# Scilab Textbook Companion for Modern Physics by K. S. Krane[1]

Created by
Balakavi Venkata Siva Sai Krishna
B.TECH
Electrical Engineering
VNIT,NAGPUR.
College Teacher
V S Kale
Cross-Checked by

August 10, 2013

[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:** K. S. Krane

**Publisher:** John Wiley & Sons

**Edition:** 2

**Year:** 1996

**ISBN:** 978-81-265-0826-6

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

# Introduction

**Scilab code Exa 1.1** Mass difference between a proton and a neutron

```
clear
clc
disp('Exa-1.1');
Mn=1.008665;Mp=1.007276                                  //Given mass
   of an electron and a proton in terms of u
Md= Mn-Mp;
                                                         //mass difference
printf('Mass difference in terms of U is %f ',Md);
Md=Md*931.50;                                            //
   converting u into Mev/c^2 by multiplying by 931.5
   MeV/c^2
printf('which equals %.3f Mev/c^2.',Md);
```

**Scilab code Exa 1.2** Total mass of proton and electron

```
clear
clc
disp('Exa-1.2');
Mp=1.007276 ; Me=5.4858*10^-4; //mass of proton and electron in terms of U
Mt=Mp+Me;                         //Total mass= sum of above masses
printf('The combined mass of an electron and a proton was found out to be %f U.',Mt);
```

**Scilab code Exa 1.3** Value of hc

```
clear
clc
disp('Exa-1.3');
h=6.621*10^-34 ; c=2.9979*10^8;                    //h is in J/s and c is in m/s
hc=h*c*((10^9)/(1.6022*10^-19));                   //1e=1.602*10^-19 J and 1 m=10^9 nm
printf('The value of hc is %f eV.nm\n',hc);
printf('Rounding off to 4 digits, we obtain %4.f eV.nm.',hc);
disp('Hence zero at the end is significant.');
```

# Chapter 2

# The Special Theroy of Relativity

**Scilab code Exa 2.1** Speed of A wrt B

```
clear
clc
disp('Exa-2.1');
v1=60; v2=40        //Velocities of cars wrt to
   observer in km/hr
vr=v1-v2;           //relative velocity
printf('The value of relative velocity is %4.f km/h.
   ',vr);
```

---

**Scilab code Exa 2.2** Velocity of plane wrt ground

```
clear
clc
disp('Exa-2.2');
```

```
Va_w=[320 0]; Vw_g=[0 65];            //Vp/q=[X Y]=> velocity of object p wrt q along X(east) and Y(north) directions.
Va_g=Va_w + Vw_g;                      //net velocity
k=norm(Va_g);                          //magnitude
s=atan(Va_g(1,2)/Va_g(1,1))*180/%pi;     //angle in rad*180/pi for conversion to degrees
printf('The magnitude of velocity Va/g(airplane wrt ground) is %.3f Km/h at %.3f degrees north of east.',k,s);
```

**Scilab code Exa 2.3** Time taken in each case

```
clear
clc
disp('Exa-2.3'); // The problem is entirely theoretical hence following the standard procedure we obtain
printf('The time required for round trip is 2*L/(c*(1-(u/c)^2)). \n');
printf('The time required to swim across and return is 2*L/(c*sqrt((1-(u/c)^2)))');
```

**Scilab code Exa 2.4** Minimum speed for suvival of muons

```
clear
clc
disp('Exa-2.3');
```

```
Lo=100*(10^3);c=3*(10^8); //Given values//all the
    quantities are converted to SI units
d=2.2*(10^-6);       //time between its birth and
    decay
t=Lo/c               //where Lo is the distance from
    top of atmosphere to the Earth. c is the velocity
     of light. t is the time taken
u=sqrt(1-((d/t)^2));    // using time dilaion fromula
     for finding u where u is the minimum velocity in
     terms of c;
printf('Hence the minimum speed required is %f c.',u
    );
```

**Scilab code Exa 2.5** Apparent thickness of Earth atmosphere

```
clear
clc
disp('Exa-2.5');
Lo=100*(10^3);   //Lo is converted to Km
u=0.999978;       ////u/c is taken as u since u is
    represented in terms of c.
L=Lo*(sqrt(1-u^2)); // from the length contraction
    formula
printf('Hence the apparent thickness of the Earth''s
     surface is %.2f metres.',L);
```

**Scilab code Exa 2.6** Solution for a b c d e

```
clear all
```

```
clc
disp('Exa-2.6(a)');
L=65; c=3*10^8;u=0.8*c;
t=L/u ;            //The value of time taken as measured by the observer
printf('The time for rocket to pass a point as measured by O is %.2e.\n',t);          //The value of time taken as measured by the observer
disp('Exa-2.6(b)');
Do=65;                               //given length
Lo= L/sqrt(1-(u/c)^2);                        //contracted length of rocket
printf('Actual length according to O is %.2f.\n',Lo);
disp('Exa-2.6(c)');
D=Do*(sqrt(1-(u/c)^2));        //contracted length of platform.
printf('Contracted length according to O'' is %.2e.\n',D);
disp('Exa-2.6(d)');
t1=Lo/u;                        //time needed to pass according to O'.
printf('Time taken according to O is %.2e.\n',t1);
disp('Exa-2.6(e)');
t2=(Lo-D)/u;                          //time intervals between the two instancs
printf('Time taken according to O'' is %.2e.\n',t2);
disp('The value of t1 and t2 did not match');
```

**Scilab code Exa 2.7** Speed of missile wrt earth

```
clear all
clc
```

```
disp('Exa-2.7');
v1=0.6; u=0.8; c=1; // all the values are measured
    in terms of c hence c=1
v= (v1+u)/(1+(v1*u/c^2));
printf('The speed of missile as measured by an
    observer on earth is %.2f c.',v);
```

**Scilab code Exa 2.8** Speed of galaxy wrt earth

```
clear all
clc
disp('Exa-2.8');
w1=600;w2=434;  // w1=recorded wavelength;w2=actual
    wavelength
                 // c/w1 = c/w2 *(sqrt(1-u/c)/(1+u/c))
k=w2/w1;
x=(1-k^2)/(1+k^2);          //solving for u/c
printf('The speed of galaxy wrt earth is %.2f c',x);
```

**Scilab code Exa 2.9** Velocity of rocket2 wrt rocket1

```
clear all
clc
disp('Exa-2.9');
v1x=0.6;v1y=0;v2x=0;v2y=.8;c=1;      // all the
    velocities are taken wrt c
v21x=(v2x-v1x)/(1-(v1x*v2x/c^2));    //using lorentz
    velocity transformation
v21y=(v2y*(sqrt(1-(v1x*c)^2)/c^2))/(1-v1y*v2y/c^2)
```

```
printf('The velocity of rocket 2 wrt rocket 1 along
    x and y directions is  %.2f c & %.2f c
    respectively',v21x,v21y);
```

**Scilab code Exa 2.10** Time interval between the events

```
clear all
clc
disp('Exa-2.10');
u=0.8*c;L=65;c=3*10^8;                //all values are
    in terms of c
t=u*L/(c^2*(sqrt(1-((u/c)^2))));   //from the
    equation 2.31
printf('The time interval between the events is %e
    sec which equals %.2f usec.',t,t*10^6);
```

**Scilab code Exa 2.11** momentum of proton

```
clear all
clc
disp('Exa-2.11');
m=1.67*10^-27;c= 3*10^8;v=0.86*c;       //all the
    given values and constants
p=m*v/(sqrt(1-((v/c)^2)));               // in terms of
     Kgm/sec
printf('The value of momentum was found out to be %
    .3e Kg-m/sec.\n',p);
c=938;v=0.86*c;mc2=938                   // all the
    energies in MeV where mc2= value of m*c^2
```

```
pc=(mc2*(v/c))/(sqrt(1-((v/c)^2)));     //expressing in terms of Mev
printf('The value of momentum was found out to be %.2f Mev.',pc);
```

**Scilab code Exa 2.12** Various energies of proton

```
clear all
clc
disp('Exa-2.12');
pc=1580; mc2=938;E0=938;     // all the energies in MeV mc2=m*c^2 and pc=p*c
E=sqrt(pc^2+mc2^2);
printf('The relativistic total energy is %.2f MeV.\n',E);          //value of Energy E
K=E-E0;                                    //value of possible kinetic energy
printf('The kinetic energy of the proton is %.1f MeV.',K);
```

**Scilab code Exa 2.13** Velocity and momentum of electron

```
clear all
clc
disp('Exa-2.13');
E=10.51; mc2=0.511; //all the values are in MeV
p=sqrt(E^2-mc2^2);    //momentum of the electron
printf('The momentum of electron is %.1f MeV/c\n',p);
```

```
v=sqrt(1-(mc2/E)^2);   //velocity in terms of c
printf('The velocity of electron is %.4f c',v);
```

**Scilab code Exa 2.14** Solution for a b

```
clear all
clc
disp('Exa-2.14');
k=50;mc2=0.511*10^-3;c=3*10^8; // all the values of
   energy are in GeV and c is in SI units
v=sqrt(1-(1/(1+(k/mc2))^2));    //speed of the
   electron in terms of c
k=c-(v*c);                          //difference in
   velocities
printf('Speed of the electron as a  fraction of c is
    %.12f*10^-12.\n',v*10^12); // v=(v*10^12)
   *10^-12; so as to obtain desired accuracy in the
   result
printf('The difference in velocities is %.1f cm/s.',
   k*10^2);
```

**Scilab code Exa 2.15** Rate of decrese of the mass of Sun

```
clear all
clc
disp('Exa-2.15');
r=1.5*10^11; I=1.4*10^3;      //radius and intensity
   of sun
```

```
s=4*%pi*r^2                    //surface area of the
    sun
Pr=s*I                         // Power radiated in J/
    sec
c=3*10^8;                      //velocity of light
m=Pr/c^2                       //rate od decrease of mass
printf('The rate of decrease in mass of the sun is %
    .1e kg/sec.',m);
```

**Scilab code Exa 2.16** Kinetic energy of pion in each case

```
clear all
clc
disp('Exa-2.16');
K=325; mkc2=498;  //kinetic energy and rest mass
    energy of kaons
mpic=140; //given value
Ek=K+mkc2;
pkc=sqrt(Ek^2-mkc2^2);
//consider the law of conservation of energy which
    yields Ek=sqrt(p1c^2+mpic^2)+sqrt(p2c^2+mpic^2)
// The above equations (4th degree,hence no direct
    methods)can be solved by assuming the value of
    p2c=0.
p1c=sqrt(Ek^2-(2*mpic*Ek));
//consider the law of conservation of momentum.
    which gives p1c+p2c=pkc implies
p2c=pkc-p1c;
k1=(sqrt(p1c^2+(mpic^2))-mpic); //corresponding
    kinetic energies
k2=(sqrt((p2c^2)+(mpic^2))-mpic);
printf('The corresponding kinetic energies of the
    pions are %.0f MeV and %.1f MeV.',k1,k2);
```

**Scilab code Exa 2.17** Threshold kinetic energy to produce antiprotons

```
clear all
clc
disp('Ex-2.17');
mpc2=938;c=3*10^8;   //mpc2=mp*c^2,mp=mass of proton
Et=4*mpc2;           //final total energy
E1=Et/2;E2=E1;        //applying conservation of
    momentum and energy
v2=c*sqrt(1-(mpc2/E1)^2);     //lorentz
    transformation
u=v2;v=(v2+u)/(1+(u*v2/c^2));
E=mpc2/(sqrt(1-(v/c)^2));
K=E-mpc2;
printf('The threshold kinetic energy is %.3f Gev',K
    /10^3);
```

# Chapter 3

# Review of Electromagnetic waves

**Scilab code Exa 3.1** Atomic Spacing of Nacl

```
clear
clc
disp('Exa-3.1');
w=0.250; theta=26.3;n=1 // n=1 for hydrogen atom and
    rest all are given values
d=n*w/(2*sind(theta));      // bragg's law
printf('Hence the atomic spacing is %.3f nm.',d);
```

**Scilab code Exa 3.2** Time taken to release an electron

```
clear
clc
disp('Exa-3.2');
I=120;r=0.1*10^-9;Eev=2.3    //I-intensity in W/m^2 r
    in m & E in electron volt
```

```
A=%pi*r^2;K=1.6*10^-19;       // A=area and K is conversion factor from ev to joules
t= Eev*K/(I*A);               //time interval
printf('The value of time interval was found out to be %.1f sec',t);
```

**Scilab code Exa 3.3** Solution for a and b

```
clear
clc
disp('Exa-3.3(a)');
w=650*10^-9;h=6.63*10^-34;c=3*10^8;   //given values and constant taken in comfortable units
E=h*c/w; printf('The Energy of the electron is %.3e J ',E);
E=E/(1.6*10^-19);printf('which is equivalent to %f eV\n',E);
printf('The momentum of electron is p=E/c i.e %.2f/c \n',E);
disp('Exa-3.3(b)');
E2=2.40;                              //given energy of photon.
w2=h*c*10^9/(E2*1.6*10^-19);          //converting the energy in to eV and nm
printf('The wavelength of the photon is %.2f nm',w2);
```

**Scilab code Exa 3.4** Solution for a b and c

```
clear
clc
disp('Exa-3.4(a)');
hc=1240; phi=4.52                    //both the values
    are in eV
w1=hc/phi;
printf('The cutoff wavelength of the tungsten metal
    is %.3fnm\n ',w1);
disp('Exa-3.4(b)');
w2=198;      //given value of wavelength
Kmax=(hc/w2)-phi;printf('The max value of kinetic
    energy is %.3f eV\n',Kmax);
disp('Exa-3.4(c)');
Vs=Kmax; printf('The numerical value of the max
    kinetic energy is same as stopping potential in
    volts.Hence %.2f V',Vs);
```

**Scilab code Exa 3.5** Solution for a b and c

```
clear
clc
disp('Exa-3.5(a)');
T1=293; Kw=2.898*10^-3;
w1=Kw/T1;
printf('The wavelength at which emits maximum
    radiation is %.2f um.\n',w1*10^6);
disp('Exa-3.5(b)');
w2=650*10^-9;
T2=Kw/w2;
printf('The temperature of the object must be raised
     to %.0f K.\n',T2);
disp('Exa-3.5(c)');
x=(T2/T1)^4; printf('Thus the thermal radiation at
```

```
higher temperature is %.2e times the room (lower) tempertaure.\n',x);
```

**Scilab code Exa 3.6** Solution for a b c and d

```
clear
clc
disp('Exa-3.6(a)');
w1=0.24;wc=0.00243;theta=60;    //given values w=wavelength(lambeda)
w2=w1+(wc*(1-cosd(theta)));
printf('The wavelength of x-rays after scattering is %.4f nm\n',w2);
disp('Exa-3.6(b)');
hc=1240;
E2=hc/w2;E1=hc/w1; printf('The energy of scattered x-rays is %.0f eV\n',E2);
disp('Exa-3.6(c)');
K= E1-E2; //The kinetic energy is the difference in the energy before and after the collision;
printf('The kinetic energy of the x-rays is %.3f eV\n',K);
disp('Exa-3.6(d)');
phi2=atand(E2*sind(theta)/(E1-E2*cosd(theta)))
printf('The direction of the scattered eletron is %.1f degrees',phi2);
```

# Chapter 4

# The Wavelike properties of particles

**Scilab code Exa 4.1** Solution for a b c d and e

```
clear
clc
disp("Ex: 4.1 ");
h=6.6*10^-34;                                    // h(
   planck's constant)= 6.6*10^-34
m1= 10^3;v1=100;;                                  // for
   automobile
w1= h/(m1*v1);                                  // ['w'-
   wavelength in metre'm'-mass in Kg 'v'-velocity in
    metres/sec.] of the particles
printf("Wavelength of the automobile is %1.2e m\n",
   w1 );
m2=10*(10^-3);v2= 500;                            // for
   bullet
w2=h/(m2*v2);
printf("Wavelength of the bullet is %1.2e m\n ",w2 )
   ;
m3=(10^-9)*(10^-3); v3=1*10^-2;
w3=h/(m3*v3);
```

```
printf("Wavelength of the smoke particle is %1.2e m\n",w3 );
m4=9.1*10^-31;k=1*1.6*10^-19;                    // k-kinetic energy of the electron & using 1ev = 1.6*10^-19 joule
p=sqrt(2*m4*k);                                  // p=momentum of electron ;from K=1/2*m*v^2
w4=h/p;
printf("Wavelength of the electron(1ev) is %1.2fnm\n",w4*10^9 );
hc=1240;pc=100                                   // In the extreme relativistc realm, K=E=pc; Given pc=100 MeV,hc=1240MeV
w5= hc/pc;
printf("Wavelength of the electron (100Mev) is %1.2f fm\n",w5);
```

**Scilab code Exa 4.2** Minimum uncertainity in wavelength

```
clear
clc
disp('Ex-4.2');
// w=wavelength; consider k=2*(pi/w);
// differentiate k w.r.t w and replace del(k)/del(w) = 1 for equation.4.3
// which gives del(w)= w^2 /(2*pi*del(x)), hence
w=20; delx=200; // delx=200cm and w=20cm
delw=(w^2)/(delx*2*%pi);
printf('Hence uncertainity in length is %1.2f cm',delw);
```

**Scilab code Exa 4.3** Validity of the claim

```
clear
clc
disp('Ex-4.3')
delt=1;                    //consider time interval of 1 sec
delw=1/delt;             // since delw*delt =1 from equation 4.4
delf=0.01              //calculated accuracy is 0.01Hz
delwc =2*%pi*delf // delwc-claimed accuracy from w =2*pi*f
printf('The minimum uncertainity calculated is 1rad/sec. The claimed accuracy is %.3f rad/sec\n',delwc);
if delw==delwc then disp('Valid claim');
end
if delw~=delwc then disp('Invalid claim');
end
```

**Scilab code Exa 4.4** Solution for a and b

```
clear
clc
disp('Ex-4.4(a)');
m=9.11*10^-31;v=3.6*10^6; //'m','v' - mass an velocity of the electron in SI units
h=1.05*10^-34; //planck's constant in SI
```

```
p=m*v; //momentum
delp=p*0.01;//due to 1% precision in p
delx = h/delp//uncertainity in position
printf('Uncertainity in position is %1.2f nm',delx
    *10^9);
disp('Ex-4.4((b)')
printf('Since the motion is strictly along X-
    direction , its velocity in Y direction is
    absolutely zero.\n So uncertainity in velocity
    along y is zero=> uncertainity in position along
    y is infinite. \nSo nothing can be said about its
     position/motion along Y')
```

**Scilab code Exa 4.5** Solution for a and b

```
clear
clc
disp('Ex-4.5(a)');
m=0.145;v=42.5; //'m','v' - mass an velocity of the
    electron in SI units
h=1.05*10^-34; //planck's constant in SI
p=m*v; //momentum
delp=p*0.01;//due to 1% precision in p
delx = h/delp//uncertainity in position
printf('Uncertainity in position is %1.2e',delx);
disp('Ex-4.5(b)');
printf('Motion along y is unpredictable as long as
    the veloity along y is exactly known(as zero).');
```

**Scilab code Exa 4.6** Uncertainity in x component

```
clc
clear
disp('Ex-4.6')
printf('The uncertainity in the poisition of
    electron after it passes through the slit is
    reduced to width of the slit\n delx=a\n');
printf('The uncertainity in momentum = h/a\n');
printf('Position of landing(angle t) = sin t = tan t
    = delz/dely =(h/a)/2*pi*a= w/2*pi*a  \nwhere w=
    wavelenghth\n');
printf('Rewriting  the above expression a*sint = w
    /(2*pi)\n which is similar to a*sint = w  (
    neglect 2*pi)as found out by first minimum in
    diffraction by a slit of width a');
disp('It proves a close connection between wave
    behaviour and uncertainity principle');
```

---

**Scilab code Exa 4.7** Range of kinetic energy of an electron

```
clear
clc
disp('Ex-4.7');
mc2=2.15*10^-4;           //mc2 is the mass of the
    electron, concidered in Mev for the simplicity in
    calculations
hc=197                    // The value of h*c in Mev.
    fm for simplicity
delx= 10                  // Given uncertainity in
    position=diameter of nucleus= 10 fm
delp= hc/delx ;           //Uncertainiy in momentum per
    unit 'c' i.e (Mev/c) delp= h/delx =(h*c)/(c*delx)
```

```
;hc=197 Mev.fm  1Mev=1.6*10^-13 Joules')
p=delp;                  // Equating delp to p  as a consequence of equation 4.10
K1=[[p]^2]+[mc2]^2   // The following 3 steps are the steps invlolved in calculating K.E= sqrt((p*c)^2 + (mc^2)^2)- m*c^2
K1=sqrt(K1)
K1= K1-(mc2);
printf('Kinetic energy was found out to be %d Mev',K1)
```

**Scilab code Exa 4.8** Solution for a b and c

```
clear
clc
disp('Ex-4.8')
h=6.58*10^-16; // plack's constant
delt1=26*10^-9;E=140*10^6 //given values of lifetime and rest energy of charged pi meson
delE=h/delt1; k=delE/E;  // k is the measure of uncertainity
printf('Uncertainity in energy of charged pi meson is %1.2e\n',k);
delt2=8.3*10^-17;E=135*10^6;  //given values of lifetime and rest energy of uncharged pi meson
delE=h/delt2; k=delE/E;
printf('Uncertainity in energy of uncharged pi meson is %1.2e\n',k);
delt3=4.4*10^-24;E=765*10^6;        //given values of lifetime and rest energy of rho meson
delE=h/delt3; k=delE/E;
printf('Uncertainity in energy of rho meson is %.1f\n',k);
```

**Scilab code Exa 4.9** minimum velocity of the billiard ball

```
clear
clc
disp('Ex-4.9')
h=1.05*10^-34;  //value of planck's constant in J.
   sec
delx= 1;        // uncertainity in positon=
   dimension of the ball
delp=h/delx;    // uncertainity in momentum
m=0.1;          //mass of the ball in kg
delv=delp/m;    // uncertainity in velocity
printf('The value of minimum velocity was found out
   to be %1.2e m/sec',delv);
```

**Scilab code Exa 4.10** Group velocity of a wave packet in terms of phase velocity

```
clc
clear
disp('Ex-4.10');
printf(' Group velocity is found out from Eq. 4.18.\
   n Since k=2*pi/w ; Vphase= w/k \n w/k = sqrt(g/k)
    /n w=sqrt(g*k)');
printf('\ndifferetiating on both sides\n');
printf('dw=1/2 * sqrt(g) * k^-1/2 * dk\n dw= 1/2 *
   sqrt(g/k)\n Hence Vgroup= Vphase/2');
```

# Chapter 5

# The Schrodinger Equation

**Scilab code Exa 5.1** Displacement and velocity of the object

```
clear
clc
disp('Exa-5.1');//The solution involves very complex
    symbolic equation solving and approximations.
   Hence only answers are displayed
printf('The displacement and velocity of the ball
   are found out in 2 steps\n step1-before reaching
   the surface of water and \n step2-Inside water
   till it rises back to surface\n');
printf('The values are as follows: v1(t)=-g*t and y1
   (t)=H-((g/2)*t^2))\n');
printf('In region 2: v2(t)=(-B/m*sqrt(2*H/g))+(B/m-g
   )*t; y2(t)= H+ HB/mg -B/m*sqrt(2*H/g)+ (B/m-g)');
```

**Scilab code Exa 5.2** Solution for a and b

```
clear
```

```
clc
disp('Exa-5.2(a)');
h=1.05*10^-34;m=9.11*10^-31;L=10^-10;      // all
   the values are taken in SI units
E1=h^2*%pi^2/(2*m*L^2); E2=4*E1;            //
   Energies are calculated
delE=(E2-E1)/(1.6*10^-19);                  //
   Difference in energy is converted to eV
printf('Energy to be supplied is %.0f eV.\n',delE);
disp('Exa-5.2(b)');
x1=0.09*10^-10;x2=0.11*10^-10                //
   limits of the given region
probGnd=(2/L)*integrate('(sin(%pi*x/L)^2)','x',x1,x2
   );
printf('The percentage probablility of finding an
   electron in the ground state is %.2f.\n',probGnd
   *100);
disp('Exa-5.2(c)');
x1=0,x2=0.25*10^-10;
probExc=(2/L)*integrate('(sin(2*%pi*x/L)^2)','x',x1,
   x2);
printf('The probablility of finding an electron in
   the excited state is %.2f.\n',probExc);
```

**Scilab code Exa 5.3** Proof for average value of x

```
clear
clc
disp('Ex-5.3');
x1=0;x2=L;
xavg=(2/L)*integrate('sin(%pi*x/L)^2','x',x1,x2);
printf('The average value of x is found out to be L
   /2 which apparently is independent of Qunatum
```

```
state.');
```

# Chapter 6

# The Rutherford Bohr model of an atom

**Scilab code Exa 6.1** Average deflection angle per collision

```
clear
clc
disp('Exa-6.1');
R=0.1;Z=79; x=1.44;           //x=e^2/4*pi*epsi0
zkR2=2*Z*x/R                  // from zkR2= (2*Z*e^2)*
    R^2/(4*pi*epsi0)*R^3
mv2=10*10^6;                  //MeV=>eV
theta=sqrt(3/4)*zkR2/mv2;     //deflection angle
theta=theta*(180/%pi);        //converting to
    degrees
printf('Hence the average deflection angle per
    collision is %.2f degrees.',theta);
```

**Scilab code Exa 6.2** Solution for a and b

```
clear
clc
disp('Exa-6.2(a)');
Na=6.023*10^23;p=19.3;M=197;
n=Na*p/M;           //The number of nuclei per atom
t=2*10^-6;Z=79;K=8*10^6;x=1.44; theta=90;  //x=e^2/4*pi*epsi0
b1=t*Z*x*cotd(theta/2)/(2*K)          //impact parameter b
f1=n*%pi*b1^2*t                       //scattering angle greater than 90
printf('The fraction of alpha particles scattered at angles greater than 90 degrees is %.2e\n',f1);
disp('Exa-6.2(b)');
theta=45
b2=t*Z*x*cotd(theta/2)/(2*K);
f2=n*%pi*b2^2*t;                     //scattering angle greater than 45
fb=f2-f1                           //scattering angle between 45 to 90
printf('The fraction of particles with scattering angle from 45 to 90 is %.3e\n',fb);
```

**Scilab code Exa 6.3** Distance of closest approach

```
clear
clc
disp('Exa-6.3');
Z=79;x=1.44;K=8*10^6;z=2;     //where x=e^2/4*pi*epsi0;z=2 for alpha particles
d=z*x*Z/K;                    //distance
printf('The distance of closest approasch is %.2e nm.',d*10^-9)
```

**Scilab code Exa 6.4** Three longest wavelengths of the Paschen series

```
clear
clc
disp('Exa-6.4');
sl=820.1;n0=3;  //given values
n=4;w=sl*(n^2/(n^2-n0^2)); printf('The 3 longest
   possible wavelengths are %.0f nm,',w);
n=5;w=sl*(n^2/(n^2-n0^2)); printf('%.0f nm,',w);
n=6;w=sl*(n^2/(n^2-n0^2)); printf('& %.0f nm ',w);
```

**Scilab code Exa 6.5** Various wavelegths in Balmer and Lymann series

```
clear
clc
disp('Exa-6.5');
sl=364.5;n=3;            //given variables and
   various constants are declared in the subsequent
   steps wherever necessary
w1=sl*(n^2/(n^2-4));     //longest wavelength of
   balmer
c=3*10^8;
f1=c/(w1*10^-9);         //corresponding freq.
n0=1;n=2;
w2=91.13*(n^2/(n^2-n0^2)); //first longest of lymann
f2=c/(w2*10^-9);          //correspoding freq
n0=1;n=3
```

```
w3=91.13*(n^2/(n^2-n0^2));       //second longest of lymann
f3=3*10^8/(w3*10^-9)              //corresponding freq.
printf('The freq. corresponding to the longest wavelength of balmer is %e & First longest wavelength  of Lymann is %e.\n',f1,f2);
printf('The sum of which s equal to %e\n',f1+f2);
printf('The freq. corresponding to 2nd longest wavelength was found out to be %e\n.Hence Ritz combination principle is satisfied.',f3);
```

**Scilab code Exa 6.6** wavelengths of transition

```
clear
clc
disp('Exa-6.6');
Rinfi=1.097*10^7;   //known value
n1=3;n2=2;          //first 2 given states
w=(n1^2*n2^2)/((n1^2-n2^2)*Rinfi);printf('Wavelength of trnasition from n1=3 to n2=2 is %.3f nm\n',w*10^9);
n1=4;n2=2;         //second 2 given states
w=(n1^2*n2^2)/((n1^2-n2^2)*Rinfi);printf('Wavelength of trnasition from n1=3 to n2=2 is %.3f nm',w*10^9);
```

**Scilab code Exa 6.7** Two longest wavelengths of triply ionized beryllium

```
clear
clc
disp('Exa-6.7');
n1=3;n2=2;Z=4;hc=1240;
delE=(-13.6)*(Z^2)*((1/(n1^2))-((1/n2^2)));
w=(hc)/delE; //for transition 1
printf('The wavelngth of radiation for transition
   (2->3) is %f nm\n',w);
n1=4;n2=2;   // n values for transition 2
delE=(-13.6)*(Z^2)*((1/n1^2)-(1/n2^2));
w=(hc)/delE;
printf('The wavelngth of radiation emitted for
   transition(2->4) is %f nm',w);
```

# Chapter 7

# The Hydrogen atom in wave mechanics

**Scilab code Exa 7.1** Proof

```
clear
clc
disp('Exa-7.1'); //The problem is entirely
   theoretcial.
printf('The solution obtained is r=4*ao i.e the most
    likely distance from origin for an electron in n
   =2,l=1 state.');
```

---

**Scilab code Exa 7.2** Probability of finding an electron closer to nucles than Bohrs orbit

```
clear
clc
disp('Exa-7.2');
```

```
// calculating radial probability P= (4/ao^3)*inegral(r^2 * e^(-2r/ao)) between the limits 0 and ao for r
Pr=integrate('((x^2)*%e^(-x))/2','x',0,2);//simplifying where as x=2*r/a0; hence the limits change between 0 to 2
printf('Hence the probability of finding the electron nearer to nucleus is %.3f',Pr);
```

**Scilab code Exa 7.3** Probability of finding an electron inisde Bohr Radius

```
clear
clc
disp('Exa-7.3');
//for l=0;
// employing the formula for probability distribution similarly as done in Exa-7.2
Pr1= integrate('(1/8)*((4*x^2)-(4*x^3)+(x^4))*%e^(-x)','x',0,1);            //x=r/ao; similrly limits between 0 and 1.
Pr2=integrate('(1/24)*(x^4)*(%e^-x)','x',0,1);                    //x=r/ao; similrly limits between 0 and 1.
printf('The probability for l=0 electron is %.3f and for l=1 electron is %.4f.',Pr1,Pr2);
```

**Scilab code Exa 7.4** Length of angular momentum vectors

```
clear
```

```
clc
disp('Exa-7.4');
l=1;          //given value of l
am1=sqrt(l*(l+1));       //angular momentum==sqrt(l(l
    +1)) h
l=2                      //given l
am2=sqrt(l*(l+1));
printf('The angular momenta are found out to be %.3f
     h and %.3f h respectively for l=1 and l=2.',am1,
    am2);
```

**Scilab code Exa 7.5** possible Z components of the vector L

```
clear
clc
disp('Exa-7.5'); //Thoretical question
disp('The possible values for m are [+2,-2] and
    hence any of the 5 components [-2h,2h] are
    possible for the L vector.');
printf('Length of the vector as found out previously
     is %.2f*h.',sqrt(6));//angular momentum==sqrt(l(
    l+1)) h
```

**Scilab code Exa 7.6** Seperation of beams as they leave the magnet

```
clear
clc
disp('Exa-7.6');
```

```
uz=9.27*10^-24; t=1.4*10^3; x=3.5*10^-2;     //various constants and given values
m=1.8*10^-25;v=750;                          // mass and velocity of the particle
d=(uz*t*(x^2))/(m*(v^2));                    //net separtion
printf('The distance of separation is %.2f mm',d*10^3);
```

**Scilab code Exa 7.7** Change in wavelength

```
clear
clc
disp('Exa-7.7');
n1=1;n2=2;hc=1240;     //hc=1240 eV.nm
E=(-13.6)*((1/n2^2)-(1/n1^2));    //Energy calcuation
w=hc/E;                        //wavelength
u=9.27*10^-24; B=2;            //constants
delE= u*B/(1.6*10^-19);        //change in energy
delw=((w^2/hc))*delE;               //change in wavelength
printf('The change in wavelength is %.5f nm.',delw);
```

# Chapter 8

# Many Electron Atoms

**Scilab code Exa 8.1** Energy of Ka X ray of sodium

```
clear
clc
disp('Exa-8.1');
hc=1240*10^-9;Rinfi=1.097*10^7;Z=11;   //for sodium
   atom;and other constants in MeV
delE=3*hc*Rinfi*(Z-1)^2/4              //change in
   energy
printf('The energy of the Ka x-ray of the sodium
   atom is %.3f KeV.',delE/10^3);
```

**Scilab code Exa 8.2** Solution for a and b

```
clear
clc
disp('Exa-8.2(a)');
EKa=21.990;EKb=25.145;EK=25.514
                                       //all the
```

```
    values are in KeV
ELo=EKb-EKa;printf('The enrgy of La of X-ray is %.3
    fKeV.\n',ELo);          //Energy of La X-ray
disp('Exa-8.2(b)');
EL=-EK+EKa;printf('Hence the binding energy of the L
     electon is %.3fKeV.',EL); // for electron L
    electron
```

**Scilab code Exa 8.3** Total orbital and spin quantum numbers of carbon

```
clear
clc
disp('Exa-8.3'); // theoretical
l=1; Lmax=l+l;Lmin=l-l;printf('Value of L ranges
    from %d to %d i.e %d %d %d\n.',Lmin,Lmax,Lmin,1,
    Lmax);
s=1/2; Smax=s+s;Smin=s-s;printf('Values of S are %d
    &%d',Smax,Smin);
```

**Scilab code Exa 8.4** Total orbital and spin quantum numbers of nitrogen

```
clear
clc
disp('Exa-8.4');
l=1; Lmax=l+l;Lmin=l-l;printf('Considering any two
    electrons,Value of L2e ranges from %d to %d i.e
    %d %d %d.\n',Lmin,Lmax,Lmin,1,Lmax);
printf('Adding the angular momentum of the third
    electron to L2emax gives the maximum  whole
```

```
   angular momentum as 2+1=3; and subtracting it
   from L2e=1 gives 0\n')
s=1/2; Smax=s+s;Smin=s-s;printf('Values of S2e are
   %d &%d.\n',Smax,Smin);
printf('Adding and subtracting the spin of third to
   S2e=1 and S2e=0 respectively gives the spins 3/2
   and 1/2 for the 3 electron system.');
```

**Scilab code Exa 8.5** Hunds rule to find ground state quantum numbers of nitrogen

```
clear
clc
disp('Exa-8.5');
disp('The nitrogen atom has a configuration of 1s2,2
   s2,2p3.');
disp('Let us maximize the net spin of all the 3
   electrons by assigning a spin of 1/2 to each of
   them. Hence S=3/2.');
disp('To maximize Ml,the consistent values of L for
   the 3 electrons left are 1 -1 and 0.Thus L=0 & S
   =3/2 are the ground state quantum numbers for
   nitrogen.');
```

**Scilab code Exa 8.6** Ground state L and S of oxygen

```
clear
clc
disp('Exa-8.6');
```

```
disp('The Oxygen atom has a configuration of 1s2,2s2
    ,2p4. 4 electrons in the outer most shell.');
disp('Let us maximize the net spin  by assigning a
    spin of 1/2 to 3 of them but the fourth should
    have spin of -1/3. Hence S=3/2-1/2=1.');
disp('The consistent values of L for the 3 electrons
     are 1 -1 and 0.To maximize Ml, assign a L 0f +1
    to the fourth electron.Thus L=1 & S=1 are the
    ground state quantum numbers for Oxygen.');
```

# Chapter 9

# Molecular Structure

**Scilab code Exa 9.1** Charge on the sphere

```
clear all
clc
disp('Ex-9.1');
E=-2.7;
K=9*(10^9)*((1.6*(10^-19))^2)/(0.106*10^-9);//taking all the values in meters. 1/(4*pi*e0)=9*10^9 F/m
q=((K-E*10^-9)/(4*K))*10^-9;                    //balancin by multiplying 10^-9 on numerator. to eV.vm terms
printf('Charge on the sphere required is %.2f times the charge of electron.',q);
```

**Scilab code Exa 9.2** Solution for a and b

```
clear
clc
```

```
disp('Exa-9.2(a)');
K=1.44; Req=0.236; // K=e^2/(4*pi*e0)=1.44 eV.nm
Uc=-K/(Req);          //coulomb energy
printf('The coulomb energy at an equilirium
    separation distance is %.2f eV\n',Uc);
E=-4.26; delE=1.53;     //various standars values of
    NaCl
Ur=E-Uc-delE;
printf('The pauli''s repulsion energy is %.2f eV\n',
    Ur);
disp('Exa-9.2(b)');
Req=0.1;          //pauli repulsion energy
Uc=-K/(Req);
E=4; delE=1.53;
Ur=E-Uc-delE;
printf('The pauli''s repulsion energy respectively
    is is %.2f eV\n',Ur);
```

---

**Scilab code Exa 9.3** vibrational frequency and photon energy of H2

```
clear
clc
disp('Exa-9.3');
delE=0.50; delR=0.017*10^-9;       //delE= E-Emin;
    delR=R-Rmin;
k=2*(delE)/(delR^2);c=3*10^8;      //force constant
m=(1.008)*(931.5*10^6)*0.5;        //mass of
    molecular hydrogen
v= sqrt(k*c^2/m)/(2*%pi);           //vibrational
    frequency
h=4.14*(10^-15);
E=h*v;
printf('The value of corresponding photon energy is
```

```
%.2f eV',E);
```

**Scilab code Exa 9.4** Energies and wavelengths of 3 lowest radiations emitted by molecular H2

```
clear
clc
disp('Exa-9.4');
hc=1240;          //in eV.nm
m=0.5*1.008*931.5*10^6;                    //mass of
   hydrogen atom
Req=0.074;                                 //equivalent
   radius
a=((hc)^2)/(4*(%pi^2)*m*(Req^2));   //reduced mass of
    hydrogen atom
for L=1:3,
        delE= L*a; printf('The value of energy is %f
            eV\n',delE);
        w=(hc)/delE;printf('The respective
           wavelength is is %f um\n',w*10^-3);
end
```

**Scilab code Exa 9.5** Rotational Inertia of molecule

```
clear
clc
disp('Exa-9.5');
delv=6.2*(10^11);      //change in frequency
h=1.05*(10^-34);         //value of h in J.sec
```

```
I= h/(2*%pi*delv);        //rotational inertia
printf('The value of rotational inertia is %.2e kg
    m2 ',I);
I=I/(1.684604e-045);
printf('which in terms of amu is %.3f u.nm2',I);
```

**Scilab code Exa 9.6** Solution for a and b

```
clear
clc
disp('Ex-9.6(a)');
delE=0.358;hc=4.14*10^-15;             //hc in eV.nm
    and delE=1.44eV(given values)
f=(delE)/hc;                           //frequency
printf('The frequency of the radiation is %.3e.\n',f
    );
m=0.98;                               //mass in terms
    of u
k=4*%pi^2*m*f^2;                      //value of k in
    eV/m^2
printf('The force constant is %.3e.\n',k);
disp('Ex-9.6(b)');
hc=1240; m=0.98*1.008*931.5*10^6; Req=0.127;       //
    various constants in terms of
s=((hc)^2)/(4*(%pi^2)*m*(Req^2));                 //
    expeted spacing
printf('The spacing was found out to be %f which is
    very close to the graphical value of 0.0026 eV.',
    s);
```

# Chapter 10

# Statistical Physics

**Scilab code Exa 10.1** Various Speeds obtained from maxwell speed distribution

```
clear
clc
disp('Exa-10.1'); //Theoretical Question
//**Install and use maxim tool for symbolic
   integration. remove the '//'(comment markings)
   below and run the program.
//Vm=integrate('(v^3)*(e^(-b*v^2))','x',0,%infi);
//rest of the results follow from above
printf('The average speed is found out to be (8*k*T/
   m)^1/2)\n');
printf('The RMS speed is (3*k*T/m)^1/2\n');
printf('The Most probable speed is found out to be
   (2*k*T/m)^1/2 \n where all the symbols used are
   conventional constants.');
```

**Scilab code Exa 10.2** Frequency distribution of emitted light

```
clear
clc
disp('Exa-10.2');        //The solution is purely
    theoretical and involves a lot of approximations.
printf('The value of shift in frequency was found
    out to be delf=7.14*fo*10^-7*sqrt(T) for a star
    composing of hydrogen atoms at a temperature T.\n
    ');
T=6000;                          //temperature for sun
delf=7.14*10^-7*sqrt(T);.....//change in frequency
printf('The value of frequency shift for sun(at 6000
    deg. temperature) comprsing of hydrogen atoms is
    %.1e times the frequency of the light.',delf);
```

**Scilab code Exa 10.3** Solution for a and b

```
clear
clc
disp('Exa-10.3(a)');
kT=0.0252;E=10.2                                  // at
    room temperature, kT=0.0252 standard value and
    given value of E
n2=2;n1=1; g2=2*(n2^2);g1=2*(n1^2);             //
    values for ground and excited states
t=(g2/g1)*%e^(-E/kT);                            //
    fraction of atoms
printf('The number of hydrogen atoms required is %e
    which weighs %e Kg\n',1/t,(1/t)*(1.67*10^-27));
disp('Ex-10.3(b)');
t=0.1/0.9;k=8.65*10^-5                            //
    fracion of atoms in case-2 is given
T=-E/(log(t/(g2/g1))*k);                          //
    temperature
```

```
printf('The value of temperature at which 1/10 atoms
    are in excited state is %.3f K',T);
```

**Scilab code Exa 10.4** Solution for a and b

```
clear
clc
disp('Exa-10.4(a)'); //theoretical
printf('The energy of interaction with magnetic
   field is given by uB and the degeneracy of the
   states are +-1/2 which are identical.\nThe ratio
   is therefore pE2/pE1 which gives e^(-2*u*B/k*T)')
   ;
disp('Ex-10.4(b)');
uB=5.79*10^-4;          //for a typical atom
t=1.1;k=8.65*10^-5;     //ratio and constant k
T=2*uB/(log(t)*k);      //temperature
printf('The value of temperature ar which the given
   ratio exists is %.2f K',T);
```

**Scilab code Exa 10.5** Fermi Energy Ef for sodium

```
clear
clc
disp('Exa-10.5');
p=0.971; A=6.023*10^23; m=23.0;     // various given
   values and constants
c= (p*A/m)*10^6;                    // atoms per unit
    volume
```

```
hc=1240; mc2=0.511*10^6;             // hc=1240 eV.nm
E= ((hc^2)/(2*mc2))*(((3/(8*%pi))*c)^(2/3)); //value of fermi energy
printf('The fermi energy for sodium is %f eV',E*10^-18);//multiply by 10^-18 to convert metres^2 term to nm^2
```

# Chapter 11

# Properties of Ionic Crystals

**Scilab code Exa 11.1** Solution for a and b

```
clear
clc
disp('Exa-11.1(a)');
c=769*10^3; Na=6.023*10^23; JeV=1.6*10^-19; //various constants and given values
Be=c/(Na*JeV);          //Binding energy of an ion pair in the lattice
printf('The experimental value was found out to be %.4f eV.\n',Be);
disp('Exa-11.1(b)');
n=9;a=1.7476; R=0.281; k= 1.44;      //Given values and consstants
Bc=k*a*(1-(1/n))/R;                  //ionic binding energy eperimentally
printf('The calculated value of the binding energy is %.4f eV.',Bc);
```

**Scilab code Exa 11.2** Energy per neytral atom to take apart a crystal of Nacl

```
clear
clc
disp('Exa-11.2');
a=3.61;// amount of energy required to remove an
    electron from Cl- ion
b=-5.14 //amount of energy returned when an electron
     is added to Na+ ion\
c=7.98 //binding energy of NaCl atom
E=a+b+c  //suom of all the energies
printf('The net energy to be supplied is %.3f eV',E)
    ;
```

---

**Scilab code Exa 11.3** Solution for a and b

```
clear
clc
disp('Ex-11.3(a)');
Na=6.023*10^23; p=8.96*10^3; M=63.5*10^-3;    //Na=
    avagadro's number,p=density,M=molar mass
n= p*Na/M;                                     //
    density of charge carriers
printf('The density of charge carriers in copper is
    %e atoms/m3\n',n);
s=5.88*10^7;m=9.11*10^-31;e=1.6*10^-19;      //charge
    & mass of an electron,resistance per unit length
t= s*m/(n*e^2);                              //average
     time between collisions
printf('The average time between collisions of
    conducting electrons is %e sec.\n',t);
disp('Ex-11.3(b)');
```

```
Ef=7.03*1.6*10^-19;             //converting given
    enrgy to J
Vf=sqrt(2*Ef/m);                 //fermi velocity
l=Vf*t;                         //mean free path
printf('The average mean free path is %e m = %.1f nm
    ',l,l*10^9);
```

# Chapter 12

# Nuclear Structure and Radioactivity

**Scilab code Exa 12.1** Solution for a and b

```
clear
clc
disp('Exa-12.1(a)');
Z=2;A=4;N=A-Z;          // Given values
printf('The following method of representing atoms
    is followed throughout the chapter\n\t\t x,ySz\n
    where x=atomic number y=mass number z= Neutron
    Number S=symbol of the atom\n\n')
printf('The helium can be reperesented as %d,%dHe%d\
    n',Z,A,N);
disp('Exa-12.1(b)');
Z=50;N=66;A=Z+N;        //// Given values and
    standard formulae
printf('The helium can be reperesented as %d,%dSn%d\
    n',Z,A,N);
disp('Exa-12.1(c)');
A=235;N=143;Z=A-N;
printf('The helium can be reperesented as %d,%dU%d',
    Z,A,N);
```

**Scilab code Exa 12.2** Approximate nuclear radii

```
clear
clc
disp('Exa-12.2');
r0=1.2;          //standard value.
A=12;
r= r0*A^(1/3);printf('The value of mean radius for C is: %.2f fm\n',r);
A=70;          //given value
r= r0*A^(1/3);printf('The value of mean radius for C is: %.2f fm\n',r);
A=209;
r= r0*A^(1/3);printf('The value of mean radius for C is: %.2f fm',r);
```

**Scilab code Exa 12.3** Density of typical nucleus and resultant mass

```
clear
clc
disp('Exa-12.3');
m=1.67*10^-27; r0=1.2*10^-15; v=4*%pi*(r0^3)/3 //standard values of mass radius and volume
p=m/v;                                          //denisty
printf('Density of typical nucleus is %.0e kg/m3 \n',p);
```

```
r0=0.01;v=4*%pi*(r0^3)/3;p=2*10^17;                    ////hypothetical values
m1=p*v;
printf('The mass of the hypothetical nucleus would be %.0e Kg',m1);
```

**Scilab code Exa 12.4** Total Binding Energy

```
clear
clc
disp('Exa-12.4');
N=30;Z=26;A=56;Mn=1.008665;Mp=1.007825;m=55.934939;c2=931.5;  //given values and constants for case-1
B=((N*Mn)+(Z*Mp)-(m))*c2;                                //binding energy(per nucleon)
printf('Binding nergy per nucleon for 26,56Fe30 is %.3f MeV\n',B/A);
N=146;Z=92;A=238;Mn=1.008665;Mp=1.007825;m=238.050785;c2=931.5;      //given values and constants for case-2
B=((N*Mn)+(Z*Mp)-(m))*c2;                                //binding energy(per nucleon)
printf('Binding nergy per nucleon for 26,56Fe30 is %.3f MeV',B/A);
```

**Scilab code Exa 12.5** Solution for a b c and d

```
clear
clc
disp('Exa -12.5(a)');
t12=2.7*24*3600;                    //converting days into seconds
w=0.693/t12;                        //lambeda
printf('The decay constant is %e\n /sec',w);
disp('Exa -12.5(b)');
printf('The decay constant is equal to probability of decay in one second hence %e \n',w);
disp('Exa -12.5(c)');
m=10^-6;Na=6.023*10^23; M=198;      //given values and constants
N=m*Na/M;                           //number of atoms in the sample
Ao=w*N;                  //activity
printf('The activity was found out to be %e Ci',Ao);
disp('Exa -12.5(d)');
t=7*24*3600;              //given time
A=Ao*%e^-(w*t);            //activity
printf('The activity after one week was found out to be %.2e decays/sec',A);
```

**Scilab code Exa 12.6** Atoms at the time of solidification

```
clear
clc
disp('Exa-12.6');
t1=4.55*10^9;t2=7.04*10^8;          //given values of time at 2 different instants
age=t1/t2;
r=2^age;
printf('The original rock hence contained %.1f*Na
```

```
atoms of 235U where Na is the Avagadro''s Number
=6.023*10^23',r);
```

**Scilab code Exa 12.7** Kinetic energy of alpha particle emitted in alpha decay

```
clear
clc
disp('Exa-12.7');
m236Ra=226.025403;
m222Rn=222.017571;
m4He=4.002603;c2=931.5; //mass of various elements
   and c2=c^2
Q=(m236Ra-m222Rn-m4He)*c2;//Q of the reaction
A=226
K=((A-4)/A)*Q;                          //kinetic
   energy
printf('The kinetic energy of the alpha particle is
   %.3f Mev',K);
```

**Scilab code Exa 12.8** Q value of 14C emission

```
clear
clc
disp('Exa-12.8');
m226Ra=226.025403; //mass of various elements
m212Pb=211.991871;
m14c=14.003242;
c2=931.5;       //value of c^2
```

```
Q=(m226Ra-m212Pb-m14c)*c2;       //Q of the reaction
printf('The value of Q for 14c emission is %.3f MeV\
   n',Q);
printf('The probability of 14c emission is 10^-9
   times that of an alpha particle since the energy
   barrier for 14c emission is\n nearly 3 times
   higher and thicker.')
```

**Scilab code Exa 12.9** Maximum Kinetic energy of emitted electron

```
clear
clc
disp('Ex-12.9');
m23Ne=22.994465; //mass of various elements
m23Na=22.989768;
c2=931.5;            //value of c^2
Q=(m23Ne-m23Na)*c2;      //Q of the reaction
printf('Hence the maximum kinetic energy of the
   emitted electrons is %.3f MeV',Q);
```

**Scilab code Exa 12.10** Q value of various decays

```
clear
clc
disp('Exa-12.10(a)');
m40K=39.963999;           //mass of various particles
m40Ca=39.962591;
c2=931.5;                     //value of c^2 in MeV
Qb1=(m40K-m40Ca)*c2;         //Q value of the reaction
```

```
printf('The Q value for -VE beta emission is %.3f Mev \n',Qb1);
disp('Exa-12.10(b)');
m40K=39.963999;            //mass of various particles
m40Ar=39.962384;
me=0.000549;
Qb2=(m40K-m40Ar-2*me)*c2;          //Q value of the reaction
printf('The Q value for +VE beta emission is %.3f Mev \n',Qb2);
disp('Exa-12.10(c)');
m40K=39.963999;
m40Ar=39.962384;
Qec=(m40K-m40Ar)*c2;
printf('The Q value for +VE beta emission is %.3f Mev \n',Qec);
```

**Scilab code Exa 12.11** Maximum kinetic energy of emitted beta particle

```
clear
clc
disp('Exa-12.11');
Mg=12.000000; //mass of the carbon atom in amu
c2=931.5;
Eg=4.43;      //given energy of gamma ray
Mex=Mg+(Eg/c2);      //mass in excited state
Me=0.000549;         //mass of an electron
Q=(12.018613-Mex-2*Me)*c2;      //Q of the particle
printf('The maximum value of kinetic energy is %.2f MeV',Q);
```

**Scilab code Exa 12.12** Rate of energy production per gram of uranium

```
clear
clc
disp('Exa-12.12');
m238U=238.050786; //mass of various quantities
m206Pb=205.974455;
m4He=4.002603;
c2=931.5;      //constants
Na=6.023*10^23;    //avagadro's number
Q=(m238U-m206Pb-8*m4He)*c2;
t12=(4.5)*10^9*(3.16*10^7);     //half life years to seconds conversion
w=0.693/t12;                           // lambeda
NoD=(Na/238)*w;       //number of decays
E=NoD*Q*(1.6*10^-19)*10^6;          //rate of liberation of energy,converting MeV to eV
printf('Rate of energy liberation is %.1e W',E);
```

**Scilab code Exa 12.13** Ages of the given rocks

```
clear
clc
disp('Exa-12.13');
R=0.5;t12=4.5*10^9;                  //value of radius and half-life
t1=(t12/0.693)*log(1+(1/R));         //age of rock 1
R=1.0;
```

```
t2=(t12/0.693)*log(1+(1/R));        //age of rock-2
R=2.0
t3=(t12/0.693)*log(1+(1/R));        //age of rock 3
printf('The ages of rock samples are %.1e, %.1e, %.1
   e years respectively',t1,t2,t3);
```

**Scilab code Exa 12.14** Solution for a and b

```
clear
clc
disp('Ex-12.14(a)');
P=2*10^14; V=2*10^-14; R=8.314; T=295;Na
   =6.023*10^23;    //varoius constants and given
   values
n=P*V/(R*T);         //ideal gas law
N=Na*n;f=10^-12          //avagadaro's number and
   fracction of carbon molecules
t12=5730*3.16*(10^7);    //half life
A=(0.693/t12)*N*f;     //activity
D1w=A*7*24*60*60;         //decays per second
 printf('The no of decays pers second is %4.0f \n',
    D1w);
 disp('Ex-12.14(b)');
c1=1420;                //concentration at instant 1
c2=D1w;                 //concentration at instant 2
t12y=5730;             //half life
t=t12y*log(c2/c1)/0.693;             //age of the
   sample
printf('Age of the sample is %.2f years',t);
```

# Chapter 13

# Nuclear Reaction and Applications

**Scilab code Exa 13.1** Rate of production of neutron

```
clear
clc
disp('Ex-13.1')
v=1*1*10^-6*10^2; p=7.9; m=p*v;Na=6.023*10^23      //
   given values and various constants in suitable
   units
M=56;N=m*Na/M;                                   //
   number of atoms
i=3*10^-6;
q=1.6*10^-19;
Io=i/q;                  //intensity
s=0.6*10^-24;S=1;             //given values in
   suitable units
R=N*s*Io/S;                    //rate of neutrons
printf('The rate of neutrons emitted from the target
    is %.2e particles per second',R);
```

**Scilab code Exa 13.2** Resultant activity of 198Au

```
clear
clc
disp('Exa-13.2')
A=197; m=30*10^-3;phi=3*10^12;     //given values and various constants taken in suitable units
Ar=99*10^-24; Na=6.023*10^23
R=(phi*Na*Ar*m/A);          //rate or production of gold
t=2.7*24*60          // time of decay
Act=R*(0.693/t);      //activity /sec
ActCi=Act/(2.7*10^-4);      // in terms of curie(Ci)
printf('The activity is found out to be %.2e/sec i.e %.2e Ci',Act,ActCi);
```

---

**Scilab code Exa 13.3** Solution for a and b

```
clear
clc
disp('Exa-13.3(a)')
v=1.5*1.5*2.5*(10^-6)*10^2; //volume in cm3
p=8.9;          //density in g/cm3
m=p*v;Na=6.023*10^23 //mass and Avagadro's number
M=58.9;          //Given values
N=m*Na/M;
i=12*10^-6;       //thickness of beam
q=1.6*10^-19;
```

```
Io=i/(2*q);        //intensity
s=0.64*10^-24;     //Given values
S=1.5*1.5;
R=N*s*Io/S;         //rate of production of 61Cu
printf('The rate of neutrons emitted from the target
    is %.2e particles/second\n',R);
disp('Exa-13.3(b)')
act=R*(1-(%e^((0.693)*(-2/3.41))));           //
   activity
printf('The activity after 2.0h is %e/sec',act);
```

**Scilab code Exa 13.4** Solution for a and b

```
clear
clc
disp('Exa-13.4(a)');
m2H=2.014102;     //mass of various particles
mn=1.008665;m63Cu=62.929599;
m64Zn=63.929145;c2=931.5;      //c^2=931.5 MeV
Q=(m2H+m63Cu-mn-m64Zn)*c2;        //Q of the reaction
printf('The value of Q is %f MeV\n',Q);
disp('Exa-13.4(b)');
Kx=12.00;Ky=16.85;
Ky=Q+Kx-Ky                    //kinetic energy of 64Zn
printf('The value of Ky was found out to be %.2f MeV
   ',Ky);
```

**Scilab code Exa 13.5** Solution for a and b

```
clear
clc
disp('Exa-13.5(a)');
mp=1.007825;m3H=3.016049;     //mass of the particle
m2H=2.014102;c2=931.5;       //constant
Q=(mp+m3H-(2*m2H))*c2;      //Q of thereaction
printf('The value of q was found out to be %f MeV\n',Q);
disp('Exa-13.5(b)');
Kth1= -Q*(1+(mp/m3H));         //threshold energy of kinetic energy
printf('The threshold kinetic energy in case-1 is %f MeV\n',Kth1);
Kth2=-Q*(1+(m3H/mp));         //threshold kinetic energy in case2
printf('The threshold kinetic energy in case-2 is %f MeV',Kth2);
```

# Chapter 14

# The Four Basic Forces

**Scilab code Exa 14.1** Solution for a and b

```
clear
clc
disp('Ex-14.1'); //theoretical question
printf('14.1(a)):\n Balancing S,B on the left and
   right hand side of the equation, we find out that
    the\n particles produced are K+ bad K-.\n\n');
printf('14.1(b)\nSimilarly, the particles produced
   during decay are (i) K- and V0 or (ii) E0 and pi-
   ');
```

**Scilab code Exa 14.2** Energy of the proton and pi meson

```
clear
clc
disp('Exa-14.2');
mvo=1116;mp=938;mpi=140;     //mass of various
   particles
```

```
Q=(mvo-mp-mpi);                //Q value of energy
Pp=100;Ppi=100;                //momentum of various
    particles
Kp=5;Kpi=38-Kp;                //kinetic energy of
    particles
printf('The kinetic energy of the particles Kp and
    Kpi are %d MeV and %d MeV respectively',Kp,Kpi);
```

**Scilab code Exa 14.3** Maximum kinetic energy of the electron emitted in the decay

```
clear
clc
disp('Exa-14.3');
Q=105.2                        // The Q value for the
    given decay
Muc2=105.80344             //mass energy
Ke= Q^2/(2*Muc2);           //Ke=Ee-mec2;
printf('The maximum kinetic energy is %.2f MeV',Ke);
```

**Scilab code Exa 14.4** maximum energy of the positron nad pi mesons

```
clear
clc
disp('Ex-14.4');
mkc2=494; mpic2=135;mec2=0.5;// mass of various
    particles
Q1=mkc2-mpic2-mec2;         //Q of reaction
 // the neutrino has negligible energy
```

```
deff('y=f(x)','sqrt(x^2+135^2)+x-494');// assigning
   the Q to sum of energies and simplifying
//k=fsolve(x);
printf('The value of maximum kinetic enrgy for pi-
   meson  and positron are %d MeV & %d MeV',266,229)
   ;
```

**Scilab code Exa 14.5** Q values for reaction

```
clear
clc
disp('Exa-14.5');
mpi_=140;mp=938;mKo=498;mLo=1116; //mass of various
   particles
Q1= mpi_+mp-mKo-mLo;          //Q value of reaction
   1
mK_=494;mpio=135;
Q2=mK_+mp-mLo-mpio;          //Q value of reaction 2
printf('The Q values of reactions 1 and 2 are %d MeV
    and %d MeV',Q1,Q2);
```

**Scilab code Exa 14.6** Threshold Kinetic energy to produce pi mesons

```
clear
clc
disp('Ex-14.6');
mpic2=135;        //mass ennergy of pi particle
Q=-mpic2;
mp=938;mpi=135;
```

```
Kth=(-Q)*((4*mp)+mpi)/(2*(mp));   //threshold energy
printf('The threshold kinetic energy is %.2f MeV',
   Kth);
```

**Scilab code Exa 14.7** Threshold Energy of the given reaction

```
clear
clc
disp('Ex-14.7');
mpc2=938;    //rest energy of proton
Q=mpc2+mpc2-(4*mpc2);         //Q value of reaction
Kth=(-Q)*(6*mpc2/(2*mpc2));         // thershold
   kinetic energy
printf('The threshold kinetic energy is %.2f MeV',
   Kth);
```

**Scilab code Exa 14.8** Solution for a and b

```
clc;
clear all;
disp('Ex-14.8');             //theoretical
printf('The reaction can be rewritten as follows U1+
   U--->S+S1. which implies that U and U1
   annihiliate creating S and S1\n');
disp('The pi+ has the quark composition Ud1.Since no
    quarks are present in the final state. One
   possible way to get rid of the quarks is to
   change U into d');
```

```
printf('U--->d+W(+). Hence the remaining processes
    are d+d(+)--->energy and \n W(+)--->u(+) and vu.'
    );
```

# Chapter 15

# Astrophysics and Genereal Relativity

**Scilab code Exa 15.1** Change in wavelength in solar spectrum due to gravitaional shift

```
clear
clc
disp('Ex-15.1');
w=121.5;    //lambeda
G=6.67*10^-11; //Various given values and constants
M= 1.99*10^30;
R= 6.96*10^8;
c=3*10^8;
k= G*M/(R*c^2);     //(delLambeda)/(lambeda)
delw=k*w;             //del(lambeda)
printf('The change in wavelength due to
   gravitational shift is %.3f pm\n',delw*10^3);
k=5.5*10^-5;//due to thermal Doppler broadening
   effect
delw=k*w;
printf('The change in wavelength due to thermal
   Doppler broadening effect is %.1f pm',delw*10^3);
```

**Scilab code Exa 15.2** Maximum energy of neutrino in the first reaction of proton proton cycle

```
clear
clc
disp('Ex-15.2');
mp=938.280;    //mass of various particles
me=0.511;
m2h=1875.628;
mic2=2*mp;              //mass energy on L.H.S
mfc2=m2h+me;   //mass energy on R.H.S
Q=mic2-mfc2;    //Q value of reation
pc=Q;
mc2=1875.628;
K=(pc^2)/(2*mc2);      //kinetic threshold energy
Emax=Q-K;          //maximum energy
printf('The maximum neutrino energy is %.3f MeV',
   Emax);
```

# Chapter 16

# The Cosmic Microwave Background Radiation

**Scilab code Exa 16.1** Resultin temperature of interstellar space

```
clear
clc
disp('Ex-16.1');
N2=0.25;N1=0.75;                                    //
   various given values
L2=1;L1=0;
E1_E2=-4.7*(10^-4);                                 //
   Energy difference
a=(N2/N1); b=(((2*L2)+1)/((2*L1)+1));c=E1_E2; //
   various terms involved in the formula of ratio of
    population
kT=(c/log(a/b));                                   //
   value of k*T
k=0.0000856;                                      //
   constant
T=kT/k;                    //temperature of
   interstellar space
printf('The temperature of interstellar space was
   found out to be %.1f K',T);
```

**Scilab code Exa 16.2** Solution for a and b

```
clear
clc
disp('Ex-16.2');
mc2=940*10^6; k=8.6*10^-5; //various constants and given values in suitable units
T= mc2/k;        //temperature of the photons
printf('The temperature of the photons must be %.1e K\n',T);
t=((1.5*10^10)/T)^2;        //age of universe when the photons have the above temperature
printf('The age of the universe for the temperature of the photon to be as obtained above is %.0e seconds',t);
```

**Scilab code Exa 16.3** Relative number of neutrons and protons among the nucleus

```
clear
clc
disp('Ex-16.3');
k=8.62*10^-5; //various values and constants
T= 1.5*10^10;
delE=1.3*10^6;
a= delE/(k*T);   //exponent in boltzmann factor
b=%e^-a;         //ratio of neutron to protons
```

```
r=(1/(1+b))*100;      //relative number of protons
printf('The percentage of protons is %.0f and
   neutrons is %.0f.',r,100-r);
```