angular width of central maximum

```
// Scilab Code Ex3.7:: Page-3.23 (2009)
clc; clear;
lambda = 6000e-010;      // Wavelength of light used,
    m
a = 15e-007;     // Width of the slit, m
// For a single slit Fraunhofer diffraction, a*sind(
   theta) = n*lambda, solving for theta
theta = asind(lambda/a); // Half angular width of
   central maximum, degrees

printf("\nThe angular width of central maximum = %2d
    degrees", 2*ceil(theta));

```

```
// Result
// The angular width of central maximum = 48 degrees
```

Scilab code Exa 3.8 Distance between first minima and the next minima from the axis

```
// Scilab Code Ex3.8:: Page-3.23 (2009)
clc; clear;
lambda = 5000e-010;     // Wavelength of light used, m
a = 0.7e-002;     // Width of the slit, m
f = 0.5;          // Focal length of the lens, m
n = 1;       // Order of diffraction
// For minima, a*sind(theta_n) = n*lambda
// Also theta_n = n*lambda/a = x1/f, solving for x1
x1 = f*n*lambda/a;          // Position of first minima, cm
// For secondary maxima, a*sind(theta_n) = (2*n+1)*lambda/2
// Also theta_n = 3*lambda/(2*a) = x2/f, solving for x2
n = 1;       // Order of diffraction for first secondary minima
x2 = 3*f*lambda/(2*a);       // Position of first secondary maxima, cm

printf("\nThe distance between first minima and the next minima from the axis = %4.2e cm", x2-x1);

// Result
// The distance between first minima and the next minima from the axis = 1.79e-005 cm
```

Scilab code Exa 3.9 Width of central maxima in diffraction pattern

```
// Scilab Code Ex3.9:: Page-3.24 (2009)
clc; clear;
lambda = 6600e-008;      // Wavelength of light used, cm
a = 0.018;     // Width of the slit, cm
f = 200;         // Focal length of the lens, cm
n = 1;       // Order for first order diffraction
// As a*sin(theta) = n*lambda, a*theta = n*lambda
// As theta = lambda/a and theta = x/f, solving for x
x = lambda*f/a;      // Half angular width at central maximum, cm

printf("\nThe width of central maximum = %3.1f cm", 2*x);

// Result
// The width of central maximum = 1.5 cm
```

Scilab code Exa 3.10 Slit width in Fraunhoffer single slit experiment

```
// Scilab Code Ex3.10:: Page-3.24 (2009)
clc; clear;
f = 250;         // Focal length of the lens, cm
x = 0.8;         // Half width of central maxima, cm
lambda = 5500e-008;      // Wavelength of light used, cm
// As x = f*lambda/a, solving for a
a = f*lambda/x;      // Slit width in Fraunhofer single slit experiment

printf("\nThe slit width = %5.3f cm", a);

```

```
// Result
// The slit width = 0.017 cm
```

Scilab code Exa 3.11 Half angular width of central maxima

```
// Scilab Code Ex3.11:: Page-3.25 (2009)
clc; clear;
lambda = 5500e-008;     // Wavelength of light used, cm
a = 8.5e-005;     // Width of the slit, cm
n = 1;        // Order of diffraction
// For a single slit Fraunhofer diffraction, a*sind(theta) = n*lambda, solving for theta
theta = asind(n*lambda/a);   // Half angular width at central maximum in Fraunhoffer diffraction, degrees

printf("\nThe half angular width at central maximum in Fraunhoffer diffraction = %4.1f degrees", theta);

// Result
// The half angular width at central maximum in Fraunhoffer diffraction = 40.3 degrees
```

Scilab code Exa 3.12 Wavelength of light used in Fraunhoffer diffraction due to single slit

```
// Scilab Code Ex3.12:: Page-3.25 (2009)
clc; clear;
a = 0.04;        // Slit width, cm
x = 0.5;         // Half width of central maximum, cm
f = 300;         // Focal length of the lens, cm
```

```
// As x = lambda*f/a, solving for lambda
lambda = a*x/f;      // Wavelength of light used in Fraunhoffer diffraction due to single slit, cm

printf("\nThe wavelength of light used in Fraunhoffer diffraction due to a single slit = %4d angstrom", lambda/1e-008);

// Result
// The wavelength of light used in Fraunhoffer diffraction due to a single slit = 6666 angstrom
```

Scilab code Exa 3.13 Width of central maxima from position of first secondary minima

```
// Scilab Code Ex3.13:: Page-3.25 (2009)
clc; clear;
a = 0.045;         // Slit width, cm
lambda = 5500e-008;      // Wavelength of light used, cm
f = 250;         // Focal length of the lens, cm
x = lambda*f/a;      // Position of central maxima, cm

printf("\nThe position of central maxima = %5.3f cm", x);
printf("\nThe width of central maxima from first minima = %5.3f cm", 2*x);

// Result
// The position of central maxima = 0.306 cm
// The width of central maxima from first minima = 0.611 cm
```

Scilab code Exa 3.14 Wavelength of monochromatic light used in illuminating a slit

```
// Scilab Code Ex3.14:: Page-3.26 (2009)
clc; clear;
a = 0.025;        // Slit width, cm
n = 2;           // Order of diffraction
f = 400;         // Focal length of the lens, cm
x = 2.1;         // Position of central maxima, cm
// As theta = n*lambda/a and theta = x/f, solving for lambda
lambda = x*a/(n*f);      // Wavelength of light used, cm
printf("\nThe wavelength of light used = %4d angstrom", lambda/1e-008);

// Result
// The wavelength of light used = 6562 angstrom
```

Scilab code Exa 3.15 Distance between second dark and next bright fringe on the axes

```
// Scilab Code Ex3.15:: Page-3.26 (2009)
clc; clear;
a = 0.25;        // Slit width, cm
lambda = 5890e-008;      // Wavelength of light, cm
f = 80;          // Focal length of the lens, cm
n = 2;           // Order of diffraction
// As for minima, theta = n*lambda/a and theta = x/f, solving for x
x2 = 2*lambda*f/a;        // Position of 2nd dark fringe, cm
```

```
// As for maxima, theta = (2*n+1)*lambda/(2*a) and theta = x/f, solving for x
x2_prime = 5*lambda*f/(2*a);        // Position of 2nd bright fringe, cm
delta_x = x2_prime-x2;       // Distance between 2nd dark and next bright, cm
printf("\nThe distance between 2nd dark and next bright fringe = %4.2e cm", delta_x);

// Result
// The distance between 2nd dark and next bright fringe = 9.42e-003 cm
```

Scilab code Exa 3.16 Width of the slit from first order diffraction

```
// Scilab Code Ex3.16:: Page-3.27 (2009)
clc; clear;
lambda = 5500e-008;      // Wavelength of light used, cm
x = 3.9e-001;         // Half width of central maximum, cm
f = 220;          // Focal length of the lens, cm
n = 1;        // Order for first order diffraction
// As a*sin(theta) = n*lambda, a*theta = n*lambda
// As theta = lambda/a and theta = x/f, solving for a
a = lambda*f/x;      // Half angular width at central maximum, cm

printf("\nThe width of the slit = %3.1e cm", a);

// Result
// The width of the slit = 3.1e-002 cm
```

Scilab code Exa 3.18 Fraunhoffer diffraction due to double slits

```
// Scilab Code Ex3.18:: Page-3.30 (2009)
clc; clear;
a = 0.019e-003;      // Width of each slit , m
b = 2.0e-004;        // Width of opacity between two slits , m
lambda = 5000e-010; // Wavelengh of light used, m
D = 0.6;          // Distance between slit and the screen , m
// As angular separation , theta = x/D = lambda/(a+b), solving for x
x = D*lambda/(a+b);      // Fringe spacing on the screen , m
// As half angular separation , theta1 = x1/D = lambda/(2*(a+b)), solving for x1
x1 = D*lambda/(2*(a+b));        // Distance between central maxima and first minima, m

printf("\nThe fringe spacing on the screen = %4.2f mm", x/1e-003);
printf("\nThe distance between central maxima and first minima = %4.2f mm", x1/1e-003);

// Result
// The fringe spacing on the screen = 1.37 mm
// The distance between central maxima and first minima = 0.68 mm
```

Scilab code Exa 3.19 Fringe separation in Fraunhoffer double slit diffraction pattern

```
// Scilab Code Ex3.19:: Page-3.31 (2009)
clc; clear;
f = 150;    // Distance between screen and slit , cm
a = 0.005;  // Slit width, cm
b = 0.06;   // Distance between slits , cm
lambda = 5500e-008;     // Wavelength of light used, cm
// As half angular separation, theta1 = x1/f = lambda/(2*(a+b)), solving for x1
x1 = f*lambda/(2*(a+b));        // Distance between central maxima and first minima, cm
delta_theta = lambda/(2*(a+b)); // Angular separation between two consecutive minima, radians
printf("\nThe distance between central maxima and first minima = %4.2e cm", x1);
printf("\nThe angular separation between two consecutive minima = %3.1e radians", delta_theta);

// Result
// The distance between central maxima and first minima = 6.35e-002 cm
// The angular separation between two consecutive minima = 4.2e-004 radians
```

Scilab code Exa 3.20 Positions of first secondary maxima and minima in double slit diffraction

```
// Scilab Code Ex3.20:: Page-3.32 (2009)
clc; clear;
f = 120;    // Distance between screen and slit , cm
a = 0.019;  // Slit width, cm
b = 0.041;   // Distance between slits , cm
lambda = 6500e-008;     // Wavelength of light used,
```

```
cm
// As theta1 = x1/f = lambda/(2*(a+b)), solving for x1
x1 = f*lambda/(2*(a+b));        // Position of first secondary minima, cm
// As theta2 = x2/f = lambda/(a+b), solving for x2
x2 = f*lambda/(a+b);        // Position of first secondary maxima, cm

printf("\nThe position of first secondary minima = %5.3f cm", x1);
printf("\nThe position of first secondary maxima = %4.2f cm", x2);

// Result
// The position of first secondary minima = 0.065 cm
// The position of first secondary maxima = 0.13 cm
```

Scilab code Exa 3.21 Missing orders of spectra in Fraunhoffer double slit diffraction

```
// Scilab Code Ex3.21:: Page-3.34 (2009)
clc; clear;
a = 0.2;   // Slit width, mm
b = 0.8;    // Distance between slits , mm
p = [1 2 3 4];       // Orders of pth diffraction maxima
// As diffraction of pth diffraction maxima, a*sin(theta)=p*lambda --- (i)
// and that of nth diffraction maxima, (a+b)*sin(theta)=n*lambda --- (ii)
// Dividing (ii) by (i), we have
// (a+b)/a = n/p, solving for n
n = (a+b)/a*p;   // Orders of nth diffraction maxima

```

```
printf("\nThe missing orders of spectra in diffraction maxima, n = %d, %d, %d, %d,...", n(1), n(2), n(3), n(4));


// Result
// The missing orders of spectra in diffraction maxima, n = 5, 10, 15, 20,...
```

Scilab code Exa 3.22 Angles of diffraction for the principal maxima for two lines of sodium

```
// Scilab Code Ex3.22:: Page-3.45 (2009)
clc; clear;
lambda1 = 5890e-008;     // Wavelength of D1 line of Na, cm
lambda2 = 5896e-008;     // Wavelength of D2 line of Na, cm
N = 3000/0.5;            // No. of lines per cm of grating, lines/cm
a_plus_b = 1/N;          // Grating element, cm
n = 1;      // Order of diffraction for principal maxima
// As (a+b)*sin(theta1) = n*lambda, solving for theta1
theta1 = asind(n*lambda1/(a_plus_b)); // Angle of diffraction for the principal maxima of D1 line, degrees
theta2 = asind(n*lambda2/(a_plus_b)); // Angle of diffraction for the principal maxima of D2 line, degrees
printf("\nThe angle of diffraction for the principal maxima of D1 line = %5.2f degrees", theta1);
printf("\nThe angle of diffraction for the principal maxima of D2 line = %5.2f degrees", theta2);
```

```
// Result
// The angle of diffraction for the principal maxima of D1 line = 20.70 degrees
// The angle of diffraction for the principal maxima of D2 line = 20.72 degrees
```

Scilab code Exa 3.23 Highest order spectrum which can be seen in monochromatic light

```
// Scilab Code Ex3.23:: Page-3.45 (2009)
clc; clear;
lambda = 5500e-008;     // Wavelength of light used, cm
N = 15000;              // No. of lines per inch of grating, lines/inch
a_plus_b = 2.54/N;          // Grating element, cm
n = 1;        // Order of diffraction for principal maxima
// As (a+b)*sin(theta_n) = n*lambda and for maximum possible order of spectra sin(theta_n) = 1
// So (a+b) = n*lambda, solving for n
n = (a_plus_b)/lambda;    // The highest order spectrum which can be seen in monochromatic light

printf("\nThe highest order spectrum which can be seen in monochromatic light = %d", n);

// Result
// The highest order spectrum which can be seen in monochromatic light = 3
```

Scilab code Exa 3.24 Angle of separation in second order of diffraction spectrum

```
// Scilab Code Ex3.24: : Page-3.46 (2009)
clc; clear;
lambda1 = 5890e-008;     // Wavelength of D1 line, cm
lambda2 = 5896e-008;     // Wavelength of D2 line, cm
N = 15000;               // No. of lines per inch of grating, lines/inch
a_plus_b = 2.54/N;           // Grating element, cm
n = 2;       // Order of diffraction for secondary maxima
// As (a+b)*sin(theta_n) = n*lambda, solving for theta1 and theta2
theta1 = asind(n*lambda1/a_plus_b);      // Direction of secondary maxima with lambda1, degrees
theta2 = asind(n*lambda2/a_plus_b);      // Direction of secondary maxima with lambda2, degrees

printf("\nThe angle of separation in second order diffraction spectrum = %3.1f degrees", theta2-theta1);

// Result
// The angle of separation in second order diffraction spectrum = 0.1 degrees
```

Scilab code Exa 3.25 Separation of two lines in first order spectrum

```
// Scilab Code Ex3.25:: Page-3.46 (2009)
clc; clear;
lambda1 = 5500e-008;     // First wavelength, cm
lambda2 = 3700e-008;     // Second wavelength, cm
N = 15000;               // No. of lines per inch of grating, lines/inch
```

```
a_plus_b = 2.54/N;           // Grating element, cm
f = 120;         // Focal length of the lens, cm
n = 1;       // Order of diffraction for principal maxima
// As (a+b)*sin(theta_n) = n*lambda, solving for theta1 and theta2
theta1 = asind(n*lambda1/a_plus_b);     // Direction of principal maxima with lambda1, degrees
theta2 = asind(n*lambda2/a_plus_b);     // Direction of principal maxima with lambda2, degrees
// As tand(theta) = x/f, solving for x1 - x2 = dx
dx = f*(tand(theta1)-tand(theta2));     // Linear separation of two lines in first order spectrum, cm

printf("\nThe linear separation of two lines in first order spectrum = %5.2f cm", dx);

// Result
// The linear separation of two lines in first order spectrum = 14.34 cm
```

Scilab code Exa 3.26 Difference in the deviation in the first and third order spectra

```
// Scilab Code Ex3.26:: Page-3.47 (2009)
clc; clear;
lambda = 5000e-008;     // Wavelength of light used, cm
N = 5000;             // No. of lines per cm of grating, lines/cm
a_plus_b = 1/N;           // Grating element, cm
n = 1;       // Order of diffraction for first order spectra
// As (a+b)*sin(theta_n) = n*lambda, solving for
```

```
theta for first and third orders
theta1 = asind(n*lambda/a_plus_b);      // Direction of principal maxima with lambda1, degrees
n = 3;       // Order of diffraction for third order spectra
theta3 = asind(n*lambda/a_plus_b);      // Direction of principal maxima with lambda2, degrees
delta_theta = theta3 - theta1;  // Angular separation in the first and third order spectra,

printf("\nThe difference in the deviation in the first and third order spectra = %4.1f degrees", delta_theta);

// Result
// The difference in the deviation in the first and third order spectra = 34.1 degrees
```

Scilab code Exa 3.27 Order of diffraction for the given grating element and wavelength of light

```
// Scilab Code Ex3.27:: Page-3.48 (2009)
clc; clear;
lambda = 6500e-008;     // Wavelength of light used, cm
N = 10000;              // No. of lines per cm of grating, lines/cm
a_plus_b = 1/N;           // Grating element, cm
theta_n = 90;    // Direction for maximum possible orders, degrees
// As (a+b)*sin(theta_n) = n*lambda, solving for theta for n
n = a_plus_b*sind(theta_n)/lambda;   // Order of diffraction for

```

```
printf("\nThe order of diffraction for the given grating element and wavelength of light = %d", n);

// Result
// The order of diffraction for the given grating element and wavelength of light = 1
```

Scilab code Exa 3.28 Number of lines ruled on the grating surface

```
// Scilab Code Ex3.28:: Page-3.48 (2009)
clc; clear;
lambda1 = 6500e-008;     // Wavelength of first line, cm
lambda2 = 4500e-008;     // Wavelength of scecond line, cm
theta1 = 18;         // Direction of lower order, degrees
theta2 = 18;         // Direction of higher order, degrees
// As (a+b)*sin(theta1) = n*lambda1 and (a+b)*sin(theta2) = (n+1)*lambda2, solving for n
n = lambda2/(lambda1 - lambda2);    // Order of diffraction for first wavelength
// As a_plus_b = n*lambda1/sind(theta1), solving for a_plus_b
a_plus_b = ceil(n)*lambda1/sind(theta1);  // Grating element, cm
N = 1/a_plus_b;     // No. of lines on the grating surface, lines/cm

printf("\nThe number of lines ruled on the grating surface = %4d lines/cm", N);

// Result
```

```
// The number of lines ruled on the grating surface
   = 1584 lines/cm
```

Scilab code Exa 3.29 Angles at which first and second order maxima are observed

```
// Scilab Code Ex3.29:: Page-3.48 (2009)
clc; clear;
lambda = 6328e-008;     // Wavelength of He-Laser, cm
a_plus_b = 1/6000;      // Grating element, cm
n = 1;        // First order of diffraction for given
   wavelength
// As (a+b)*sin(theta1) = n*lambda, solving for
   theta1
theta1 = asind(n*lambda/a_plus_b);      // Angle at
   which first order maximum is observed, degrees
n = 2;        // second order of diffraction for given
    wavelength
theta2 = asind(n*lambda/a_plus_b);      // Angle at
   which second order maximum is observed, degrees

printf("\nThe angle at which first order maximum is
   observed = %4.1f degrees", theta1);
printf("\nThe angle at which second order maximum is
    observed = %4.1f degrees", theta2);

// Result
// The angle at which first order maximum is
   observed = 22.3 degrees
// The angle at which second order maximum is
   observed = 49.4 degrees
```

Scilab code Exa 3.30 Least width of plane transmission grating

```
// Scilab Code Ex3.30:: Page-3.49 (2009)
clc; clear;
lambda1 = 5890e-008;     // Wavelength of D1 line of Na, cm
lambda2 = 5896e-008;     // Wavelength of D2 line of Na, cm
d_lambda = lambda2-lambda1;       // Linear separation of two lines just seen as separate, cm
P = 500;          // Number of lines per cm on grating, lines/cm
n = 2;       // Order of diffraction
// As resolving power of grating, lambda/d_lambda = n*N, solving for N
N = lambda1/(d_lambda*n);     // No. of lines required per cm on grating, lines/cm
w = N/P;          // Least width of grating, cm

printf("\nThe least width of plane transmission grating = %5.3f cm", w);

// Result
// The least width of plane transmission grating = 0.982 cm
```

Scilab code Exa 3.31 Minimum grating width required to resolve two wavelengths

```
// Scilab Code Ex3.31:: Page-3.49 (2009)
clc; clear;
theta1 = 18;          // Direction at which first spectral line appears, degrees
theta2 = 18+5/(60*60);  // Direction at which second spectral line appears, degrees
d_theta = (theta2-theta1)*%pi/180;        // Angular separation of two spectral lines, radians
```

```
d_lambda = 50e-010;       // Linear separation of two spectral lines just seen as separate, cm
DP = d_theta/d_lambda;       // Dispersive power of grating
n = 1;       // Order of diffraction
// As dispersive power of grating d_theta/d_lambda = DP = n/((a_plus_b)*cosd(theta1)), solving for a_plus_b
a_plus_b = n/(DP*cosd(theta1));      // Grating element, cm
// But a_plus_b*sind(theta1)=n*lambda1, solving for lambda1
lambda1 = a_plus_b*sind(theta1)/n;       // Wavelength of first spectral line, cm
lambda2 = lambda1+d_lambda/1e-002;            // Wavelength of second spectral line, cm
// As resolving power of grating, lambda/d_lambda = n*N, solving for N
N = lambda1/(d_lambda*n);     // No. of lines required per cm on grating
w = N*a_plus_b;      // Minimum grating width required to resolve two wavelengths, cm

printf("\nThe wavelength of first spectral line = %4.0f angstrom", lambda1/1e-008);
printf("\nThe wavelength of second spectral line = %4.0f angstrom", lambda2/1e-008);
printf("\nThe minimum grating width required to resolve two wavelengths = %3.1f cm", w);

// Result
// The wavelength of first spectral line = 6702 angstrom
// The wavelength of second spectral line = 6752 angstrom
// The minimum grating width required to resolve two wavelengths = 2.9 cm
```

Scilab code Exa 3.32 Angle of diffraction for maxima in first order

```
// Scilab Code Ex3.32:: Page-3.50 (2009)
clc; clear;
// Function to convert theta into degree-minute
function[degre, minute]=deg_2_degminsec(theta)
    degre = floor(theta);
    minute = (theta-floor(theta))*60;
endfunction

N = 15000;      // No. of lines on the grating per
    inch, lines/inch
a_plus_b = 2.54/N;      // Grating element, cm
lambda = 6000e-008;     // Wavelength of light used,
     cm
n = 1;      // Order of diffraction spectra
// But a_plus_b*sind(theta)=n*lambda, solving for
    theta
theta = asind(n*lambda/a_plus_b);       // Direction
     in which first order spectra is seen, degrees
[deg, mint] = deg_2_degminsec(theta);
printf("\nThe angle of diffraction for maxima in
    first order = %2d degrees %2d min", deg, mint);

// Result
// The angle of diffraction for maxima in first
    order = 20 degrees 45 min
```

Scilab code Exa 3.33 Wavelength of light used in obtaining second order diffraction maximum

```
// Scilab Code Ex3.33:: Page-3.50 (2009)
```

```
clc; clear;
N = 12000;        // No. of lines on the grating per inch, lines/inch
a_plus_b = 2.54/N;        // Grating element, cm
n = 2;        // Order of diffraction spectra
theta = 39; // Angle of diffraction for maxima in second order, degrees
// But a_plus_b*sind(theta)=n*lambda, solving for lambda
lambda = a_plus_b*sind(theta)/n;        // Wavelength of light used, cm

printf("\nThe wavelength of light used in obtaining second order diffraction maximum = %4d angstrom", lambda/1e-008);

// Result
// The wavelength of light used in obtaining second order diffraction maximum = 6660 angstrom
```

Scilab code Exa 3.34 Number of visible orders using diffraction grating

```
// Scilab Code Ex3.34:: Page-3.51 (2009)
clc; clear;
lambda = 5890e-008;        // Wavelength of light used, cm
N = 6000;        // No. of lines on the grating per inch, lines/inch
a_plus_b = 2.54/N;        // Grating element, cm
theta_max = 90;        // Direction of maxima for maximum possible orders
// But a_plus_b*sind(theta_max)=n*lambda, solving for n
n = a_plus_b*sind(theta_max)/lambda;        // Number of visible orders
```

```
printf("\nThe number of visible orders using diffraction grating = %d", n);

// Result
// The number of visible orders using diffraction grating = 7
```

Scilab code Exa 3.35 Distance between two wavelengths seen as separate

```
// Scilab Code Ex3.35:: Page-3.51 (2009)
clc; clear;
lambda = 5500e-008;      // Mean of two wavelengths, cm
theta = 35;          // Angle of diffraction for maxima in second order
d_theta = 0.15;      // Angular separation between two neighbouring wavelengths, radians
d_lambda = lambda*cotd(theta)*d_theta;   // Distance between two wavelengths seen as separate, cm

printf("\nThe distance between two wavelengths seen as separate = %d angstrom", d_lambda/1e-008);

// Result
// The distance between two wavelengths seen as separate = 1178 angstrom
```

Scilab code Exa 3.36 Number of lines per cm on grating surface

```
// Scilab Code Ex3.36:: Page-3.51 (2009)
clc; clear;
```

```
lambda1 = 5500e-008;      // First wavelength of light , cm
lambda2 = 4500e-008;      // Second wavelength of light , cm
theta = 45;      // Angle of diffraction for lower order , degrees
n = lambda2/(lambda1-lambda2);   // Lower order of diffraction
// But a_plus_b*sind(theta)=n*lambda, solving for a_plus_b
a_plus_b = floor(n)*lambda1/sind(theta);        // Grating element , cm
N = 1/a_plus_b;      // No. of lines per cm on grating surface , lines/cm

printf("\nThe number of lines per cm on grating surface = %4d lines/cm", ceil(N));

// Result
// The number of lines per cm on grating surface = 3215 lines/cm
```

Scilab code Exa 3.37 Total number of lines on grating surface

```
// Scilab Code Ex3.37:: Page-3.52 (2009)
clc; clear;
lambda = 6500e-008;      // Wavelength of light used , cm
theta = 19.5;    // Angle of diffraction for maxima in first order , degrees
l = 3.5;     // Length of the grating , cm
n = 1;       // Order of diffraction
// But a_plus_b*sind(theta)=n*lambda, solving for a_plus_b
a_plus_b = n*lambda/sind(theta);        // Grating
```

```
element, cm
N = 1/a_plus_b;       // No. of lines per cm on grating surface, lines/cm
N_total = l*N;        // Total number of lines on grating surface

printf("\nThe total number of lines on grating surface = %5d", N_total);

// Result
// The total number of lines on grating surface = 17974
```

Scilab code Exa 3.38 Angular separation between the sodium D1 and D2 lines

```
// Scilab Code EX3.38:: Page-3.52 (2009)
clc;clear;
function [mint, secnd]=degmin(theta)
    mint = (theta-floor(theta))*60;
    secnd = (mint-floor(mint))*60
endfunction
lambda_D1 = 5890e-008;   // Wavelength of sodium D1 line, cm
lambda_D2 = 5896e-008;   // Wavelength of sodium D2 line, cm
n = 2;   // Order of diffraction
N = 6500;    // Number of lines per cm on grating, lines/cm
a_plus_b = 1/6500;   // Grating element, cm
// As a_plus_b*sin(theta1)=n*lambda1, solving for theta1
theta1 = asind(n*lambda_D1/a_plus_b);
// As a_plus_b*sin(theta2)=n*lambda2, solving for theta1
```

```
theta2 = asind(n*lambda_D2/a_plus_b);
d_theta = theta2-theta1;     // Angular separation between the sodium D1 and D2 lines, degrees
[mint, secnd] = degmin(d_theta);  // Call deg_2_degmin function

printf("\nThe angular separation between the sodium D1 and D2 lines = %d minutes %d seconds", mint, secnd);

// Result
// The angular separation between the sodium D1 and D2 lines = 4 minutes 10 seconds
// Since theta1 and theta2 are rounded off in the textbook, therefore the answer is mismatching.
```

Scilab code Exa 3.39 Minimum number of lines in a grating

```
// Scilab Code EX3.39:: Page-3.55 (2009)
clc;clear;
lambda1 = 5890e-008;   // Wavelength of sodium D1 line, cm
lambda2 = 5896e-008;   // Wavelength of sodium D2 line, cm
d_lambda = lambda2-lambda1; // Difference in the wavelength of two lines, cm
n = 2;   // Order of diffraction
// As lambda/d_lambda = n*N, solving for N
N = lambda1/(d_lambda*n);  // Minimum number of lines in a grating
printf("\nThe minimum number of lines in a grating = %3d lines", N);

// Result
// The minimum number of lines in a grating = 490
```

lines

Scilab code Exa 3.40 Linear separation of two points on the moon

```
// Scilab Code EX3.40:: Page-3.56 (2009)
clc;clear;
lambda = 5500e-008;  // Wavelength of most sensitive
    color to an eye, cm
a = 400;    // Aperture of the telescope, cm
D = 3.8e+010;   // Distance of the moon from the
    earth, cm
d_theta = 1.22*lambda/a;    // Limit of resolution
    of telescope, radians
// As d_theta = x/D, solving for x
x = d_theta*D;  // Linear separation of two points
    on the moon, cm

printf("\nThe linear separation of two points on the
    moon = %5.2f m", x/1e+002);

// Result
// The linear separation of two points on the moon =
    63.74 m
```

Scilab code Exa 3.41 Minimum required number of lines on the plane transmission grating

```
// Scilab Code EX3.41:: Page-3.56 (2009)
clc;clear;
lambda1 = 5890e-008;  // Wavelength of sodium D1
    line, cm
lambda2 = 5896e-008;  // Wavelength of sodium D2
    line, cm
```

```
d_lambda = lambda2-lambda1; // Wavelength difference
    , cm
n = 2;       // Order of diffraction
// As lambda/d_lambda = n*N, solving for N
N = 1/n*(lambda1+lambda2)/(2*d_lambda);    //
   Minimum required number of lines on the plane
   transmission grating

printf("\nThe minimum required number of lines on
   the plane transmission grating = %3d", N);

// Result
// The minimum required number of lines on the plane
     transmission grating = 491
```

Scilab code Exa 3.42 Number of lines on the plane transmission grating to just resolve the sodium lines

```
// Scilab Code EX3.42:: Page-3.57 (2009)
clc;clear;
lambda1 = 5890e-008;   // Wavelength of sodium D1
   line , cm
lambda2 = 5896e-008;   // Wavelength of sodium D2
   line , cm
d_lambda = lambda2-lambda1; // Wavelength difference
    , cm
w = 2.5;     // Width of the grating , cm
n = 2;       // Order of diffraction
// As lambda/d_lambda = n*N, solving for N
N = 1/n*(lambda1+lambda2)/(2*d_lambda);    //
   Minimum required number of lines on the plane
   transmission grating

printf("\nThe number of lines on the plane
   transmission grating to just resolve the sodium
```

```
    lines = %3d", N/w);

// Result
// The number of lines on the plane transmission grating to just resolve the sodium lines = 196
```

Scilab code Exa 3.43 Minimum width of the grating to resolve the sodium lines in third order

```
// Scilab Code EX3.43:: Page-3.57 (2009)
clc;clear;
lambda1 = 5890e-008;   // Wavelength of sodium D1 line, cm
lambda2 = 5896e-008;   // Wavelength of sodium D2 line, cm
d_lambda = lambda2-lambda1; // Wavelength difference, cm
n = 3;       // Order of diffraction
P = 2500;    // Number of lines per unit length of grating
// As lambda/d_lambda = n*N, solving for N
N = 1/n*(lambda1+lambda2)/(2*d_lambda);    // Total lines on the grating
w = N/P;     // Minimum width of the grating, cm
printf("\nThe minimum width of the grating to resolve the sodium lines in third order = %5.3f cm", w);

// Result
// The minimum width of the grating to resolve the sodium lines in third order = 0.131 cm
```

Scilab code Exa 3.44 Dispersive power and diffraction angle for grating

```
// Scilab Code EX3.44:: Page-3.57 (2009)
clc;clear;
w = 2;       // Width of the grating, cm
P = 4500;    // Total number of lines on the grating
a_plus_b = w/P; // Grating element, cm
lambda1 = 5890e-008;   // Wavelength of sodium D1 line, cm
lambda2 = 5896e-008;   // Wavelength of sodium D2 line, cm
lambda = (lambda1+lambda2)/2;   // Mean wavelength of light used, cm
d_lambda=lambda2-lambda1;    // Difference in wavelengths of D-lines of sodium, cm
n = 2;  // Order of diffraction
// As a_plus_b*sind(theta)=n*lambda, solving for theta
theta = asind(n*lambda/a_plus_b);    // Angle of diffraction, degrees
DP = n/(a_plus_b*cosd(theta));       // Dispersive power of grating
d_theta = DP*d_lambda*180/%pi;   // Angular separation between D-lines, degrees
RP = lambda/d_lambda;    // Required resolving power of grating for sodium lines
N = 2.54/a_plus_b;  // No. of lines per cm on grating, lines/cm
RP_cal = n*N;        // Calculated resolving power of grating

printf("\nThe angle of diffraction for maxima in second order = %6.4f degrees", d_theta);
printf("\nAs %5.3e > %3d, D-lines can be resolved.", RP_cal, RP);

// Result
// The angle of diffraction for maxima in second order = 0.0160 degrees
// As 1.143e+04 > 982, D-lines can be resolved.
```

Scilab code Exa 3.45 Distance between centres of images of the two stars

```
// Scilab Code EX3.45:: Page-3.58 (2009)
clc;clear;
lambda = 5500e-010;   // Wavelength of light used, m
a = 0.01;    // Diameter of objective of telescope, m
f = 3.0;     // Focal length of tlescope objective, m
// For telescope, the limit of resolution,
// theta = x/f = 1.22*lambda/a, solving for x
x = 1.22*lambda/a*f;     // Distance between centres
    of imgaes of the two stars

printf("\nThe distance between centres of imgaes of
    the two stars = %4.2e m", x);

// Result
// The distance between centres of imgaes of the two
     stars = 2.01e-04 m
```

Scilab code Exa 3.46 Aperture of the objective of the microscope

```
// Scilab Code EX3.46:: Page-3.59 (2009)
clc;clear;
lambda = 5461e-008;   // Wavelength of light used, cm
d = 4e-005;       // Separation distance between two
    self-luminous objects, cm
NA = 1.22*lambda/(2*d);      // Numerical aperture of
     microscope, cm

printf("\nThe numerical aperture of the objective of
     the microscopes = %6.4f cm", NA);
```

```

// Result
// The numerical aperture of the objective of the
   microscopes = 0.8328 cm
```

Chapter 4

Polarization

Scilab code Exa 4.1 Refractive index of the material

```
// Scilab Code Ex4.1:: Page-4.5 (2009)
clc; clear;
ip = 60;    // Polarizing angle, degrees
mu = tand(ip);  // Refractive index of the material
   from Brewster's law
printf("\nThe refractive index of the material = %5
   .3f", mu);

// Result
// The refractive index of the material = 1.732
```

Scilab code Exa 4.2 Polarization by reflection

```
// Scilab Code Ex4.2:: Page-4.6 (2009)
clc; clear;
ip = 57;    // Polarizing angle, degrees
mu = tand(ip);  // Refractive index of the material
   from Brewster's law
```

```
printf("\nThe refractive index of the material = %4
    .2f", mu);

// Result
// The refractive index of the material = 1.54
```

Scilab code Exa 4.3 Angle of refraction of the ray

```
// Scilab Code Ex4.3:: Page-4.6 (2009)
clc; clear;
mu = 1.53;  // Refractive index of the material from
     Brewster's law
// As mu = tand(ip), solving for ip
ip = atand(mu);     // Polarizing angle, degrees
// But mu = sind(ip)/sind(r), solving for r
r = asind(sind(ip)/mu);     // Angle of refraction,
    degrees

printf("\nThe angle of refraction of the ray = %4.1f
     degrees", r);

// Result
// The angle of refraction of the ray = 33.2 degrees
```

Scilab code Exa 4.4 Angle of minimum deviation for green light

```
// Scilab Code Ex4.4:: Page-4.6 (2009)
clc; clear;
ip = 60;     // Polarizing angle, degrees
A = 60;     // Angle of equilateral prism, degrees
mu = tand(ip);  // Refractive index of the material
    from Brewster's law
```

```
// For angle of minimum deviation in prism, delta_m,
    refractive index
// mu = sind((A+delta_m)/2)/sind(A/2), solving for
   delta_m
delta_m = 2*asind(mu*sind(A/2))-A;      // Angle of
   minimum deviation, degrees

printf("\nThe angle of minimum deviation for green
   light = %2d degrees", ceil(delta_m));

// Result
// The angle of minimum deviation for green light =
   60 degrees
```

Scilab code Exa 4.5 Polarizing angles of the materials for given refractive indices

```
// Scilab Code Ex4.5:: Page-4.7 (2009)
clc; clear;
mu = [1.33 1.65 1.55];      // Refractive indices of
    the material
// As mu = tand(ip), solving for ip
ip = atand(mu);     // Brewster's law gives
   polarizing angle, degrees
for i =1:1:3
printf("\nmu = %4.2f, ip = %4.1f degrees", mu(i), ip
   (i));
end

// Result
// mu = 1.33, ip = 53.1 degrees
// mu = 1.65, ip = 58.8 degrees
// mu = 1.55, ip = 57.2 degrees
```

Scilab code Exa 4.6 Angle of rotation of analyser

```
// Scilab Code Ex4.6:: Page-4.8 (2009)
clc; clear;
E0 = 1;     // For simplicity assume maximum intensity through polarizer and analyser to be unity, unit
E = 1/6*E0; // One-sixth of the maximum intensity, unit
// From Malus law, E = E0*cosd(theta)^2, solving for theta
theta = acosd(sqrt(E));   // Angle through which analyser should be rotated, degrees
printf("\nThe angle of rotation of analyser = %4.1f degrees", theta);

// Result
// The angle of rotation of analyser = 65.9
```

Scilab code Exa 4.7 Angles of rotation of analyser for given transmitted light intensities

```
// Scilab Code Ex4.7:: Page-4.8 (2009)
clc; clear;
E0 = 1;     // For simplicity assume maximum intensity through polarizer and analyser to be unity, unit
light_fraction = [0.25 0.45 0.65 0.75 0.0];
for i = 1:1:5
E = light_fraction(i)*E0; // Light fraction of the maximum intensity, unit
```

```
// From Malus law, E = E0*cosd(theta)^2, solving for theta
theta = acosd(sqrt(E));    // Angle through which analyser should be rotated, degrees
printf("\nE = %4.2fE0, theta = %4.1f degrees", light_fraction(i), theta);
end

// Result
// E = 0.25E0, theta = 60.0 degrees
// E = 0.45E0, theta = 47.9 degrees
// E = 0.65E0, theta = 36.3 degrees
// E = 0.75E0, theta = 30.0 degrees
// E = 0.00E0, theta = 90.0 degrees
```

Scilab code Exa 4.8 Angle of minimum deviation for green light

```
// Scilab Code Ex4.8:: Page-4.9 (2009)
clc; clear;
ip = 60;          // Polarizing angle, degrees
mu = tand(ip);   // Brewster's law giving refractive index
A = 60;       // Angle of prism, degrees
d = (mu - 1)*A; // Angle of minimum deviation for green light, degrees

printf("\nThe angle of minimum deviation for green light = %5.2f degrees", d);

// Result
// The angle of minimum deviation for green light = 43.92 degrees
```

Scilab code Exa 4.9 Ratio of ordinary to extraordinary ray intensities

```
// Scilab Code Ex4.9:: Page-4.9 (2009)
clc; clear;
theta = 30;      // Angle which the plane of vibration makes with the incident beam, degrees
// As intensity of ordinary and extraordinary ray are
// E_E = A^2*cosd(theta)^2 and E_O = A^2*sind(theta)^2, solving for E_E/E_O
EE_ratio_EO = cotd(30)^2;     // Ratio of ordinary and extraordinary ray intensities

printf("\nThe ratio of ordinary to extraordinary ray intensities = %d", EE_ratio_EO);

// Result
// The ratio of ordinary to extraordinary ray intensities = 3
```

Scilab code Exa 4.10 Thickness of quarter wave plate

```
// Scilab Code Ex4.10:: Page-4.23 (2009)
clc; clear;
mu_o = 1.658;    // Refractive index of ordinary wave
mu_e = 1.486;    // Refractive index of extraordinary wave
lambda = 5893e-008; // Wavelength of light used, m
// As (mu_o - mu_e)*t = lambda/4, solving for t
t = lambda/(4*(mu_o - mu_e));    // Thickness of quarter-wave plate, cm

printf("\nThe thickness of quarter-wave plate = %3.1e cm", t);

```

```
// Result
// The thickness of quarter-wave plate = 8.6e-005 cm
```

Scilab code Exa 4.11 Least thickness of plate for which emergent beam is plane polarised

```
// Scilab Code Ex4.11:: Page-4.23 (2009)
clc; clear;
mu_o = 1.5442;    // Refractive index of ordinary wave
mu_e = 1.5533;    // Refractive index of extraordinary wave
lambda = 5000e-008; // Wavelength of light used, m
// As (mu_o - mu_e)*t = lambda/4, solving for t
t = lambda/(4*(mu_e - mu_o));    // Least thickness of plate for which emergent beam is plane polarised, cm

printf("\nThe least thickness of plate for which emergent beam is plane polarised = %4.2e cm", t);

// Result
// The least thickness of plate for which emergent beam is plane polarised = 1.37e-003 cm
```

Scilab code Exa 4.12 Difference in refractive indices of rays

```
// Scilab Code Ex4.12:: Page-4.23 (2009)
clc; clear;
lambda = 5893e-008; // Wavelength of light used, m
t = 0.005;    // Thickness of the crystal, cm
// As for quarter wave plate, mu_diff*t = (mu_o - mu_e)*t = lambda/4, solving for mu_diff
```

```
mu_diff = lambda/(4*t);      // The difference in refractive indices of rays, cm
printf("\nThe least thickness of plate for which emergent beam is plane polarised = %4.2e cm", mu_diff);

// Result
// The least thickness of plate for which emergent beam is plane polarised = 2.95e-003 cm
```

Scilab code Exa 4.13 The thickness of a half wave plate

```
// Scilab Code Ex4.13:: Page-4.24 (2009)
clc; clear;
mu_o = 1.54;    // Refractive index of ordinary wave
mu_e = 1.45;    // Refractive index of extraordinary wave
lambda = 5500e-008; // Wavelength of light used, m
// As for a half wave plate, (mu_o - mu_e)*t = lambda/4, solving for t
t = lambda/(2*(mu_o - mu_e));    // The thickness of a half wave plate for wavelength, cm

printf("\nThe thickness of a half wave plate for wavelength = %4.2e cm", t);

// Result
// The thickness of a half wave plate for wavelength = 3.06e-004 cm
```

Scilab code Exa 4.14 The thickness of a quarter wave plate

```
// Scilab Code Ex4.14:: Page-4.24 (2009)
```

```
clc; clear;
mu_o = 1.55;    // Refractive index of ordinary wave
mu_e = 1.52;    // Refractive index of extraordinary
    wave
lambda = 5500e-008; // Wavelength of light used, m
// As for a half wave plate, (mu_o - mu_e)*t =
    lambda/4, solving for t
t = lambda/(4*(mu_o - mu_e));    // The thickness of
    a quarter wave plate for wavelength, cm

printf("\nThe thickness of a quarter wave plate for
    wavelength = %4.2e cm", t);

// Result
// The thickness of a quarter wave plate for
    wavelength = 4.58e-004 cm
```

Scilab code Exa 4.15 The thickness of a half wave plate quartz

```
// Scilab Code Ex4.15:: Page-4.24 (2009)
clc; clear;
mu_o = 1.51;    // Refractive index of ordinary wave
mu_e = 1.55;    // Refractive index of extraordinary
    wave
lambda = 6000e-008; // Wavelength of light used, m
// As for a half wave plate, (mu_o - mu_e)*t =
    lambda/4, solving for t
t = lambda/(2*(mu_e - mu_o));    // The thickness of
    a quarter wave plate for wavelength, cm

printf("\nThe thickness of a half wave plate quartz
    = %4.2e cm", t);

// Result
// The thickness of a half wave plate quartz = 7.50e
```

```
-004 cm
```

Scilab code Exa 4.16 Difference between refractive indices

```
// Scilab Code Ex4.16:: Page-4.24 (2009)
clc; clear;
lambda = 5890e-008; // Wavelength of light used, m
t = 7.5e-004;    // Thickness of the crystal, cm
// As for quarter wave plate, mu_diff*t = (mu_e - mu_o)*t = lambda/4, solving for mu_diff
mu_diff = lambda/(4*t);      // The difference in refractive indices of rays, cm
printf("\nThe difference between refractive indices = %6.4f cm", mu_diff);

// Result
// The difference between refractive indices = 0.0196 cm
```

Scilab code Exa 4.17 Specific rotation of superposition

```
// Scilab Code Ex4.17:: Page-4.34 (2009)
clc; clear;
theta = 15.2;    // Angle through which plane of polarization is rotated, degrees
c = 0.2;     // Concentration of sugar, g/cc
l = 25;      // Length of sugar, cm
S = 10*theta/(l*c);      // Specific rotation of superposition, degrees

printf("\nThe specific rotation of superposition = %4.1f cm", S);

```

```
// Result
// The specific rotation of superposition = 30.4 cm
```

Scilab code Exa 4.18 Strength of sugar solution

```
// Scilab Code Ex4.18: : Page-4.34 (2009)
clc; clear;
theta = 15.2;    // Angle through which plane of polarization is rotated, degrees
S = 65;      // Specific rotation of sugar solution, degrees
l = 15;      // Length of sugar, cm
// As S = 10*theta/(l*c), solving for c
c = 10*theta/(l*S);      // Concentration of sugar, g/cc

printf("\nThe strength of sugar solution = %4.2f g/cc", c);

// Result
// The strength of sugar solution = 0.16 g/cc
```

Scilab code Exa 4.19 Quantity of sugar contained in the tube in the form of solution

```
// Scilab Code Ex4.19:: Page-4.34 (2009)
clc; clear;
theta = 15;    // Angle through which plane of polarization is rotated, degrees
S = 69;      // Specific rotation of sugar solution, degrees
l = 10;      // Length of sugar, cm
V = 50;      // Volume of the tube, cc
```

```
// As S = 10*theta/(l*c), solving for c
c = 10*theta/(l*S);      // Concentration of sugar, g/cc
M = c*V;     // Mass of sugar in solution, g

printf("\nThe quantity of sugar contained in the tube in the form of solution = %5.2f g", M);

// Result
// The quantity of sugar contained in the tube in the form of solution = 10.87 g
```

Scilab code Exa 4.20 Specific rotation of sugar solution from the given data

```
// Scilab Code Ex4.20:: Page-4.35 (2009)
clc; clear;
theta = 8;    // Angle through which plane of polarization is rotated, degrees
M = 10;       // Amount of sugar, g
l = 14;       // Length of the tube, cm
V = 44;       // Volume of sugar solution, cc
c = M/V;      // Concentration of sugar, g/cc
S = 10*theta/(l*c); // Specific rotation of sugar solution from the given data, degrees

printf("\nThe specific rotation of sugar solution from the given data = %4.1f degrees", S);

// Result
// The specific rotation of sugar solution from the given data = 25.1 degrees
```

Scilab code Exa 4.21 Angle of rotation of the plane of polarization

```
// Scilab Code Ex4.21:: Page-4.35 (2009)
clc; clear;
m = 15;      // Amount of sugar, g
S = 66; // Specific rotation of sugar solution from the given data, degrees
l = 20;      // Length of the tube, cm
V = 100;      // Volume of sugar solution, cc
c = m/V;      // Concentration of sugar, g/cc
// As S = 10*theta/(l*c), solving for theta
theta = S*l*c/10;        // Angle of rotation of the plane of polarization, degrees

printf("\nThe angle of rotation of the plane of polarization = %4.1f degrees", theta);

// Result
// The angle of rotation of the plane of polarization = 19.8 degrees
```

Scilab code Exa 4.22 Angle of rotation of the optically active solution

```
// Scilab Code Ex4.22: : Page-4.35 (2009)
clc; clear;
l = 5;      // Length of the tube, dm
m = 50;      // Amount of sugar, g
S = 50; // Specific rotation of sugar solution, degrees
V = 150;      // Volume of sugar solution, cc
c = m/V;      // Concentration of sugar, g/cc
// As S = theta/(l*c), solving for theta
theta = S*l*c;        // Angle of rotation of the optically active solution

```

```
printf("\nThe angle of rotation of the optically active solution = %4.1f degrees", theta);

// Result
// The angle of rotation of the optically active solution = 83.3 degrees
```

Scilab code Exa 4.23 Angle of rotation in a tube of new length

```
// Scilab Code Ex4.23:: Page-4.35 (2009)
clc; clear;
l = 3;      // Length of the tube, dm
theta = 17.0;        // Angle of rotation of the plane of polarization, degrees
c = 1.0;       // For simplicity assume concentration of solution to be unity, g/cc
l_prime = 2.5;    // New length of the tube, dm
c_prime = 1.25*c;  // Concentration of solution with 25 cm length of tube, g/cc
theta_prime = theta*l_prime*c_prime/(l*c);  // Angle of rotation in a tube of new length


printf("\nThe angle of rotation in a tube of new length of %3.1f cm = %4.1f degrees", l_prime, theta_prime);

// Result
// The angle of rotation in a tube of new length of 2.5 cm = 17.7 degrees
```

Scilab code Exa 4.24 Mass of sugar in the solution contained in the tube

```
// Scilab Code Ex4.24:: Page-4.36 (2009)
clc; clear;
l = 17;       // Length of the tube, cm
V = 37;       // Volume of sugar solution, cc
theta = 15;         // Angle of rotation of the plane of polarization, degrees
S = 68;           // Specific rotation of sugar solution, degrees
// As S = 10*theta/(l*c), solving for c
c = 10*theta/(l*S);      // Concentration of sugar solution, g/cc
m = c*V;          // Mass of sugar in the solution contained in the tube, g

printf("\nThe mass of sugar in the solution contained in the tube = %3.1f g", m);

// Result
// The mass of sugar in the solution contained in the tube = 4.8 g
```

Scilab code Exa 4.25 Percentage purity of the sugar sample

```
// Scilab Code Ex4.25:: Page-4.36 (2009)
clc; clear;
m = 80;         // Mass of sugar in the solution, g
theta = 9.9;         // Angle of rotation of the plane of polarization, degrees
l = 20;       // Length of the tube, cm
S_pure = 66;           // Specific rotation of pure sugar solution, degrees per dm per (g/cc)
c = 0.08;      // Concentration of sugar solution, g/cc
S = 10*theta/(l*c);   // calculated specific rotation of sugar solution, degrees per dm per (g/cc)
```

```
percent_purity = S/S_pure*100;       // Percentage purity of sugar sample, percent

printf("\nThe percentage purity of the sugar sample = %5.2f percent", percent_purity);

// Result
// The percentage purity of the sugar sample = 93.75 percent
```

Scilab code Exa 4.26 Angle of rotation produced by the polarimeter plate

```
// Scilab Code Ex4.26:: Page-4.42 (2009)
clc; clear;
lambda = 6600e-010;      // Wavelength of circularly polarized light, cm
mu_R = 1.53914;          // Refractive index of right-handed circularly polarized light
mu_L = 1.53920;          // Refractive index of left-handed circularly polarized light
t = 0.0005;    // Thickness of polarimeter plate, m
theta = %pi/lambda*(mu_L-mu_R)*t;    // Angle of rotation produced by the polarimeter plate, radian

printf("\nThe angle of rotation produced by the polarimeter plate = %4.2f degrees", theta*180/%pi);

// Result
// The angle of rotation produced by the polarimeter plate = 8.18 degrees
```

Chapter 5

Nuclear Physics

Scilab code Exa 5.1 Mass defect of He

```
// Scilab Code Ex5.1 :: Page-5.2 (2009)
clc;clear;
m_p = 1.007826;      // Mass of a proton, amu
m_n = 1.008665;      // Mass of a neutron, amu
M_He = 4.002604;     // Measured mass of He nucleuc,
    amu
delta_m = 2*m_p+2*m_n - M_He;    // Mass defect of He
    , amu
printf("\nThe mass defect of He = %f amu", delta_m);

// Result
//  The mass defect of He = 0.030378 amu
```

Scilab code Exa 5.3 Maximum energy of proton in a cyclotron

```
// Scilab Code Ex5.3 :: Page-5.16 (2009)
clc;clear;
B = 0.70;         // Magnetic field of cyclotron,
    weber/metre square
```

```
q = 1.6e-019;    // Charge of the proton, C
R = 3;       // Radius of Dee's, m
m = 1.67e-027;    // Mass of the proton, kg
E_max = B^2*q^2*R^2/(2*m);   // Maximum energy of the proton in the cyclotron, joule
printf("\nThe maximum energy of the proton in the cyclotron = %4.2e MeV", E_max/1.6e-013);

// Result
// The maximum energy of the proton in the cyclotron = 2.11e+02 MeV
// The unit has been given wrong in the textbook. It should be MeV instead of eV
```

Scilab code Exa 5.4 Energy of an electron in a betatron

```
// Scilab Code Ex5.4 :: Page-5.20 (2009)
clc;clear;
f = 1e+06;       // Frequency of revolution of electron, Hz
rate_phi_B = 25;     // Rate of change of magnetic flux, wb/s
E = f*rate_phi_B;    // Energy of 'f' revolutios, eV
printf("\nThe energy of the electron in Betatron after %g revolutions = %3.1e eV", f, E);

// Result
// The energy of the electron in Betatron after 1e+06 revolutions = 2.5e+07 eV
```

Scilab code Exa 5.5 Final energy gained by electrons in a betatron

```
// Scilab Code Ex5.5 :: Page-5.20 (2009)
```

```
clc;clear;
e = 1.6e-019;   // Charge on an electron , C
D = 2.0;     // Diameter of the stable orbit in
    betatron , m
R = D/2;     // Radius of the stable orbit in
    betatron , m
B = 0.5;     // Magnetic field of betatron , wb/metre
    square
c = 3e+08;  // final speed of electron in betatron ,
   m/s
E = B*e*R*c;     // Final energy gained by electrons
    in a betatron , eV
printf("\nThe final energy gained by electrons in
    the betatron = %3.1e eV", E/e);

// Result
// The final energy gained by electrons in the
    betatron = 1.5e+08 eV
```

Scilab code Exa 5.6 Energy produced in fission of U235

```
// Scilab Code Ex5.6 :: Page-5.27 (2009)
clc;clear;
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
A = 235;     // Atomic weight of uranium , gm/mol
N_A = 6.023e+026;    // No. of atoms present in 235
    kg of uranium
N = N_A/(A*1000); // No. of nuceli of uranium per
    gram
E = N*200;  // Energy produced by 1 g of U-235, MeV
printf("\nThe energy produced by 1 g of U-235 = %3.1
    e joule", E*e*1e+06);

// Result
// The energy produced by 1 g of U-235 = 8.2e+10
```

```
joule
```

Scilab code Exa 5.7 Power output of nuclear reactor

```
// Scilab Code Ex5.7 :: Page-5.32 (2009)
clc;clear;
A = 235;    // Atomic weight of uranium, gm/mol
N_A = 6.023e+026;   // No. of atoms present in 235 kg of uranium-235
N = N_A*5/A; // No. of nuceli of uranium in 5 kg of U-235
E = N*200;  // Energy released in the fission of 5 kg of U-235, MeV
t = 24*3600;    // Time taken to consume 5 kg of U-235, sec
P = E/t;    // Total power output of the nuclear reactor, MeV per second
printf("\nThe total power output of the nuclear reactor = %4.2e MeV per second", P);

// Result
// The total power output of the nuclear reactor = 2.97e+22 MeV per second
```

Scilab code Exa 5.8 Average current in the GM counter circuit

```
// Scilab Code Ex5.8 :: Page-5.34 (2009)
clc;clear;
e = 1.6e-019;   // Electronic charge, C
f = 450;    // Count rate of GM counter, counts/min
N = f*1e+08;    // Total number of electrons collected per min
Q = N*e;    // Charge collected per min, C
```

```
I = Q/60;    // Averge current in the GM counter, A
printf("\nThe averge current in the GM counter= %3.1
    e A", I);

// Result
// The averge current in the GM counter= 1.2e-10 A
```

Scilab code Exa 5.9 Energy needed to remove a neutron from Ca nucleus

```
// Scilab Code Ex5.9 :: Page-5.39 (2009)
clc;clear;
m_Ca_41 = 40.962278;    // Mass of one Ca-41 nuclei,
    amu
m_Ca_42 = 41.958618;    // Mass of one Ca-41 nuclei,
    amu
m_n = 1.008665;      // Mass of a neutron, amu
delta_m = m_Ca_42 - (m_Ca_41 + m_n);     //
    Difference in the mass of Ca-42 and Ca_41 nuclei,
     amu
E = delta_m*(931.49);    // Binding energy of the
    missing neutron, MeV
printf("\nThe energy needed to remove a neutron from
     Ca-42 nucleus = %5.2f MeV", abs(E));

// Result
// The energy needed to remove a neutron from Ca-42
    nucleus = 11.48 MeV
```

Chapter 6

Semiconductors and Nano Physics

Scilab code Exa 6.1 Resistivity of intrinsic semiconductor at 300 K

```
// Scilab Code Ex6.1:: Page-6.19 (2009)
clc; clear;
T = 300;     // Temperature of pure semiconductor, K
n_i = 2.5e+019; // Intrinsic carrier density, per
   metre square
e = 1.6e-019;    // Charge on an electron, C
mu_e = 0.39;     // Mobility of electrons, Sq.m/V/s
mu_h = 0.19;     // Mobility of holes, Sq.m/V/s
sigma_i = e*n_i*(mu_e+mu_h);     // Conductivity of
   intrinsic semiconductor at 300 K, mho/m
rho_i = 1/sigma_i;  // Resistivity of intrinsic
   semiconductor at 300 K, ohm-m

printf("\nThe resistivity of intrinsic semiconductor
    at 300 K = %4.2f ohm-m", rho_i);

// Result
// The resistivity of intrinsic semiconductor at 300
    K = 0.43 ohm-m
```

Scilab code Exa 6.2 Velocity of electron at Fermi level

```
// Scilab Code Ex6.2: : Page-6.19 (2009)
clc; clear;
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
E_F = 2.0*e;     // Fermi level of Po, J
m = 9.1e-031;    // Mass of an electron , kg
// As E_F = 1/2*m*v^2, solving for v
v = sqrt(2*E_F/m);     // Velocity of electron at Fermi level , m/s
printf("\nThe Velocity of electron at Fermi level = %4.2e m/s", v);

// Result
// The Velocity of electron at Fermi level = 8.39e+05 m/s
```

Chapter 7

Fiber Optics

Scilab code Exa 7.1 Critical angle and acceptance angle in an optical fibre

```
// Scilab Code Ex7.1:: Page-7.7 (2009)
clc; clear;
n1 = 1.6;        // Refractive index of core material
    of fibre
n2 = 1.3;        // Refractive index of cladding
   material of fibre
phi_C = asind(n2/n1);    // Critical angle of optical
    fibre, degrees
theta_Q = asind(sqrt(n1^2-n2^2));    // Acceptance
   angle of optical fibre, degrees

printf("\nThe critical angle of optical fibre = %4.1
   f degrees", phi_C);
printf("\nThe angle of acceptance cone = %5.1f
   degrees", 2*theta_Q);

// Result
// The critical angle of optical fibre = 54.3
   degrees
// The angle of acceptance cone = 137.7 degrees
```

Scilab code Exa 7.2 Critical angle acceptance angle and numerical aperture in an optical fibre

```
// Scilab Code Ex7.2:: Page-7.8 (2009)
clc; clear;
n1 = 1.50;        // Refractive index of core material of fibre
n2 = 1.47;        // Refractive index of cladding material of fibre
phi_C = asind(n2/n1);    // Critical angle of optical fibre, degrees
NA = sqrt(n1^2-n2^2);    // Numerical aperture for the fibre
theta_Q = asind(sqrt(n1^2-n2^2));    // Acceptance angle of optical fibre, degrees

printf("\nThe critical angle of optical fibre = %4.1f degrees", phi_C);
printf("\nThe numerical aperture for the fibre  = %5.3f", NA);
printf("\nThe angle of acceptance cone  = %5.1f degrees", theta_Q);

// Result
// The critical angle of optical fibre = 78.5 degrees
// The numerical aperture for the fibre  = 0.298
// The angle of acceptance cone  =  17.4 degrees
```

Scilab code Exa 7.3 Parameters of an optical fibre using relative refractive index difference

```
// Scilab Code Ex7.3:: Page-7.8 (2009)
clc; clear;
n1 = 1.46;      // Refractive index of the core material
delta = 0.01;    // Relative refractive index difference
NA = n1*sqrt(2*delta);    // Numerical aperture for the fibre
theta_Q = %pi*NA^2;    // Solid acceptance angle of optical fibre for small angles, radians
// As relative refractive index, delta = 1-n2/n1, solving for n2
n2 = n1*(1-delta);    // Refractive index of cladding
phi_C = asind(n2/n1);    // Critical angle of optical fibre, degrees

printf("\nThe numerical aperture for the fibre  = %4.2f", NA);
printf("\nThe solid acceptance angle of the optical fibre  = %4.2f radians", theta_Q);
printf("\nThe critical angle of optical fibre = %4.1f degrees", phi_C);

// Result
// The numerical aperture for the fibre  = 0.21
// The solid acceptance angle of the optical fibre  = 0.13 radians
// The critical angle of optical fibre = 81.9 degrees
```

Scilab code Exa 7.4 Refractive index of cladding

```
// Scilab Code Ex7.4:: Page-7.9 (2009)
clc; clear;
n1 = 1.54;      // Refractive index of the core
```

```
    material
NA = 0.45;    // Numerical aperture for the fibre
n2 = sqrt(n1^2-NA^2);    // Refractive index of cladding

printf("\nThe refractive index of cladding = %4.2f", n2);

// Result
// The refractive index of cladding = 1.47
```

Scilab code Exa 7.5 Numerical aperture for an optical fibre

```
// Scilab Code Ex7.5:: Page-7.9 (2009)
clc; clear;
n1 = 1.544;      // Refractive index of the core material
n2 = 1.412;    // Refractive index of cladding
NA = sqrt(n1^2-n2^2);    // Numerical aperture for the fibre

printf("\nThe numerical aperture for an optical fibre = %4.2f", NA);

// Result
// The numerical aperture for an optical fibre = 0.62
```

Scilab code Exa 7.6 Refractive index of the cladding

```
// Scilab Code Ex7.6:: Page-7.9 (2009)
clc; clear;
```

```
n1 = 1.544;      // Refractive index of the core material
theta0 = 35;     // Acceptance angel for an optical fibre , degrees
// As theta0 = asind(sqrt(n1^2-n2^2)), solving for n2
n2 = sqrt(n1^2-sind(theta0)^2);      // Refractive index of cladding

printf("\nThe refractive index of the cladding = %4.2f", n2);

// Result
// The refractive index of the cladding = 1.43
```

Scilab code Exa 7.7 Comparison of the acceptance angle for meridional rays with that for the skew rays

```
// Scilab Code Ex7.7:: Page-7.10 (2009)
clc; clear;
NA = 0.4;       // Numerical aperture of the optical fibre
n0 = 1;          // Refractive index of fibre in air
theta_a = asind(NA/n0);   // Acceptance angle for meridional rays , degrees
theta = 100;     // Direction through which the skew rays are bent at each reflection , degrees
r = theta/2;     // Angle of reflection , degrees
theta_as = asind(NA/(cosd(r)*n0));   // Acceptance angle for skew rays , degrees

printf("\nAcceptance angle for meridional rays = %4.1f degrees", theta_a);
printf("\nAcceptance angle for skew rays = %4.1f degrees", theta_as);
```

```

// Result
// Acceptance angle for meridional rays = 23.6 degrees
// Acceptance angle for skew rays = 38.5 degrees
```

Scilab code Exa 7.8 Normalized frequency for V number for the fibre

```
// Scilab Code Ex7.8: : Page-7.13 (2009)
clc; clear;
NA = 0.16;       // Numerical aperture of the step index fibre
n1 = 1.50;       // Refractive index of the core material
d = 65e-006;     // Diameter of the core, m
lambda = 0.9e-006;   // Wavelength of transmitted light, m
V = %pi*d/lambda*NA;   // V-number for the optical fibre

printf("\nThe V-number for the optical fibre = %5.2f", V);

// Result
// The V-number for the optical fibre = 36.30
```

Scilab code Exa 7.9 Number of modes in the step index fibre

```
// Scilab Code Ex7.9:: Page-7.13 (2009)
clc; clear;
NA = 0.28;       // Numerical aperture of the step index fibre
d = 55e-006;     // Diameter of the core, m
```

```
lambda = 0.9e-006;   // Wavelength of transmitted
    light , m
M_N = (2.22*d*(NA)/lambda)^2;    // Number of modes
    in the step index fibre

printf("\nThe number of modes in the step index
    fibre = %4d degrees", M_N);

// Result
// The number of modes in the step index fibre =
    1442 degrees
```

Scilab code Exa 7.10 Radius of core for single mode operation in step index fibre

```
// Scilab Code Ex7.10:: Page-7.14 (2009)
clc; clear;
n1 = 1.480;      // Refractive index of core material
n2 = 1.47;      // Refractive index of cladding
    material
lambda = 850e-006;   // Wavelength of light used, m
NA = sqrt(n1^2-n2^2);        // Numerical aperture of
    the step index fibre
theta0 = asind(NA);      // Maximum acceptance angle
    for the fibre, degrees
M_N = 1;     // Number of modes in step index cable
// As number of modes, M_N = 1/2*V^2, solving for V
V = sqrt(2*M_N);     // V-number for the fibre
// As V = 2*%pi*a/lambda*NA, solving for a
a = V*lambda/(2*%pi*NA);     // Radius of core for
    single mode operation in step index fibre, m

printf("\nThe radius of core for single mode
    operation in step index fibre = %3.1e", a);

```

```
// Result
// The radius of core for single mode operation in step index fibre = 1.1e-03
// The ansswer is quoted wrong in the textbook
```

Scilab code Exa 7.11 Signal attenuation in optical fibre

```
// Scilab Code Ex7.11: : Page-7.16 (2009)
clc; clear;
Pi = 1.5;  // Input power to the optical fibre, mW
Po = 0.5;  // Output power to the optical fibre, mW
L = 0.12;   // Length of the optical fibre, km
alpha_dB = 10/L*log10(Pi/Po); // Signal attenuation in optical fibre, dB/km

printf("\nThe signal attenuation in optical fibre = %4.1f dB/km", alpha_dB);

// Result
// The signal attenuation in optical fibre = 39.8 dB/km
```

Chapter 8

Laser

Scilab code Exa 8.1 Difference between upper and lower energy levels for the most prominent wavelength

```
// Scilab Code Ex8.1:: Page-8.8 (2009)
clc; clear;
lambda = 31235;    // Wavelength of prominent
    emission of laser, aangstrom
E = 12400/lambda;    // Energy difference between the
     two levels, eV

printf("\nThe difference between upper and lower
    energy levels for the most prominent wavelength
    = %5.3f eV", E);

// Result
// The difference between upper and lower energy
    levels for the most prominent wavelength  = 0.397
     eV
```

Scilab code Exa 8.2 Frequency and wavelength of carbon dioxide laser

```
// Scilab Code Ex8.2:: Page-8.8 (2009)
clc; clear;
E = 0.121;    // Energy difference between the two levels, eV
lambda = 12400/E;    // Wavelength of the radiation, angstrom
f = 3e+08/(lambda*1e-010);   // Frequency of the radiation, Hz

printf("\nThe wavelength of the radiation = %8.1f angstrom", lambda);
printf("\nThe frequency of the radiation = %4.2e Hz", f);

// Result
// The wavelength of the radiation = 102479.3 angstrom
// The frequency of the radiation = 2.93e+13 Hz
```

Scilab code Exa 8.3 Energy of one emitted photon and total energy available per laser pulse

```
// Scilab Code Ex8.3:: Page-8.8 (2009)
clc; clear;
lambda = 7000;   // Wavelength of the Ruby laser, angstrom
e = 1.6e-019;    // Energy equivalent of 1 eV, J/eV
N = 2.8e+019;    // Total number of photons
E = 12400/lambda;    // Energy of one emitted photon, eV
E_p = E*e*N;     // Total energy available per laser pulse, joule

printf("\nThe energy of one emitted photon = %4.2e J", E*e);
```

```
printf("\nThe total energy available per laser pulse = %4.2f joule", E_p);

// Result
// The energy of one emitted photon = 2.83e-19 J
// The total energy available per laser pulse = 7.94 joule
```

Scilab code Exa 8.4 Relative population of levels in Ruby laser

```
// Scilab Code Ex8.4:: Page-8.9 (2009)
clc; clear;
lambda = 7000;   // Wavelength of the emitted light, angstrom
k = 8.6e-005;    // Boltzmann constant, eV/K
dE = 12400/lambda;   // Energy difference of the levels, eV
T = [300 500];    // Temperatures of first and second states, K
for i = 1:1:2
    N2_ratio_N1 = exp(-(dE/(k*T(i))));   // Relative population
    printf("\nThe relative population at %d K = %3.1e", T(i), N2_ratio_N1);
end

// Result
// The relative population at 300 K = 1.5e-30
// The relative population at 500 K = 1.3e-18
// The answer is given wrong in the textbook for first part.
```

Scilab code Exa 8.5 Population of two states in He Ne laser

```
// Scilab Code Ex8.5:: Page-8.9 (2009)
clc; clear;
lambda = 7000;  // Wavelength of the emitted light, angstrom
k = 8.6e-005;   // Boltzmann constant, eV/K
dE = 12400/lambda;  // Energy difference of the levels, eV
T = 27+273;   // Temperatures of the state, K
N2_ratio_N1 = exp(-(dE/(k*T)));  // Relative population
printf("\nThe relative population of two states in He-Ne laser at %d K = %3.1e", T, N2_ratio_N1);


// Result
// The relative population of two states in He-Ne laser at 300 K = 1.5e-30
// The answer is given wrong in the textbook
```