Scilab Textbook Companion for Modern Physics by G. Aruldhas and P. Rajagopal[1]

Created by
Pankaj Biswas
Fundamentals of Physics
Physics
Shri Mata Vaishno Devi University
College Teacher
None
Cross-Checked by
None

July 31, 2019

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

Book Description

Title: Modern Physics

Author: G. Aruldhas and P. Rajagopal

Publisher: Phi Learning, New Delhi

Edition: 1

Year: 2014

ISBN: 978-81-203-2597-5

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

The Special Theory of Relativity

Scilab code Exa 1.2 Relativistic velocity addition

```
1 // Ex1_2: Page:10 (2014)
2 clc; clear;
3 c = 1;  // For simplicity assume speed of light to
     be unity, m/s
4 v = 0.6*c;  // Speed of first rocket w.r.t. the
     earth, m/s
5 u_prime = 0.9*c;     // Speed of second rocket w.r.t.
      the first, m/s
6 // Case 1: Firing in the same direction as that of
     the first
7 u = (u_prime + v)/(1 + u_prime*v/c^2);  // Speed of
     2nd rocket w.r.t. earth from Velocity Addition
     Rule
8 printf("\nThe speed of second rocket w.r.t. earth
     fired in the same direction = %5.3fc", u);
9 // Case 2: Firing in the opposite direction as that
     of the first
10 u = (-u_prime + v)/(1 - u_prime*v/c^2);  // Speed of
      2nd rocket w.r.t. earth from Velocity Addition
```

```
    Rule
printf("\nThe speed of second rocket w.r.t. earth
    fired in the opposite direction = %5.3fc", u);

// Result
// The speed of second rocket w.r.t. earth fired in
    the same direction = 0.974c
// The speed of second rocket w.r.t. earth fired in
    the opposite direction = -0.652c
```

Scilab code Exa 1.3 Length contraction

```
// Ex1_3 Page:12 (2014)
clc; clear;
c = 1;  // For simplicity assume speed of light to
    be unity, m/s
L0 = 1; // For simplicity assume the classical
    length of the rod to be unity, m
d = 1;   // Percentage difference of relativistic
    length from the classical length
L = (1 - d/100)*L0;   // Relativistic length, m
v = poly(0, "v");   // Declare speed variable
v = roots(1 - v^2 - (L/L0)^2);    // Relativistic
    speed of the moving object
printf("\nThe speed at which the relativistic value
    for length differ from the classical value by one
     percent = %5.3fc", v(1));

// Result
// The speed at which the relativistic value for
    length differ from the classical value by one
    percent = 0.141c
```

Scilab code Exa 1.4 Speed of an unstable particle

```
1 // Ex1_4 Page:12 (2014)
2 clc; clear;
3 c = 3e+08;  // Speed of light in vacuum, m/s
4 delta_tau = 2.6e-08;    // Mean lifetime of an
     unstable particle at rest, s
5 d = 20; // Distance travelled by the unstable
     particle before it decays, m
6 v = poly(0, "v");   // Declare the speed variable
7 v = 1/sqrt(roots(d^2*v - (d/c)^2 - delta_tau^2));
     // Speed of the particle in Lab frame from Time
     Dilation relation, m/s
8 printf("\nThe speed at which the unstable particle
     20 m distance before decaying = %3.1e m/s", v);
9
10 // Result
11 // The speed at which the unstable particle 20 m
     distance before decaying = 2.8e+08 m/s
```

Scilab code Exa 1.5 Proper lifetime of the particles

```
1 // Ex1_5 Page:13 (2014)
2 clc; clear;
3 c = 3e+08;  // Speed of light in vacuum, m/s
4 v = 0.9*c;  // Speed of beam of particles, m/s
5 delta_t = 5e-06;    // Mean lifetime of particles as
     measured in the Lab frame, s
6 delta_tau = delta_t*sqrt(1-(v/c)^2);    // The
     proper lifetime of the particles from Time
     Dilation Relation, s
7 printf("\nThe proper lifetime of the particles = %4
     .2e s", delta_tau);
8
9 // Result
```

```
10 // The proper lifetime of the particles = 2.18e-06 s
```

Scilab code Exa 1.6 Relativistic variation of mass with velocity

```
1 // Ex1_6 Page:15 (2014)
2 clc; clear;
3 c = 1;  // For simplicity assume speed of light to
     be unity, m/s
4 m0 = 1; // For simplicity assume the rest mass, kg
5 d = 1;   // Percentage difference of relativistic
     mass from the rest mass, kg
6 m = m0*(1+20/100);   // Effective mass of the body
     while moving, kg
7 v = poly(0, "v");   // Declare speed variable
8 v = roots(1 - (v/c)^2 - (m0/m)^2);    //
     Relativistic speed of the moving body
9 printf("\nThe relativistic speed of the moving body
     = %5.3fc", v(1));
10
11 // Result
12 // The relativistic speed of the moving body = 0.553
     c
```

Scilab code Exa 1.7 Rate of decrease of mass

```
1 // Ex1_7 Page:17 (2014)
2 clc; clear;
3 c = 3e+08;  // Speed of light in vacuum, m/s
4 delta_E = 4e+026;   // Rate of radiation of energy
     from the sun, J/s
5 delta_m = delta_E/c^2;  // The decrease in mass, kg
6 printf("\nThe decrease in mass = %4.2e kg", delta_m)
     ;
```

```
7
8 // Result
9 // The decrease in mass = 4.44e+09 kg
```

Scilab code Exa 1.8 Momentum and velocity of an electron

```
1 // Ex1_8 Page:18 (2014)
2 clc; clear;
3 c = 1;  // For simplicity assume speed of light to
     be unity, m/s
4 m_e = 0.512;  // The rest energy of an electron, MeV
5 T = 10; // The kinetic energy of the electron, MeV
6 E = T + m_e;    // Total energy of the electron, MeV
7 p = sqrt(E^2 - m_e^2)/c;    // Momentum of the
     electron from Relativistic Energy Relation, MeV
8 v = poly(0, "v");   // Declare speed variable
9 v = roots(1 - (v/c)^2 - (m_e/E)^2);    // Speed of
     the electron from Relativistic Energy Variation
     formula, m/s
10 printf("\nThe momentum of the electron = %4.1f MeV/c
     ", p);
11 printf("\nThe speed of the electron = %6.4fc", v(1))
     ;
12
13 // Result
14 // The momentum of the electron = 10.5 MeV/c
15 // The speed of the electron = 0.9988c
```

Chapter 2

Particle Nature of Radiation

Scilab code Exa 2.1 Number of photons emitted per second

```
1 // Ex2_1 Page:28 (2014)
2 clc; clear;
3 h = 6.626e-034; // Planck's constant, Js
4 nu = 100e+06;   // Operational frequency of the
     source, Hz
5 P = 100e+03;    // Power radiated by the source, W
6 E = h*nu;   // Energy radiated by the source, J
7 n = P/E;    // Number of quanta of energy per second
8 printf("\nThe number of photons emitted per second =
     %5.3e photons/s", n);
9
10 // Result
11 // The number of photons emitted per second = 1.509e
     +30 photons/s
```

Scilab code Exa 2.2 Maximum kinetic energy and speed of photoelectrons

```
1 // Ex2_2 Page:31 (2014)
```

```
2 clc; clear;
3 h = 6.626e-034; // Planck's constant, Js
4 c = 3e+08;  // Speed of light, m/s
5 e = 1.6e-019;   // Energy equivalent of 1 eV, J
6 m = 9.1e-031;   // Mass of an electron, kg
7 w0 = 2.28;  // Work function of sodium, eV
8 lambda = 400e-09;   // Wavelength of light, m
9 nu = c/lambda;    // Frequency of light, Hz
10 KE_max = h*nu/e - w0;  // Maximum kinetic energy of
   photoelectrons, eV
11 v = sqrt(2*KE_max*e/m);   // Speed of photoelectrons
   , m/s
12 printf("\nThe maximum kinetic energy of
   photoelectrons = %5.3f eV", KE_max);
13 printf("\nThe speed of photoelectrons = %4.2e m/s",
   v);
14
15 // Result
16 // The maximum kinetic energy of photoelectrons =
   0.826 eV
17 // The speed of photoelectrons = 5.39e+05 m/s
```

Scilab code Exa 2.3 Cutoff wavelength and stopping potential of Al

```
1 // Ex2_3 Page:31 (2014)
2 clc; clear;
3 h = 6.626e-034; // Planck's constant, Js
4 c = 3e+08;  // Speed of light, m/s
5 e = 1.6e-019;   // Energy equivalent of 1 eV, J
6 w0 = 4.2;  // Work function of aluminium, eV
7 lambda = 2000e-10;   // Wavelength of incident light
   , m
8 lambda0 = h*c/(w0*e);  // The cut-off wavelength for
    aluminium, m
9 E = h*c/(lambda*e); // Energy of 2000 angstrom
```

```
    photon , eV
10 KE = E - w0;     // Kinetic energy of fastest
    electron , eV
11 printf("\nThe cut-off wavelength for aluminium = %4d
     angstrom", lambda0/1e-010);
12 printf("\nThe stopping potential = %4.2f V", KE);
13
14 // Result
15 // The cut-off wavelength for aluminium = 2958
    angstrom
16 // The stopping potential = 2.01 V
```

Scilab code Exa 2.4 Momentum and effective mass of a photon

```
1 // Ex2_4 Page:33 (2014)
2 clc; clear;
3 h = 6.626e-034; // Planck's constant , Js
4 c = 3e+08;  // Speed of light , m/s
5 lambda = 0.2e-09;   // Wavelength of photon , m
6 p = h/lambda;   // Momentum of the photon , kgm/s
7 m = p/c;     // Effective mass of photon , kg
8 printf("\nThe momentum of the photon = %3.1e kg-m/s"
    , p);
9 printf("\nThe effective mass of photon = %3.1e kg",
    m);
10
11 // Result
12 // The momentum of the photon = 3.3e-24 kg-m/s
13 // The effective mass of photon = 1.1e-32 kg
```

Scilab code Exa 2.5 Wavelength of scattered X rays at different angles

```
1 // Ex2_5 Page:35 (2014)
```

```
clc; clear;
h = 6.626e-034; // Planck's constant, Js
c = 3e+08;  // Speed of light, m/s
m0 = 9.1e-031;  // Mass of the electron, kg
lambda = 0.15e-09;  // Wavelength of the incident X-rays, m
theta = zeros(3);   // Declare a row matrix for theta
theta = [0, 90, 180];   // Scattering angles of photons, degree
for i = 1:3
    lambda_prime = lambda + h/(m0*c)*(1 - cosd(theta(i)));  // New wavelength due to Compton Shift, m
    printf("\nThe wavelength of X-rays scattered at %d degrees = %5.3f nm", theta(i), lambda_prime/1e-09);
end

// Result
// The wavelength of X-rays scattered at 0 degrees = 0.150 nm
// The wavelength of X-rays scattered at 90 degrees = 0.152 nm
// The wavelength of X-rays scattered at 180 degrees = 0.155 nm
```

Scilab code Exa 2.6 Wavelength of a photon for pair production

```
// Ex2_6 Page:36 (2014)
clc; clear;
h = 6.626e-034; // Planck's constant, Js
c = 3e+08;  // Speed of light in vacuum, m/s
E0 = 0.511; // Rest energy of an electron, MeV
e = 1.6e-019;   // Energy equivalent of 1 eV, J
```

```
E = 2*E0*e*1e+06; // Rest energy of electron-
    positron pair, J
lambda = h*c/E; // Wavelength of photon to bring
    pair production, m
printf("\nWavelength of a photon that can produce an
     electron-positron pair = %4.2e m", lambda);

// Result
// Wavelength of a photon that can produce an
    electron-positron pair = 1.22e-12 m
```

Chapter 3

Atomic Models

Scilab code Exa 3.1 Alpha particle scattering from gold foil

```
1 // Ex3_1 Page:45 (2014)
2 clc; clear;
3 t = 2e-06;  // Thickness of gold foil, m
4 T = 8;  // Kinetic energy of alpha-particles, MeV
5 rho = 19.3; // Density of gold foil, g/cm-cube
6 k = 8.984e+09;  // Coulomb's constant, N-Sq.m/Sq.C
7 Z = 79; // Atomic number of gold
8 Z_prime = 2;    // Atomic number of He nucleus
9 e = 1.6e-019;   // Charge on an electron, C
10 A = 197;    // Atomic weight of gold
11 N = 6.02e+023;   // Avogadro's number, atoms/mol
12 n = N*rho*1e+06/A;    // Number density of atoms,
    atoms/metre-cube
13 b = k*Z*Z_prime*e^2/(2*T*1e+06*e);  // Impact
    parameter for alpha particle, m
14 f = %pi*b^2*n*t;    // Fraction of alpha-particles
    scattered at angles greater than 90 degrees
15 printf("\nFraction of alpha-particles scattered at
    angles greater than 90 degrees = %3.1e", f);
16
17 // Result
```

```
18 // Fraction of alpha-particles scattered at angles
   greater than 90 degrees = 7.5e-05
```

Scilab code Exa 3.3 Wavelength of photon for ionizing an H atom

```
1 // Ex3_3 Page:48 (2014)
2 clc; clear;
3 h = 6.626e-034; // Planck's constant, Js
4 c = 3e+08;  // Speed of light in vacuum, m/s
5 e = 1.6e-019;   // Charge on an electron, C
6 T = 10.5;   // Kinetic energy of ejected electron,
   eV
7 E = 13.6 + T;   // Energy of the photon, eV
8 lambda = h*c/(E*e); // Wavelength of incident photon
    from Planck's Quantum relation, m
9 printf("\nWavelength of the photon that would ionize
    a hydrogen atom = %5.2f nm", lambda/1e-09);
10
11 // Result
12 // Wavelength of the photon that would ionize a
   hydrogen atom = 51.55 nm
```

Scilab code Exa 3.6 Wavelength of H alpha line for deuterium

```
1 // Ex3_6 Page:51 (2014)
2 clc; clear;
3 M = 1;    // For simplicity assume mass of a proton
   to be unity, amu
4 m_e = 1/1836;    // Mass of an electron, amu
5 m_d = 2*M;    // Mass of a deuterium, amu
6 lambda = 6562.8; // Wavelength of H_alpha line of
   hydrogen, angstrom
```

```
7  mu = M/(1 + M/m_e); // Reduced mass of an electron-
       proton system, amu
8  mu_prime = m_d/(1 + m_d/m_e);    // Reduced mass of
       an electron-deuterium system, amu
9  lambda_prime = lambda*mu/mu_prime;  // Wavelength of
        H_alpha line of deuterium, angstrom
10 printf("\nThe wavelength of H_alpha line of
       deuterium = %4d angstrom", lambda_prime);
11
12 // Result
13 // The wavelength of H_alpha line of deuterium =
       6561 angstrom
```

Chapter 4

Wave Mechanical Concepts

Scilab code Exa 4.2 de Broglie wavelength of an electron

```
1 // Ex4_2 Page:59 (2014)
2 clc;clear;
3 e = 1.6e-019;     // Energy conversion factor, J/eV
4 m = 9.1e-031;     // Mass of an electron, kg
5 h = 6.626e-034;   // Planck's constant, Js
6 c = 3e+08;     // Speed of light in vacuum, m/s
7 // Non-relativistic case:
8 E = 1;     // Kinetic energy of an electron, eV
9 p = sqrt(2*m*E*e);     // Momentum of the electron,
    kg-m/s
10 lambda = h/p*1e+09;     // de-Broglie wavelength of
    electron, nm
11 printf("\nNon-relativistic Case:\nThe de-Broglie
    wavelength of electron = %3.1f nm", lambda);
12 // Relativistic case:
13 KE = 100;     // Kinetic energy of an electron, MeV
14 p = KE*e/c*1e+06;     // Relativistic momentum of the
     electron, kg-m/s
15 lambda = h/p;     // de-Broglie wavelength of
    electron, m
16 printf("\nRelativistic case:\nThe de-Broglie
```

```
    wavelength of electron = %5.3e m", lambda);
17
18
19 // Result
20 // Non-relativistic Case:
21 // The de-Broglie wavelength of electron = 1.2 nm
22 // Relativistic case:
23 // The de-Broglie wavelength of electron = 1.242e-14
    m
```

Scilab code Exa 4.3 Uncertainty in position of an electron

```
1 // Ex4_3 Page:64 (2014)
2 clc;clear;
3 m = 9.1e-031;     // Mass of an electron, kg
4 h_cross = 1.05e-034;  // Reduced Planck's constant,
    Js
5 v = 4e+06;      // Speed of the electron, m/s
6 p = m*v;      // Momentum of the electron, kg-m/s
7 delta_p = p/100;      // Uncertainty in momentum of
    the electron, kg-m/s
8 delta_x = h_cross*1e+09/(2*delta_p);      //
    Uncertainty in position of the electron, nm
9 printf("\nThe uncertainty in position of the
    electron = %4.2f nm", delta_x);
10
11 // Result
12 // The uncertainty in position of the electron =
    1.44 nm
```

Scilab code Exa 4.4 Width of the spectral line

```
1 // Ex4_4 Page:64 (2014)
```

```
clc;clear;
c = 3e+08;     // Speed of the electron, m/s
lambda = 4000e-010;     // Wavelength of the spectral line, m
delta_t = 1e-08;     // Average lifetime of an excited atomic state, s
delta_lambda = lambda^2/(4*%pi*c*delta_t);     // Natural width of the spectral line, m
printf("\nThe natural width of the spectral line = %4.2e m", delta_lambda);

// Result
// The natural width of the spectral line = 4.24e-015 m
```

Chapter 6

Quantum Mechanics of Simple Systems

Scilab code Exa 6.1 Probability of finding an electron in a square well

```
1 // Ex6_1 Page:90 (2014)
2 clc;clear;
3 a = 2e-010;     // Length of the square well, m
4 x1 = 0;     // Lower limit of position, m
5 x2 = 0.25e-010;     // Upper limit of position, m
6 P = integrate('(sqrt(2/a)*sin(%pi*x/a))^2', 'x', x1,
    x2); // Probabilitiy of finding the electron in
   the given region
7 printf("\nThe probabilitiy of finding the electron
   in the region x = 0 to 0.25e-010 = %6.4f ", P);
8
9 // Result
10 // The probabilitiy of finding the electron in the
   region x = 0 to 0.25e-010 = 0.0125
```

Scilab code Exa 6.2 Zero point energy and force constant of a CO molecule

```
// Ex6_2 Page:96 (2014)
clc;clear;
v0 = 6.43e+013;     // The vibrational frequecy of CO
    molecule, Hz
e = 1.6e-019;     // Energy conversion factor, J/eV
mu = 1.1385e-026;     // The reduced mass of CO
   molecule, kg
h = 6.626e-034;     // Planck's constant, Js
E0 = 1/(2*e)*h*v0;     // Zero point energy, eV
k = 4*(22/7*v0)^2*mu;     // Force constant of the CO
    bond, N/m
printf("\nThe zero point energy of the CO bond = %5
   .3f eV", E0);
printf("\nThe force constant of the CO bond = %4d N/
   m", k);

// Result
// The zero point energy of the CO bond = 0.133 eV
// The force constant of the CO bond = 1859 N/m
```

Scilab code Exa 6.6 Energy difference between the first two rotational levels of C

```
// Ex6_6 Page:104 (2014)
clc;clear;
e = 1.6e-019;     // Energy conversion factor, J/eV
h_cross = 1.054e-034;     // reduced Planck's
   constant, Js
r = 1.131e-010;     // Intermolecular separation of
   CO molecule, m
m_Carbon = 19.9217e-027;     // Mass of carbon atom,
   kg
m_Oxygen = 26.5614e-027;     // Mass of oxygen atom,
   kg
mu = m_Carbon*m_Oxygen/(m_Carbon+m_Oxygen);     //
   Reduced mass of CO molecule, kg
```

```
I = mu*r^2;     // Moment of inertia of CO molecule, kg-Sq.m
delta_E = h_cross^2/(I*e);     // Energy difference between the first two rotational energy levels of CO molecule, eV
printf("\nThe energy difference between the first two rotational energy levels of CO molecule = %4.2e eV", delta_E);

// Result
// The energy difference between the first two rotational energy levels of CO molecule = 4.77e-004 eV
```

Scilab code Exa 6.7 Orbital angular momentum

```
// Ex6_7 Page:105 (2014)
clc;clear;
l = 1;     // Orbital angular momentum quantum number
m = [1, 0, -1];     // The possible z-components of l
printf("\nThe possible orientations of vector L with respect to the z-axis are:");
for i = 1:3
    theta = acosd(m(i)/sqrt(l*(l+1)));
    printf("\ntheta = %d degree (m = %d)", theta, m(i));
end

// Result
// The possible orientations of vector L with respect to the z-axis are:
// theta = 45 degree (m = 1)
// theta = 90 degree (m = 0)
// theta = 135 degree (m = -1)
```

Chapter 7

Atomic Physics

Scilab code Exa 7.1 Energy difference between 2p states of hydrogen atom

```
1 // Ex7_1 Page:113 (2014)
2 clc;clear;
3 e = 1.6e-019;     // Energy conversion factor, J/eV
4 mu_B = 9.27e-024;     // Bohr magneton, J/T
5 B = 3;      // Magnetic field, T
6 m_l = [-1, 1];     // Orbital magnetic quantum number
7 dE = mu_B*B*(m_l(2)-m_l(1))/e;     // Energy
    difference between m_l = -1 and m_l = +1, eV
8 printf("\nThe energy difference between m_l = -1 and
     m_l = +1 components in 2p state = %4.2e eV", dE)
    ;
9
10 // Result
11 // The energy difference between m_l = -1 and m_l =
    +1 components in 2p state = 3.48e-004 eV
```

Scilab code Exa 7.3 d electron of a H atom subjected to the vector atom model

```
// Ex7_3 Page:118 (2014)
clc;clear;
l = 2;     // Orbital angular momentum quantum number
s = 0.5;     // Spin quantum number
state = ["D(5/2)", "D(3/2)"];     // States of the d-
    electron
j = [l + s, l - s];     // Total angular momentum
    quantum number
for i = 1:2
    theta(i) = acosd((j(i)*(j(i)+1) - l*(l+1) - s*(s
        +1))/(2*sqrt(l*(l+1))*sqrt(s*(s+1))));
    printf("\nThe angle between L and S for %s state
        = %5.2f degree", state(i), theta(i));
end

// Result
// The angle between L and S for D(5/2) state =
    61.87 degree
// The angle between L and S for D(3/2) state =
    135.00 degree
```

Scilab code Exa 7.6 LS coupling of electrons in 4p and 4d subshells

```
// Ex7_6 Page:126 (2014)
clc;clear;
l1 = 1;     // Orbital angular momentum quantum
    number of first electron
l2 = 2;     // Orbital angular momentum quantum
    number of second electron
s1 = 0.5;     // Spin quantum number of first
    electron
s2 = 0.5;     // Spin quantum number of second
    electron
L_max = l1 + l2;
L_min = l2 - l1;
```

```
S_max = s1 + s2;
S_min = s1 - s2;
printf("\nThe possible values of L, S and J are:");
for L = L_max:-1:L_min
    for S = S_max:-1:S_min
        J_max = L + S; J_min = L - S;
        printf("\nL = %d, S = %d, J = ", L, S);
        for J = J_max:-1:J_min
            if (J <> J_min) then
                printf("%d, ", J);
            else
            printf("%d", J);
            end
        end
    end
end

// Result
// The possible values of L, S and J are:
// L = 3, S = 1, J = 4, 3, 2
// L = 3, S = 0, J = 3
// L = 2, S = 1, J = 3, 2, 1
// L = 2, S = 0, J = 2
// L = 1, S = 1, J = 2, 1, 0
// L = 1, S = 0, J = 1
```

Scilab code Exa 7.10 Lande g factor for various states

```
// Ex7_10 Page:136 (2014)
clc;clear;
// Case 1: For pure orbital angular momentum
S = poly(0, 'S');     // Total spin angular momentum
    variable
S = 0;     // S value for pure orbital angular
    momentum
```

```
L = poly(0, 'L');     // Total orbital angular
    momentum variable
J = L + S;     // J value for pure orbital angular
    momentum
g = horner(1 + (J*(J + 1) + S*(S + 1) - L*(L + 1))
    /(2*J*(J + 1)), 0); // Lande's g-factor
printf("\nFor pure orbital angular momentum, g = %d"
    , g);
// Case 2: For pure spin angular momentum
S = poly(0, 'S');     // Total spin angular momentum
    variable
L = 0;     // L value for pure spin angular momentum
J = L + S;     // J value for pure spin angular
    momentum
g = horner(1 + (J*(J + 1) + S*(S + 1) - L*(L + 1))
    /(2*J*(J + 1)), 0); // Lande's g-factor
printf("\nFor pure spin angular momentum, g = %d", g
    );
// Case 3: For state 3P1
S = 1;     // S value for pure spin angular momentum
L = 1;     // L value for pure spin angular momentum
J = L + S;     // J value for pure spin angular
    momentum
g = horner(1 + (J*(J + 1) + S*(S + 1) - L*(L + 1))
    /(2*J*(J + 1)), 0); // Lande's g-factor
printf("\nFor 3P1 state, g = %d/2", 2*g);

// Result
// For pure orbital angular momentum, g = 1
// For pure spin angular momentum, g = 2
// For 3P1 state, g = 3/2
```

Scilab code Exa 7.12 `L alpha X rays and L electrons from silver`

```
// Ex7_12 Page:141 (2014)
```

```
clc;clear;
E_K_alpha = 21.99;     // The energy in silver for
   K_alpha X-ray, keV
E_K_beta = 25.145;     // The energy in silver for
   K_beta X-ray, keV
EB_K = 25.514;     // The binding energy of K
   electron in silver, keV
E_L_alpha = E_K_beta - E_K_alpha;     // The energy
   in silver for L_alpha X-ray, keV
EB_L = -EB_K + E_K_alpha;     // The binding energy
   of L electron in silver, keV
printf("\nThe energy of the L_alpha X-ray = %5.3f
   keV", E_L_alpha);
printf("\nThe binding energy of the L electron = %5
   .3f keV", EB_L);

// Result
// The energy of the L_alpha X-ray = 3.155 keV
// The binding energy of the L electron = -3.524 keV
```

Scilab code Exa 7.13 Energy of K alpha X rays of sodium

```
// Ex7_13 Page:141 (2014)
clc;clear;
Z = 11;     // Atomic number of sodium
h = 6.626e-034;     // Planck's constant, Js
e = 1.6e-019;     // Energy conversion factor, J/eV
c = 3e+08;     // Speed of light in vacuum, m/s
R_inf = 1.097e+07;     // Rydberg constant, per metre
E_K_alpha = (3*h*c*R_inf*(Z - 1)^2)/(4*e*1e+03);
   // The energy of the K_alpha X-ray of sodium, keV
printf("\nThe energy of the K_alpha X-ray of sodium
   = %4.2f keV", E_K_alpha);

// Result
```

```
12 // The energy of the K_alpha X-ray of sodium = 1.02
     keV
```

Chapter 8

Statistical Physics

Scilab code Exa 8.2 Silver as a Fermi Dirac system

```
1 // Ex8_2 Page:164 (2014)
2 clc;clear;
3 h = 6.626e-034;     // Planck's constant, Js
4 m = 9.1e-031;     // Mass of an electron, kg
5 e = 1.6e-019;     // Energy conversion factor, J/eV
6 rho = 10.5;     // Density of silver, g/cc
7 M = 108;     // Atomic weight of silver, g/mol
8 N_A = 6.02e+023     // Avogadro's number, atoms/mol
9 n = rho*N_A/(M*1e-06);     // Number density of
    conduction electrons, per/metre-cube
10 E_F = h^2/(8*m*e)*(3/%pi*n)^(2/3);     // Fermi
    energy, eV
11 E_bar = 3/5*E_F;     // Mean energy of electron at 0
    K, eV
12 printf("\nThe number density of conduction electrons
    = %4.2e per metre-cube", n);
13 printf("\nThe Fermi energy for silver = %4.2f eV",
    E_F);
14 printf("\nThe mean energy of the electron at 0 K =
    %4.2f eV", E_bar);
15
```

```
16 // Result
17 // The number density of conduction electrons = 5.85
      e+028 per metre-cube
18 // The Fermi energy for silver = 5.51 eV
19 // The mean energy of the electron at 0 K = 3.31 eV
```

Scilab code Exa 8.3 Electronic contribution to the molar heat capacity of silver

```
1 // Ex8_3 Page:164 (2014)
2 clc;clear;
3 T = 300;     // Room temperature, K
4 E_F = 5.49;     // Fermi energy, eV
5 k = 1.38e-023;     // Boltzmann constant, J/K
6 e = 1.602e-019;     // Energy conversion factor, J/eV
7 C_v = %pi^2*k*T/(2*E_F*e);     // Molar heat capacity
     , J/mol/K
8 printf("\nThe molar heat capacity = %6.4f R", C_v);
9
10 // Result
11 // The molar heat capacity = 0.0232 R
```

Chapter 9

Molecular Spectra

Scilab code Exa 9.1 Bond length of CO molecule

```
1 // Ex9_1 Page:172 (2014)
2 clc;clear;
3 d = 3.8626;     // The average spacing between
     adjacent rotational lines of CO molecule, per cm
4 B = d/2*100;     // Rotational constant, per m
5 h = 6.626e-034;     // Planck's constant, Js
6 c = 3e+08;     // Speed of light in vacuum, m/s
7 I = h/(8*%pi^2*B*c);     // Moment of inertia of the
     CO molecule, kg/Sq.m
8 N_A = 6.022e+023;     // Avogadro's number, atoms/mol
9 M_C = 0.012;     // Isotopic masses of C-12 atom, kg/
     mol
10 M_O = 0.016;     // Isotopic masses of Om-16 atom, kg
     /mol
11 mu = M_C*M_O/((M_C + M_O)*N_A);     // Reduced mass
     of CO molecule, kg
12 r = sqrt(I/mu);     // Bond length of CO, m
13 printf("\nThe bond length of CO = %5.3e m", r);
14
15 // Result
16 // The bond length of CO = 1.128e-010 m
```

Scilab code Exa 9.3 Vibrational energy states of NO molecule

```
1 // Ex9_3 Page:175 (2014)
2 clc;clear;
3 f0 = 1876.06;     // Fundamental frequancy of NO
     molecule, per cm
4 f1 = 3724.2;     // First overtone frequency of NO
     molecule, per cm
5 A = [f0 2; f1/2 3];     // Declare the 2 X 2 matrix
     for multiplication
6 I = [1;1];     // Unity column matrix
7 X = inv(A)*I;     // Eigen value matrix
8 nu_e_bar = 1/X(1);     // Equilibrium vibrational
     frequency, per cm
9 x_e = X(2);     // Anharmonicity constant
10 E0 = 1/2*nu_e_bar;     // Zero point energy of the
     molecule, per cm
11 printf("\nThe equilibrium vibrational frequency = %7
     .2f per cm", nu_e_bar);
12 printf("\nThe anharmonicity constant = %4.2e", x_e);
13 printf("\nThe zero point energy of the molecule =
     %3d per cm", ceil(E0));
14
15 // Result
16 // The equilibrium vibrational frequency = 1903.98
     per cm
17 // The anharmonicity constant = 7.33e-003
18 // The zero point energy of the molecule = 952 per
     cm
```

Scilab code Exa 9.4 Force constant of the HCl bond

```
// Ex9_4 Page:175 (2014)
clc;clear;
PI = 3.14;
u = 1.67e-027;     // Mass equivalent of 1 amu, kg/
   amu
c = 3e+008;     // Speed of light in vacuum, m/s
lambda0 = 3.465e-006;     // Wavelength of
   vibrational absorption line of HCl molecule, m
m1 = 1.0087;     // Mass of H atom, amu
m2 = 35.453;     // Mass of Cl atom, amu
mu = m1*m2/(m1 + m2)*u;     // Reduced mass of HCl
   molecule, kg
k = 4*PI^2*mu*(c/lambda0)^2;     // Force constant of
    H-Cl bond, N/m
printf("\nForce constant of the H-Cl bond = %5.1f N/
   m", k);

// Result
// Force constant of the H-Cl bond = 484.2 N/m
```

Scilab code Exa 9.5 Vibrational frequency for Raman shift

```
// Ex9_5  Page:187 (2014)
clc;clear;
lambda_ex = 4358.3;     // Wavelength of the exciting
    radiation, angstrom
lambda_R = 4768.5;     // Wavelength of the Raman
   line, angstrom
nu_prime = 1/lambda_ex*1e+08;     // Wave number of
   the exciting radiation, per cm
nu_prime_R = 1/lambda_R*1e+08;     // Wave number of
   the Raman line, per cm
nu = nu_prime - nu_prime_R;     // Vibrational
   frequency of the sample, per cm
printf("\nThe vibrational frequency of the sample =
```

```
    %4d per cm", ceil(nu));
9
10 // Result
11 // The vibrational frequency of the sample = 1974
    per cm
```

Scilab code Exa 9.6 Bond length of hydrogen molecule

```
1  // Ex9_6  Page:188 (2014)
2  clc;clear;
3  h = 6.626e-034;     // Planck's constant, Js
4  c = 3e+08;     // Speed of light in vacuum, m/s
5  m = 1.673e-027;     // Mass of protium, kg
6  m1 = m;     // Mass of first hydrogen atom, kg
7  m2 = m;     // Mass of second hydrogen atom, kg
8  nu_prime = 346;     // Wave number of the first
    rotational Raman line of hydrogen, per cm
9  B = nu_prime*1e+02/6;     // Rotational constant of a
    hydrogen molecule, per m
10 I = h/(8*%pi^2*B*c);     // Moment of inertia of
    hydrogen molecule, kg Sq.m
11 mu = m1*m2/(m1 + m2);     // Reduced mass of hydrogen
    molecule, kg
12 r = sqrt(I/mu);     // Bond length of hydrogen
    molecule, m
13 printf("\nThe bond length of hydrogen molecule = %4
    .2e m", r);
14
15 // Result
16 // The bond length of hydrogen molecule = 7.62e-011
    m
```

Scilab code Exa 9.7 Magnetic field strength to observe an NMR spectrum

```
// Ex9_7  Page:193 (2014)
clc;clear;
h = 6.626e-034;     // Planck's constant, Js
nu = 120e+06;     // Operating frequency, Hz
g_N = 5.585;     // Nuclear g-factor
mu_N = 5.0508e-027;     // Nuclear magneton, J/T
B0 = h*nu/(g_N*mu_N);     // Magnetic field strength
   required to observe the NMR spectrum, T
printf("\nThe magnetic field strength required to
   observe the NMR spectrum = %5.3f T", B0);

// Result
// The magnetic field strength required to observe
   the NMR spectrum = 2.819 T
```

Scilab code Exa 9.8 Chemical shift in benzene and TMS

```
// Ex9_8  Page:194 (2014)
clc;clear;
h = 6.626e-034;     // Planck's constant, Js
B0 = 1.65;     // Magnetic field strength required to
    observe the NMR spectrum, T
g_N = 5.585;     // Nuclear g-factor
mu_N = 5.0508e-027;     // Nuclear magneton, J/T
nu = g_N*mu_N*B0/h;     // Operating frequency, Hz
d_nu = 510e+06;     // Frequency separation between
   protons in benzene and TMS, Hz
delta = d_nu/nu;     // Chemical shift in NMR
   spectrum, ppm
printf("\nChemical shift in NMR spectrum = %4.2f ppm
   ", delta);

// Result
// Chemical shift in NMR spectrum = 7.26 ppm
```

Scilab code Exa 9.10 Electron g factor in ESR resonance

```
1 // Ex9_10  Page:198 (2014)
2 clc;clear;
3 h = 6.626e-034;     // Planck's constant, Js
4 B0 = 1.3;     // External magnetic field, T
5 mu_B = 9.27e-024;     // Bohr's magneton, J/T
6 nu = 35e+009;     // Operating frequency, Hz
7 g = h*nu/(mu_B*B0);     // Electron g-factor
8 printf("\nThe electron g-factor for the unpaired
    electron = %5.3f", g);
9
10 // Result
11 // The electron g-factor for the unpaired electron =
     1.924
```

Chapter 10

Crystal Structure and Bonding

Scilab code Exa 10.1 Miller indices of the plane

```
1 // Ex10_1  Page:211 (2014)
2 clc;clear;
3 m = 2; n = 3; p = 1; // Coefficients of intercepts
     along the crystallographic axes
4 m_inv = 1/m;         // Reciprocal of the first
     coefficient
5 n_inv = 1/n;         // Reciprocal the second
     coefficient
6 p_inv = 1/p;         // Reciprocal the third
     coefficient
7 mul_fact = double(lcm(int32([m,n,p]))); // LCM of m,
     n and p
8 i1 = m_inv*mul_fact;     // Clear the first fraction
9 i2 = n_inv*mul_fact;     // Clear the second fraction
10 i3 = p_inv*mul_fact;    // Clear the third fraction
11 printf("\nThe Miller indices of the plane are (%d %d
     %d) ", i1, i2, i3);
12
13 // Result
14 // The Miller indices of the plane are (3 2 6)
```

Scilab code Exa 10.2 Miller indices of the plane parallel to z axis

```
1 // Ex10_2  Page:211 (2014)
2 clc;clear;
3 p = 2; q = 3/2; r = %inf; // Coefficients of
     intercepts along the x-, y- and z-axes
4 inv_p = 1/p;          // Reciprocal of the first
     coefficient
5 inv_q = 1/q;          // Reciprocal the second
     coefficient
6 inv_r = 1/r;          // Reciprocal the third
     coefficient
7 mul_fact = double(lcm(int32([p, q*2]))); // LCM of p
     and twice of q
8 m1 = inv_p*mul_fact;     // Clear the first fraction
9 m2 = inv_q*mul_fact;     // Clear the second fraction
10 m3 = inv_r*mul_fact;     // Clear the third fraction
11 printf("\nThe Miller indices of the plane parallel
     to z-axis are (%d %d %d) ", m1, m2, m3);
12
13 // Result
14 // The Miller indices of the plane parallel to z-
     axis are (3 4 0)
```

Scilab code Exa 10.6 Interplanar spacing of silver planes

```
1 // Ex10_6  Page:213 (2014)
2 clc;clear;
3 r = 0.152e-09;     // Atomic radius of silver, m
4 a = 4*r/sqrt(2);     // Lattice parameter for silver,
     m
5 // Case-I
```

```
h = 2; k = 3; l = 1; // Miller Indices for first set
    of planes
d_231 = a/(h^2+k^2+l^2)^(1/2);  // The interplanar
    spacing of (2 3 1) planes, m
printf("\nThe interplanar spacing of (2 3 1) planes
    = %6.4f nm", d_231/1e-09);
// Case-II
h = 1; k = 1; l = 0; // Miller Indices for second
    set of planes
d_110 = a/(h^2+k^2+l^2)^(1/2);  // The interplanar
    spacing of (1 1 0) planes, m
printf("\nThe interplanar spacing of (1 1 0) planes
    = %5.3f nm", d_110/1e-09);

// Result
// The interplanar spacing of (2 3 1) planes =
    0.1149 nm
// The interplanar spacing of (1 1 0) planes = 0.304
    nm
```

Scilab code Exa 10.7 Miller indices of the plane in a cubic crystal

```
// Ex10_7  Page:213 (2014)
clc;clear;
a = 0.424e-09;     // Lattice parameter of cubic
    crystal, m
p = 2; q = %inf; r = 1; // Coefficients of
    intercepts along the x-, y- and z-axes
inv_p = 1/p;          // Reciprocal of the first
    coefficient
inv_q = 1/q;          // Reciprocal the second
    coefficient
inv_r = 1/r;          // Reciprocal the third
    coefficient
mul_fact = double(lcm(int32([p, r]))); // LCM of p
```

```
    and r
h = inv_p*mul_fact;     // Clear the first fraction
k = inv_q*mul_fact;     // Clear the second fraction
l = inv_r*mul_fact;     // Clear the third fraction
d_102 = a/(h^2+k^2+l^2)^(1/2);  // The interplanar
    spacing of (1 0 2) planes, m
printf("\nThe Miller indices are (%d %d %d) ", h, k,
    l);
printf("\nThe interplanar spacing = %6.4f nm", d_102
    /1e-09);

// Result
// The Miller indices are (1 0 2)
// The interplanar spacing = 0.1896 nm
```

Scilab code Exa 10.8 Number of atoms per unit cell of lead

```
// Ex10_8  Page:214 (2014)
clc;clear;
a = 3.2e-10;     // Lattice parameter for lead, m:
M = 207.2;         // Atomic weight of Pb, gram per
    mole
rho = 11.36e+03;     // Density of Pb, kg per metre
    cube
N = 6.023D+26;     // Avogadro's No., per k-mol
// Volume of the unit cell is given by
// a^3 = M*n/(N*rho)
// Solving for n
n = a^3*rho*N/M; // Number of atoms per unit cell
printf("\nThe number of atoms per unit cell for an
    fcc lattice of lead = %d", n);

// Result
// The number of atoms per unit cell for an fcc
    lattice of lead = 1
```

Scilab code Exa 10.9 Diffraction of X rays from a crystal

```
1 // Ex10_9  Page:220 (2014)
2 clc;clear;
3 d = 2.51e-010;     // Spacing between adjacent planes
     , m
4 theta = 9;     // Glancing angle for diffraction ,
     degree
5 n = 1;     // Order of diffraction
6 lambda = 2*d*sind(theta)/n;     // Wavelength of X-
     ray from Bragg's Law, m
7 n = 2;     // New order of diffraction
8 theta = asind(2*lambda/(2*d));     // Glancing angle
     for second order diffraction , degree
9 printf("\nThe wavelength of X-rays = %6.4f angstrom"
     , lambda/1e-010);
10 printf("\nThe glancing angle for second order
     diffraction = %2d degree", theta);
11
12 // Result
13 // The wavelength of X-rays = 0.7853 angstrom
14 // The glancing angle for second order diffraction =
      18 degree
```

Scilab code Exa 10.10 Angle of incidence of X rays on a crystal plane

```
1 // Ex10_10  Page:220 (2014)
2 clc;clear;
3 lambda = 1.4e-010;     // Wavelength of X-rays , m
4 a = 5e-010;     // Lattice parameter , m
5 h = 1, k = 1, l = 1;     // Miller indices of planes
     from which the reflection occurs
```

```
6 d_111 = a/sqrt(3);    // Interplanar spacing between
      (1 1 1) planes, m
7 n = 1;    // Order of diffraction
8 theta_111 = asind(n*lambda/(2*d_111));    // Angle
    at which the X-ray is incident on (1 1 1) plane
    of the crystal, degree
9 printf("\nThe angle at which the X-ray is incident
    on (1 1 1) plane = %2d degree", theta_111);
10
11 // Result
12 // The angle at which the X-ray is incident on (1 1
    1) plane = 14 degree
```

Scilab code Exa 10.11 Lattice parameters of an fcc crystal

```
1 // Ex10_11  Page:221 (2014)
2 clc;clear;
3 h = 6.626e-034;    // Planck's constant, Js
4 e = 1.6e-019;    // Charge on an electron, C
5 m = 9.1e-031;    // Mass of an electron, kg
6 V = 120;    // Accelerating potential, volt
7 theta = 22;    // The angle at which the reflection
    maximum is observed, degree
8 lambda = h/sqrt(2*m*e*V);    // Wavelength of a
    moving electron, m
9 h = 1, k = 1, l = 1;    // Miller indices of planes
    from which the reflection occurs
10 n = 1;    // Order of diffraction
11 d_111 = n*lambda/(2*sind(theta)*1e-010);    //
    Interplanar spacing between (1 1 1) planes, m
12 a = sqrt(3)*d_111;    // Lattice parameter, m
13 printf("\nThe lattice parameter = %5.3f angstrom", a
    );
14
15 // Result
```

```
16 // The lattice parameter = 2.591 angstrom
17 // The answers vary due to round off error
```

Scilab code Exa 10.12 Potential energy of KCl system

```
1 // Ex10_12 Page:224 (2014)
2 clc;clear;
3 e = 1.6e-019;     // Charge on an electron, C
4 epsilon_0 = 8.85e-012;     // Absolute electric
    permittivity of free space, coulomb-square/N/Sq.m
5 r_0 = 0.32e-009;     // Inter-ionic distance of KCl,
    m
6 V = -e/(4*3.14*epsilon_0*r_0);     // Potential
    energy of K+ and Cl- ions, eV
7 printf("\nThe potential energy of K+ and Cl- ions =
    %5.3f eV", V);
8
9 // Result
10 // The potential energy of K+ and Cl- ions = -4.498
    eV
```

Scilab code Exa 10.13 Cohesive energy of NaCl

```
1 // Ex10_13 Page:224 (2014)
2 clc;clear;
3 e = 1.6e-019;     // Charge on an electron, C
4 epsilon_0 = 8.85e-012;     // Absolute electric
    permittivity of free space, coulomb-square/N/Sq.m
5 r_0 = 0.31e-009;     // Equilibrium separation of Na+
    and Cl- ions, m
6 alpha = 1.748;     // Madelung constant
7 n = 9;     // Repulsive exponent
8 E_ion =  5;     // Ionization energy of NaCl, eV
```

```
9  V = -1*alpha*e^2/(4*3.14*epsilon_0*r_0*e)*(1-1/n);
       // Potential energy of Na+ and Cl- ions, eV
10 E_ele = 1/2*V;     // Electron affinity, eV
11 E_trans = E_ion + E_ele;     // Electron transfer
   energy, eV
12 delta_E = E_trans/2;     // Contribution per ion to
   the cohesive energy, eV
13 E_cohesive = E_ele + delta_E;     // Cohesive energy
   per NaCl atom, eV
14 printf("\nThe cohesive energy per NaCl atom = %5.3f
   eV", E_cohesive);
15
16 // Result
17 // The cohesive energy per NaCl atom = -2.911 eV
18 // The answers vary due to round off error
```

Chapter 11

Lattice Dynamics

Scilab code Exa 11.1 Highest possible frequency of Cu and Si

```
1 // Ex11_1 Page:238 (2014)
2 clc;clear;
3 h = 6.626e-034;     // Planck's constant, Js
4 k = 1.38e-023;     // Boltzmann constant, J/K
5 // Case-I: For Cu
6 theta_D = 350;     // Debye temperature for Cu, K
7 nu_D = k*theta_D/h;     // The highest possible
     frequency for Cu, per sec
8 printf("\nThe highest possible frequency for Cu = %6
     .3fe+011 per sec", nu_D/1e+011);
9 // Case-II: For Si
10 theta_D = 550;     // Debye temperature for Si, K
11 nu_D = k*theta_D/h;     // The highest possible
     frequency for Si, per sec
12 printf("\nThe highest possible frequency for Si = %6
     .2fe+011 per sec", nu_D/1e+011);
13
14 // Result
15 // The highest possible frequency for Cu = 72.895e
     +011 per sec
16 // The highest possible frequency for Si = 114.55e
```

```
+011 per sec
```

Scilab code Exa 11.2 Specific heat and highest frequency in lead

```
1 // Ex11_2 Page:238 (2014)
2 clc;clear;
3 h = 6.626e-034;     // Planck's constant, Js
4 N = 6.02e+026;     // Avogadro's number, per k-mol
5 k = 1.38e-023;     // Boltzmann constant, J/K
6 T = 10;     // Temperature of Lead, K
7 theta_D = 105;     // Debye temperature of lead, K
8 C = 12/5*3.14^4*N*k*(T/theta_D)^3;     // Specific
    heat of Pb from Debye T-cube law, J/K/k-mol
9 printf("\nThe specific heat of Pb = %6.1f J/K/k-mol"
    , C);
10 nu_D = k*theta_D/h;     // The highest possible
    frequency for Cu, per sec
11 printf("\nThe highest frequency for lead = %5.2fe
    +011 per sec", nu_D/1e+011);
12
13 // Result
14 // The specific heat of Pb = 1674.3 J/K/k-mol
15 // The highest frequency for lead = 21.87e+011 per
    sec
16 // The answers vary due to round off error
```

Chapter 12

Band Theory of Solids

Scilab code Exa 12.1 Speed of the electron at its mean energy at 0K

```
1 // Ex12_1 Page:243 (2014)
2 clc; clear;
3 E_F = 8;     // Fermi energy, eV
4 E0_bar = 3/5*E_F;    // Average kinetic energy of a
     free electron gas at 0K
5 e = 1.6E-019;    // Energy equivalent of 1 eV, J
6 m = 9.1E-031;    // Mass of an electron, kg
7 v = sqrt(2*E0_bar*e/m); // Speed of the electron, m/
     s
8 printf("\nThe speed of the electron at its mean
     energy at 0K = %3.1e m/s", v);
9
10 // Result
11 // The speed of the electron at its mean energy at 0
     K = 1.3e+06 m/s
```

Scilab code Exa 12.2 Current density and resistivity of Cu

```
// Ex12_2 Page:244 (2014)
clc; clear;
I = 8;  // Current through the copper wire, A
d = 18e-03; // Diameter of the copper wire, m
V = 5;      // Potential difference across the wire, V
L = 1;  // Length of the copper wire, m
A = %pi*(d/2)^2;    // Area of cross-section of the wire, Sq.m
J = I/A;    // Current density in the wire, A/Sq.m
E = V/L;    // Electric field across the wire, V/m
rho = E/J;  // Resistivity of the material, ohm-m
printf("\nThe current density in the copper wire = %4.2e A/Sq.m", J);
printf("\nThe resistivity of the material = %4.2e ohm-m", rho);

// Result
// The current density in the copper wire = 3.14e+04 A/Sq.m
// The resistivity of the material = 1.59e-04 ohm-m
// The answer provided in the textbook is wrong
```

Scilab code Exa 12.3 Conductivity and resistivity of Cu

```
// Ex12_3 Page:245 (2014)
clc; clear;
tau = 1e-015;  // Average collision time of an electron, s
a = 4e-010; // Lattice parameter of cubic structure, m
n = 1;  // Number of electrons atoms per unit cell
e = 1.6e-019;   // Charge on an electron, C
m = 9.1e-031;   // Mass of an electron, kg
N = n/a^3;  // Number of electrons per unit volume,
```

```
    per metre-cube
sigma = N*e^2*tau/m;  // Conductivity of copper, mho
    /m
rho = 1/sigma; // Resistivity of copper, ohm-m
printf("\nThe conductivity of copper = %4.2e mho/m",
     sigma);
printf("\nThe resistivity of copper = %4.2e ohm-m",
    rho);

// Result
// The conductivity of copper = 4.40e+05 mho/m
// The resistivity of copper = 2.28e-06 ohm-m
```

Scilab code Exa 12.4 Electronic specific heat of potassium

```
// Ex12_4 Page:247 (2014)
clc; clear;
e = 1.6e-019;   // Energy equivalent of 1 eV, J
k = 1.38e-023;  // Boltzmann constant, J/K
N_A = 6.023e+026;    // Avogadro's number/k-mol
T = 27+273;    // Room temperature, K
E_F = 2;    // Fermi energy
C = (%pi^2*k^2*N_A*T)/(2*E_F*e);    // Electronic
    specific heat of potassium, J/k-mol/K
printf("\nThe electronic specific heat of potassium
    = %3d J/k-mol/K", C);

// Result
// The electronic specific heat of potassium = 530 J
    /k-mol/K
```

Scilab code Exa 12.5 Lorentz number of zinc

```
// Ex12_5 Page:247 (2014)
clc; clear;
e = 1.6e-019;    // Charge on an electron, C
m = 9.1e-031;    // Mass of an electron, kg
K = 327;     // Thermal conductivity of Zn, W/m/K
N_A = 6.023e+026;     // Avogadro's number/k-mol
T = 300;     // Room temperature, K
tau = 2.5e-014; // Relaxation time of electrons in
   divalent Zn, s
d = 7.13e+03;    // Density of Zn, kg/metre-cube
A = 65.38;  // Atomic weight of Zn, g/mol
N = 2*d*N_A/A;  // Number of electrons per unit
   volume, per metre-cube
sigma = N*e^2*tau/m;     // Electrical conductivity
   of Zn,  mho/m
L = K/(sigma*T);     // Lorentz number for Zn, W-ohm/
   Sq.K
printf("\nLorentz number for Zn = %5.3e W-ohm/Sq.K",
    L);

// Result
// Lorentz number for Zn = 1.180e-08 W-ohm/Sq.K
```

Scilab code Exa 12.6 Hall coefficient of Cu

```
// Ex12_6 Page:248 (2014)
clc; clear;
e = 1.6e-019;    // Charge on an electron, C
n = 5e+028; // Number of atoms per unit volume of Cu
   , per metre-cube
R_H = -1/(n*e); // Hall coefficient, metre-cube/C
printf("\nThe Hall coefficient for Cu = %4.2e metre-
   cube/C", R_H);

// Result
```

```
9 // The Hall coefficient for Cu = -1.25e-10 metre-
     cube/C
```

Chapter 13

Magnetic Properties of Solids

Scilab code Exa 13.1 Magnetism in Si

```
1 // Ex13_1 Page:256 (2014)
2 clc;clear;
3 H = 1.2e+05;     // Magnetic field in silicon, A/m
4 chi = -4.2e-006;     // Magnetic susceptibility
5 mu_0 = 4*%pi*1e-007;     // Magnetic permeability, T-
    m/A
6 M = chi*H;     // Magnetization of Si, A/m
7 B = mu_0 *(H + M);     // Magnetic flux density in Si
    , T
8 mu_r = M/H + 1;     // Relative permeability of the
    material
9 printf("\nThe magnetization of Si = %5.3f A/m", M);
10 printf("\nThe magnetic flux density in Si = %5.3f T"
    , B);
11 printf("\nThe relative permeability of the material
    = %f", mu_r);
12
13 // Result
14 // The magnetization of Si = -0.504 A/m
15 // The magnetic flux density in Si = 0.151 T
16 // The relative permeability of the material =
```

```
0.999996
```

Scilab code Exa 13.2 Diamagnetic susceptibility of He

```
1 // Ex13_2 Page:258 (2014)
2 clc;clear;
3 mu_0 = 4*%pi*1e-007;     // Magnetic permeability, T-
    m/A
4 e = 1.6e-019;     // Charge on an electron, C
5 m = 9.1e-031;     // Mass of an electron, kg
6 Z = 2;     // Atomic number of He
7 N = 28e+026;     // Number of He atoms per unit
    volume of the sample, per metre-cube
8 r_bar = 0.6e-010;     // Mean radius of He atom, m
9 chi_dia = -mu_0*Z*e^2*N*r_bar^2/(6*m);     //
    Diamagnetic susceptibility of He
10 printf("\nThe diamagnetic susceptibility of He = %5
    .3e", chi_dia);
11
12 // Result
13 // The diamagnetic susceptibility of He = -1.188e
    -007
```

Scilab code Exa 13.3 Radius of an atom of a diamagnetic material

```
1 // Ex13_3 Page:259 (2014)
2 clc;clear;
3 mu_0 = 4*%pi*1e-007;     // Magnetic permeability, H/
    m
4 e = 1.6e-019;     // Charge on an electron, C
5 m = 9.1e-031;     // Mass of an electron, kg
6 Z = 1;     // Atomic number of the material
```

```
7 a = 2.55e-010;     // Lattice constant of cubic
    structure, m
8 chi_dia = -5.6e-006;     // Diamagnetic
    susceptibility of the material
9 N = 2/a^3;     // Number of atoms per unit volume of
    the material, per metre-cube
10 r_bar = sqrt(abs(chi_dia)*6*m/(mu_0*Z*e^2*N));     //
    Radius of an atom of the material, m
11 printf("\nThe radius of an atom of the material = %5
    .3f angstrom", r_bar/1e-010);
12
13 // Result
14 // The radius of an atom of the material = 0.888
    angstrom
```

Scilab code Exa 13.4 Susceptibility of a paramagnetic salt at 300 K

```
1 // Ex13_4 Page:260 (2014)
2 clc;clear;
3 mu_0 = 4*%pi*1e-007;     // Magnetic permeability, H/
    m
4 N = 6.5e+025;     // Number of atoms per unit volume
    of the salt, per metre-cube
5 mu = 9.27e-024;     // Bohr magneton, A-Sq.m
6 k = 1.38e-023;     // Boltzmann constant, J/K
7 T = 300;     // Absolute temperature of the specimen,
    K
8 chi_para = mu_0*N*mu^2/(3*k*T);     // Susceptibility
    of paramagnetic salt
9 printf("\nThe susceptibility of paramagnetic salt =
    %4.2e", chi_para);
10
11 // Result
12 // The susceptibility of paramagnetic salt = 5.65e
    -007
```

Scilab code Exa 13.5 Susceptibity and magnetization produced in paramagnetic salt

```
1 // Ex13_5 Page:260 (2014)
2 clc;clear;
3 mu_0 = 4*3.14*1e-007;     // Magnetic permeability, H
    /m
4 mu = 9.27e-024;     // Bohr magneton, A-Sq.m
5 k = 1.38e-023;     // Boltzmann constant, J/K
6 T = 300;     // Absolute temperature of the specimen,
     K
7 H = 2e+005;     // Magnetic field to which
    paramagnetic salt is subjected, A/m
8 rho = 4370;     // Density of paramagnetic salt, kg/
    metre-cube
9 N_A = 6.021e+026;     // Avogadro's number, per k-mol
10 M = 168.5;     // Molecular weight of paramagnetic
    salt, kg/k-mol
11 N = rho*N_A/M;     // Number of atoms per unit volume
     of the salt, per metre-cube
12 chi_para = mu_0*N*(2*mu)^2/(3*k*T);     //
    Susceptibility of paramagnetic salt
13 M = chi_para*H;     // Magnetization produced in
    paramagnetic salt, A/m
14 printf("\nThe susceptibility of paramagnetic salt =
    %5.3e", chi_para);
15 printf("\nThe magnetization produced in paramagnetic
     salt = %6.2f A/m", M);
16
17 // Result
18 // The susceptibility of paramagnetic salt = 5.428e
    -004
19 // The magnetization produced in paramagnetic salt =
     108.56 A/m
```

Chapter 14

Superconductivity

Scilab code Exa 14.1 Current through a tin wire at 2 K

```
1 // Ex14_1 Page:272 (2014)
2 clc;clear;
3 mu_0 = 4*%pi*1e-007;     // Absolute magnetic
     permeability of free space, N/Sq.A
4 r = 2e-003;     // Radius of tin wire, m
5 T = 2;     // Temperature of tin wire, K
6 T_c = 3.722;     // Critical temperature of tin, K
7 B_c0 = 0.0305;     // Critical field at 0 K, T
8 B_cT = B_c0*(1-(T/T_c)^2);     // Critical field at T
     K, T
9 I_c = 2*%pi*r*B_cT/mu_0;     // Critical current
     through the tin wire from Ampere's Law, A
10 printf("\nThe critical current through the tin wire
     = %5.1f A", I_c);
11
12 // Result
13 // The critical current through the tin wire = 216.9
     A
```

Scilab code Exa 14.2 Penetration depth of lead below room temperature

```
1 // Ex14_2 Page:274 (2014)
2 clc;clear;
3 T = 5.2;     // Temperature of lead, K
4 T_c = 7.193;     // Critical temperature of lead, K
5 lambda_0 = 37;     // London penetration depth at 0 K
    , nm
6 lambda_T = lambda_0*(1-(T/T_c)^4)^(-1/2);     //
    Penetration depth of lead at T K, nm
7 printf("\nThe penetration depth of lead at %3.1f K =
    %5.2f nm", T, lambda_T);
8
9 // Result
10 // The penetration depth of lead at 5.2 K = 43.40 nm
```

Chapter 15

Lasers

Scilab code Exa 15.2 Energy and number of photons per pulse of a laser

```
1 // Ex15_2 Page:298 (2014)
2 clc;clear;
3 h = 6.626e-034;     // Planck's constant, Js
4 c = 3e+008;     // Speed of light in vacuum, m/s
5 t = 30e-003;     // Pulse width of laser, s
6 P = 0.6;     // Output power of laser per pulse, W
7 lambda = 640e-009;     // Wavelength of laser light,
    m
8 E = P*t;     // Energy deposited per laser pulse, J
9 n = E*lambda/(h*c);     // Number of photons in each
    laser pulse
10 printf("\nThe energy deposited per laser pulse = %5
    .3f J", E);
11 printf("\nThe number of photons in each pulse = %3.1
    e", n);
12
13 // Result
14 // The energy deposited per laser pulse = 0.018 J
15 // The number of photons in each pulse = 5.8e+016
```

Scilab code Exa 15.3 Area and intensity of the image

```
1 // Ex15_3 Page:298 (2014)
2 clc;clear;
3 P = 2.5e-003;     // Output power of laser source, W
4 d = 1.8e-02;     // Diameter of the aperture, m
5 a = d/2;     // Radius of the beam, m
6 lambda = 5000e-010;     // Wavelength of laser light,
    m
7 f = 20e-002;     // Focal length of the lens, m
8 A = %pi*(lambda*f/a)^2;     // Area of the spot at
    the focal plane, Sq.m
9 I = P*(a/(lambda*f))^2/%pi;     // Intensity of the
    image, W/Sq.m
10 printf("\nThe area of the spot at the focal plane =
    %4.2e Sq.m", A);
11 printf("\nThe intensity of the image = %4.2e W/Sq.m"
    , I);
12
13 // Result
14 // The area of the spot at the focal plane = 3.88e
    -010 Sq.m
15 // The intensity of the image = 6.45e+006 W/Sq.m
```

Scilab code Exa 15.4 Spread of laser beam

```
1 // Ex15_4 Page:298 (2014)
2 clc;clear;
3 lambda = 693e-009;     // Wavelength of laser beam, m
4 D = 3e-003;     // Diameter of the mirror, m
5 d = 300e+003;     // Height of satellite above the
    surface of earth, m
```

```
6  delta_theta = 1.22*lambda/D;     // Angular spread of
       the laser beam, rad
7  a = delta_theta*d;     // Diameter of the laser beam
     on the satellite, m
8  printf("\nThe angular spread of the laser beam = %4
     .2e rad", delta_theta);
9  printf("\nThe diameter of laser beam on the
     satellite = %4.1f m", a);
10
11 // Result
12 // The angular spread of the laser beam = 2.82e-004
     rad
13 // The diameter of laser beam on the satellite =
     84.5 m
```

Chapter 16

Fibre Optics and Holography

Scilab code Exa 16.1 Parameters of a step index fibre

```
1 // Ex16_1 Page:306 (2014)
2 clc;clear;
3 n1 = 1.43;     // Refractive index of core
4 n2 = 1.40;     // Refractive index of cladding
5 theta_c = acosd(n2/n1);     // Propagation angle,
     degree
6 NA = sqrt(n1^2 - n2^2);     // Numerical aperture
7 theta_a = asind(NA);     // Acceptance angle, degree
8 printf("\nPropagation angle = %4.1f degree", theta_c
     );
9 printf("\nNumerical aperture = %6.4f", NA);
10 printf("\nAcceptance angle = %5.2f degree", 2*
     theta_a);
11
12 // Result
13 // Propagation angle = 11.8 degree
14 // Numerical aperture = 0.2914
15 // Acceptance angle = 33.88 degree
```

Scilab code Exa 16.3 Output power from an optical fibre

```
1 // Ex16_3 Page:311 (2014)
2 clc;clear;
3 z = 30;     // Length of the optical fibre, km
4 alpha = 0.8;     // Fibre loss, dB/km
5 P_i = 200;     // Power input to the optical fibre,
    micro-watt
6 P_o = P_i/10^(alpha*z/10);     // Output power of the
    optical fibre, micro-watt
7 printf("\nThe output power from the optical fibre =
    %5.3f micro-watt", P_o);
8
9 // Result
10 // The output power from the optical fibre = 0.796
    micro-watt
```

Chapter 17

Nuclear Properties

Scilab code Exa 17.1 Density of a nucleus

```
1 // Ex17_1 Page:324 (2014)
2 clc;clear;
3 A = poly (0, 'A');     // Declare the mass number
     variable
4 m_n = 1.67e-027;     // Nucleon mass, kg
5 R0 = 1.2e-015;     // Nuclear constant, m
6 d = m_n*A/(4/3*%pi*R0^3*A);
7 printf("\nDensity of the nucleus = %3.1e km/metre-
     cube", horner(d,0));
8
9 // Result
10 // Density of the nucleus = 2.3e+017 km/metre-cube
```

Scilab code Exa 17.2 Kinetic energy of the alpha particle

```
1 // Ex17_2 Page:324 (2014)
2 clc;clear;
3 e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
```

```
4 R0 = 1.2e-015;     // Nuclear constant, m
5 k = 9e+09;     // Coulomb's constant, N-Sq.m/Sq.C
6 q1 = 2*1.6e-019;     // Charge on alpha particle, C
7 q2 = 90*1.6e-019;     // Charge on thorium nucleus, C
8 A_alpha = 4;     // Mass number of helium nucleus
9 A_Th = 228;     // Mass number of thorium
10 r = R0*(A_alpha^(1/3) + A_Th^(1/3));     // Distance
    between He and Th, m
11 PE = k*q1*q2/(r*e)*1e-006;     // Potential energy of
    the system, MeV
12 printf("\nThe kinetic energy of the alpha particle
    when it is far away = %4.1f MeV", PE);
13
14 // Result
15 // The kinetic energy of the alpha particle when it
    is far away = 28.1 MeV
```

Scilab code Exa 17.3 Separation between spectral lines in a mass spectrograph

```
1 // Ex17_3 Page:326 (2014)
2 clc;clear;
3 e = 1.6e-019;     // Charge on an electron, C
4 E = 2.48e+004;     // Applied electric field, N/C
5 B = 0.75;     // Applied magnetic field, T
6 R0 = 1.2e-015;     // Nuclear constant, m
7 m1 = 12*1.6605e-027;     // Mass of the C-12 ion
8 m2 = 13*1.6605e-027;     // Mass of the C-13 ion
9 m3 = 14*1.6605e-027;     // Mass of the C-14 ion
10 r1 = E*m1/(e*B^2*1e-003);     // Radius of orbit of C
    -12 ion, mm
11 r2 = E*m2/(e*B^2*1e-003);     // Radius of orbit of C
    -13 ion, mm
12 r3 = E*m3/(e*B^2*1e-003);     // Radius of orbit of C
    -14 ion, mm
13 d1 = 2*r2 - 2*r1;     // Distance between the lines
```

```
    of C-12 and C-13, mm
14 d2 = 2*r3 - 2*r2;     // Distance between the lines
    of C-13 and C-14, mm
15 d = d1/2;     // Separations between the lines when
    the ions are doubly charged, mm
16 printf("\nThe distance between the lines of C-12 and
    C-13 = %4.2f mm", d1);
17 printf("\nThe distance between the lines of C-13 and
    C-14 = %4.2f mm", d2);
18 printf("\nThe separation between the lines when the
    ions are doubly charged = %4.2f mm", d);
19
20 // Result
21 // The distance between the lines of C-12 and C-13 =
    0.92 mm
22 // The distance between the lines of C-13 and C-14 =
    0.92 mm
23 // The separation between the lines when the ions
    are doubly charged = 0.46 mm
```

Scilab code Exa 17.4 Binding energy and average binding energy per nucleon of C12

```
1  // Ex17_4 Page:327 (2014)
2  clc;clear;
3  m_n = 1.008665;     // Mass of a neutron, amu
4  m_p = 1.007825;     // Mass of a proton, amu
5  A = 12;     // Mass number of carbon, amu
6  Z = 6;     // Atomic number of carbon, amu
7  N = A - Z;     // Neutron number of carbon, amu
8  M_C12 = 12.0;     // Mass of C-12 atom, amu
9  M = Z*m_p + N*m_n;     // Total mass of constituents,
    amu
10 delta_m = M - M_C12;     // Mass deficiency, amu
11 E_B = delta_m * 931.5;     // Binding energy, MeV
12 E_B_avg = E_B/A;     // Average binding eneryg per
```

```
    nucleon , MeV
13 printf("\nThe binding energy of C(12,6) = %5.2f MeV"
    , E_B);
14 printf("\nThe average binding energy per nucleon of
    C(12,6) = %3.2f MeV", E_B_avg);
15
16 // Result
17 // The binding energy of C(12,6) = 92.16 MeV
18 // The average binding energy per nucleon of C(12,6)
    = 7.68 MeV
```

Scilab code Exa 17.6 Binding energy of the last neutron in Na

```
1 // Ex17_6 Page:335 (2014)
2 clc;clear;
3 m_n = 1.008665;     // Mass of a neutron, amu
4 M_Na22 = 21.9944;     // Mass of Na-22 atom, amu
5 M_Na23 = 22.989767;     // Mass of Na-23 atom, amu
6 delta_m = M_Na22 + m_n - M_Na23;     // Mass
    deficiency with Na-23, amu
7 E_B = delta_m*931.5;     // Energy equivalent of mass
    deficiency, MeV
8 printf("\nThe binding energy of the last neutron in
    Na-23 = %4.1f MeV", E_B);
9
10 // Result
11 // The binding energy of the last neutron in Na-23 =
    12.4 MeV
```

Scilab code Exa 17.7 Range of nuclear force

```
1 // Ex17_7 Page:341 (2014)
2 clc;clear;
```

```
3  h_cross = 1.05e-034;     // Reduced Planck's constant
       , Js
4  e = 1.6e-019;     // Charge on an electron , C
5  m_pi = 140e+006;     // Mass of pi_meson , eV/Sq.c
6  c = 3e+008;     // Speed of light in vacuum , m/s
7  r = h_cross*c/(m_pi*e);     // Range of the nuclear
       force , m
8  printf("\nThe range of the nuclear force = %3.1f fm"
       , r/1e-015);
9
10 // Result
11 // The range of the nuclear force = 1.4 fm
```

Chapter 18

Radioactive Decay

Scilab code Exa 18.1 Fraction of samples remained after various half lives

```
1 // Ex18_1 Page:347 (2014)
2 clc;clear;
3 N0 = poly(0, 'N0');     // Declare the original
     number of nuclides
4 HL = [2 5 10];     // Half lives, time units
5 for i = 1:3
6 printf("\nAfter %d half lives, the fraction remains
     = 1/%d", HL(i), 2^HL(i));
7 end
8
9 // Result
10 // After 2 half lives, the fraction remains = 1/4
11 // After 5 half lives, the fraction remains = 1/32
12 // After 10 half lives, the fraction remains =
     1/1024
```

Scilab code Exa 18.2 Radioactivity of gold

```
// Ex18_2 Page:348 (2014)
clc;clear;
t_half = 2.7*24*60*60;      // Half life of Au-198, s
lambda = 0.693/t_half;      // Decay constant of Au
    -198, per sec
M = 198;      // Molar mass of Au-198, g
m = 1e-006;      // Mass of Au-198 sample, g
N_A = 6.023e+023;      // Avogadro number, atoms/mol
t = 8*24*60*60;      // Age of the sample, s
N = m*N_A/M;      // Number of nuclei in the sample
A0 = lambda*N;      // Activity of Au-198, Ci
A = A0*exp(-lambda*t);      // Activity of the 8 days
    old sample, decays per sec
printf("\nThe decay constant of Au-198 = %4.2e per
    sec", lambda);
printf("\nThe activity of Au-198 = %5.3f Ci", A0/3.7
    e+010);
printf("\nThe activity of the 8 days old sample of
    Au-198 = %4.2e decays per sec", A);

// Result
// The decay constant of Au-198 = 2.97e-006 per sec
// The activity of Au-198 = 0.244 Ci
// The activity of the 8 days old sample of Au-198 =
     1.16e+009 decays per sec
```

Scilab code Exa 18.3 Mass of 2 mCi of radioactive C14

```
// Ex18_3 Page:348 (2014)
clc;clear;
N_A = 6.02e+023;      // Avogadro number, atoms/mol
M = 14;      // Gram atomic mass of C-14, g
t_half = 5570*365*24*60*60;      // Half life of C-14,
    s
lambda = 0.693/t_half;      // Decay constant of Au
```

```
    -198, per sec
A = 2e-003*3.7e+010;     // Activity of C-14, decays
    per sec
N = 1/lambda*A;     // The number of nuclei in the C
    -14 sample
m = N*M/N_A;     // Mass of N atoms of C-14, g
printf("\nThe mass of 2 mCi of radioactive C-14 = %4
    .2e g", m);

// Result
// The mass of 2 mCi of radioactive C-14 = 4.36e-004
    g
```

Scilab code Exa 18.5 Age of the rock on moon

```
// Ex18_5 Page:353 (2014)
clc;clear;
N1 = 1;     // Assume the number of present atoms of
    K-40
N2 = 10.2;     // No.of atoms of Ar-40 relative to K
    -40
t_half = 1.25e+009;     // Half-life of K-40, years
lambda = 0.693/t_half;     // Decay constant, per sec
t = log(1 + N2/N1)/lambda;     // Age of the rock on
    moon, years
printf("\nThe age of the rock on moon = %4.2e yr", t
    );

// Result
// The age of the rock on moon = 4.36e+009 yr
```

Scilab code Exa 18.6 Atomic mass of Th 228

```
// Ex18_6 Page:356 (2014)
clc;clear;
m_U232 = 232.037131;     // Atomic mass of U-232, u
m_He4 = 4.002603;     // Atomic mass of He-4, u
KE_alpha = 5.32;     // Kinetic energy of alpha-
    particle, MeV
m_Th228 = m_U232 - m_He4 - KE_alpha/931.5;     //
    Atomic mass of Th-228, u
printf("\nThe atomic mass of Th-228 = %10.6f u",
    m_Th228);

// Result
// The atomic mass of Th-228 = 228.028817 u
// The answers vary due to round off error
```

Scilab code Exa 18.7 Maximum and minimum energy of neutrino in C11 decay

```
// Ex18_7 Page:359 (2014)
clc;clear;
m_AX = 11.011433;     // Mass of parent nucleus, MeV
m_AX_prime = 11.009305;     // Mass of daughter
    nucleus, MeV
m_e = 0.511;     // Mass of an electron, MeV
Q_min = 0;
Q = (m_AX - m_AX_prime)*931.5 - 2*m_e;     // Q
    value for the decay, MeV
printf("\nThe maximum energy of the neutrino = %4.2f
    MeV", Q);
printf("\nThe minimum energy of the neutrino = %d
    MeV", Q_min);

// Result
// The maximum energy of the neutrino = 0.96 MeV
// The minimum energy of the neutrino = 0 MeV
```

Scilab code Exa 18.8 Kinetic energy of the neutrino

```
1 // Ex18_8 Page:359 (2014)
2 clc;clear;
3 m_K40 = 39.963999;     // Mass K-40 nucleus, MeV
4 m_Ar40 = 39.962384;     // Mass Ar-40 nucleus, MeV
5 Q = (m_K40 - m_Ar40)*931.5;      // Q value for the
    reaction, MeV
6 printf("\nThe kinetic energy of the neutrino = %5.3f
    MeV", Q);
7
8 // Result
9 // The kinetic energy of the neutrino = 1.504 MeV
```

Scilab code Exa 18.9 Maximum kinetic energy of the beta particle

```
1 // Ex18_9 Page:360 (2014)
2 clc;clear;
3 m_N12 = 12.018613;     // Mass N-12 nucleus, MeV
4 m_C12 = 12;     // Mass C-12 nucleus, MeV
5 m_e = 0.000549;     // Mass of the electron, MeV
6 E_gamma = 4.43;     // Energy of the emitted gamma
    ray. MeV
7 Q = (m_N12 - m_C12 - 2*m_e)*931.5;      // Q value
    for the reaction, MeV
8 E_max = Q - E_gamma;     // The maximum kinetic
    energy of the beta particle, MeV
9 printf("\nThe maximum kinetic energy of the beta
    particle = %5.2f MeV", E_max);
10
11 // Result
```

```
12 // The maximum kinetic energy of the beta particle =
       11.89 MeV
```

Chapter 19

Nuclear Reactions

Scilab code Exa 19.1 Q value of the reaction and KE of the residual nucleus

```
1 // Ex19_1 Page:368 (2014)
2 clc;clear;
3 K_x = 10;     // Energy of incident deutrons, MeV
4 K_y = 15;     // Energy of emitted neutrons. MeV
5 m_H2 = 2.014102;     // Atomic mass of deutrium, u
6 m_n = 1.008665;     // Mass of a neutron, u
7 m_Cu63 = 62.929599;     // Atomic mass of Cu-63, u
8 m_Zn64 = 63.929144;     // Atomic mass of Zn-64, u
9 Q = (m_H2 + m_Cu63 - m_n - m_Zn64)*931.5;     // Q-
    value of the reaction, MeV
10 K_Y = Q + K_x - K_y;     // The KE of the residual
    nucleus, MeV
11 printf("\nThe Q-value of the reaction = %5.3f MeV",
    Q);
12 printf("\nThe KE of the residual nucleus = %5.3f MeV
    ", K_Y);
13
14 // Result
15 // The Q-value of the reaction = 5.488 MeV
16 // The KE of the residual nucleus = 0.488 MeV
```

Scilab code Exa 19.2 Q value and threshold energy of the reaction

```
1 // Ex19_2 Page:368 (2014)
2 clc;clear;
3 m_x = 1.008665;     // Mass of a proton, u
4 m_y = 1.007825;     // Mass of a neutron, u
5 M_X = 18.998404;     // Atomic mass of F-19, u
6 M_Y = 19.003577;     // Atomic mass of O-19, u
7 Q = (M_X + m_x - m_y - M_Y)*931.5;     // Q-value of
    the reaction, MeV
8 K_x_min = (1 + m_x/M_X)*abs(Q);     // The threshold
    energy for the reaction, MeV
9 printf("\nThe Q-value of the reaction = %6.4f MeV",
    Q);
10 printf("\nThe threshold energy for the reaction = %4
    .2f MeV", K_x_min);
11
12 // Result
13 // The Q-value of the reaction = -4.0362 MeV
14 // The threshold energy for the reaction = 4.25 MeV
```

Scilab code Exa 19.3 Energy released in the fission reaction

```
1 // Ex19_3 Page:373 (2014)
2 clc;clear;
3 m_p = 1.007825;     // Mass of a proton, u
4 m_n = 1.008665;     // Mass of a neutron, u
5 m_U235 = 235.043924;     // Atomic mass of U-235, u
6 m_Ba141 = 140.91440;     // Atomic mass of Ba-141, u
7 m_Kr92 = 91.92630;     // Atomic mass of Kr-92, u
8 delta_m = (m_n + m_U235 - m_Ba141 - m_Kr92 - 3*m_n);
        // Mass difference, u
```

```
E = delta_m*931.5;     // Energy released in the
    fission reaction, MeV
printf("\nThe energy released in the fission
    reaction = %5.1f MeV", E);

// Result
// The energy released in the fission reaction =
    173.2 MeV
```

Scilab code Exa 19.4 Number of reactions per second in a reactor

```
// Ex19_4 Page:373 (2014)
clc;clear;
e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
E = 200;      // Energy released per fission, MeV
P = 300e+006;     // Power of the nuclear reactor, W
n = P/(E*1e+006*e);     // Number of fission
    reactions taking place per second
printf("\nThe number of fission reactions taking
    place per second = %4.2e", n);

// Result
// The number of fission reactions taking place per
    second = 9.38e+018
```

Scilab code Exa 19.5 Energy released epr gram of fuel

```
// Ex19_5 Page:378 (2014)
clc;clear;
m_D = 1.66e-027;     // Mass of the deutrium, kg
m_1H2 = 2.014102;     // Mass of deutrium, u
m_1H3 = 3.01609;     // Mass of the tritium, u
m_1H1 = 1.007825;     // Mass of protium, u
```

```
7 Q = (2*m_1H2 - m_1H3 - m_1H1)*931.5;     // Energy
    released per fusion, MeV
8 E = 0.001/(2*m_D)*Q/2;     // Energy released per
    gram of fuel, MeV
9 printf("\nThe energy released per gram of fuel = %4
    .2e MeV", E);
10
11 // Result
12 // The energy released per gram of fuel = 6.02e+023
    MeV
13 // The answer provided in the textbook is wrong
```

Scilab code Exa 19.6 Energy required for deuterium tritium fusion reaction

```
1 // Ex19_6 Page:379 (2014)
2 clc;clear;
3 e = 1.6e-019;     // Charge on an electron, C
4 k = 1.38e-023;     // Boltzmann constant, J/K
5 K = 8.99e+009;     // Coulomb's constant, N-Sq.m/Sq.C
6 r_d = 1.5e-015;     // Radius of deutrium nucleus, m
7 r_t = 1.7e-015;     // Radius of tritium nucleus, m
8 KE = K*e^2/(r_d + r_t)/2;     // Kinetic energy for
    one particle, MeV
9 T = 2*KE/(3*k);     // Temperature required for the
    deutrium-tritium fusion to occur, K
10 printf("\nThe temperature required for the deutrium-
    tritium fusion to occur = %1.0e K", T);
11
12 // Result
13 // The temperature required for the deutrium-tritium
    fusion to occur = 2e+009 K
```

Chapter 20

Nuclear Radiation Detectors and Particle Accelerators

Scilab code Exa 20.1 Better particle for studying details of the nucleus

```
1  // Ex20_1 Page:390 (2014)
2  clc;clear;
3  e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
4  m = 1.67e-027;     // Mass of a nucleon, kg
5  h = 6.626e-034;     // Planck's constant, Js
6  E = 30;     // Energy of alpha particle, MeV
7  r = 1.2e-015;     // Nuclear constant, m
8  lambda_p = h/sqrt(2*m*E*1e+006*e);     // de-Broglie
      wavelength of the proton, m
9  lambda_a = h/sqrt(2*4*m*E*1e+006*e);     // de-
      Broglie wavelength of the alpha particle, m
10 d = 2*r;     // size of the nucleon, m
11 printf("\nThe wavelength of the proton = %3.1e m",
      lambda_p);
12 printf("\nThe wavelength of the alpha particle = %3
      .1e m", lambda_a);
13 if ((lambda_p - d) < (lambda_a - d)) then
14     printf("\nThe proton is better for studying the
          nuclear details.")
```

```
15 else
16     printf("\nThe alpha particle is better for
       studying the nuclear details.")
17 end
18
19 // Result
20 // The wavelength of the proton = 5.2e-015 m
21 // The wavelength of the alpha particle = 2.6e-015 m
22 // The alpha particle is better for studying the
   nuclear details.
```

Scilab code Exa 20.2 Frequency of alternating voltage and KE of protons in a cyclo

```
1  // Ex20_2 Page:391 (2014)
2  clc;clear;
3  q = 1.6e-019;     // Charge on an electron, C
4  e = 1.6e-019;     // Energy equivalent of 1 eV, J/eV
5  m = 1.67e-027;     // Mass of a proton, kg
6  B = 2;     // Applied magnetic field, T
7  R = 0.25;     // Maximum radius of cyclotron, m
8  f = q*B/(2*%pi*m);     // Frequency needed for
   applying alternating voltage, Hz
9  K = (q*B*R)^2/(2*m);     // Kinetic energy of protons
    when they leave the cyclotron, J
10 printf("\nThe frequency needed for applying
   alternating voltage = %4.1f MHz", f/1e+006);
11 printf("\nThe kinetic energy of protons when they
   leave the cyclotron = %2d MeV", ceil(K/(e*1e+006)
   ));
12
13 // Result
14 // The frequency needed for applying alternating
   voltage = 30.5 MHz
15 // The kinetic energy of protons when they leave the
    cyclotron = 12 MeV
```

Chapter 21

Elementary Particles

Scilab code Exa 21.1 Q value of the elementary particle reaction

```
1 // Ex21_1 Page:399 (2014)
2 clc;clear;
3 m_pi = 140;     // Mass of a pion, MeV
4 m_p = 938.3;     // Mass of a proton, MeV
5 m_K = 498;     // Mass of a kaon, MeV
6 m_lambda = 1116;     // Mass of lambda hyperon, MeV
7 Q = m_pi + m_p - m_K - m_lambda;
8 printf("\nThe Q-value of the reaction = %3d MeV",
    floor(Q));
9
10 // Result
11 // The Q-value of the reaction = -536 MeV
```

Scilab code Exa 21.2 Relativistic mass of neutral pion

```
1 // Ex21_2 Page:399 (2014)
2 clc;clear;
3 m_pi_minus = 139.6;     // Mass of a pion, MeV
```

```
m_p = 938.3;     // Mass of a proton, MeV
m_n = 939.6;     // Mass of a neutron, MeV
K_n = 0.6;     // The kinetic energy of the neutron, MeV
E_pi0 = m_p + m_pi_minus - m_n - K_n;     // Mass of neutral pion, MeV
E_pi0_r = sqrt(E_pi0^2 - (m_n + K_n)^2 + m_n^2);     // Relativistic mass of neutral pion, MeV
printf("\nThe relativistic mass of neutral pion = %5.1f MeV", E_pi0_r);

// Result
// The relativistic mass of neutral pion = 133.5 MeV
```