

Scilab Textbook Companion for
Thermodynamics: A Core Course
by R. C. Srivastava, S. K. Saha And A. K.
Jain¹

Created by
Kedar Hanmant Kottawar
B.Tech
Chemical Engineering
VNIT , Nagpur (MH)
College Teacher
Shyam M Kodape
Cross-Checked by
Lavitha Pereira

August 10, 2013

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

Book Description

Title: Thermodynamics: A Core Course

Author: R. C. Srivastava, S. K. Saha And A. K. Jain

Publisher: PHI Learning Pvt. Ltd.

Edition: 2

Year: 2004

ISBN: 81-203-2498-6

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	4
2 Important Terminologies in Thermodynamics	7
3 The First Law of Thermodynamics	9
4 Defining Thermodynamic State The State Postulate	25
5 The Second Law of Thermodynamics	28
6 The Question of Ideality	55
7 Statistical Thermodynamics	66

List of Scilab Codes

Exa 2.1	numerical	7
Exa 2.2	numerical	8
Exa 3.1	numerical	9
Exa 3.2	numerical	10
Exa 3.3	numerical	10
Exa 3.4	numerical	11
Exa 3.5	numerical	12
Exa 3.6	numerical	12
Exa 3.7	numerical	13
Exa 3.8	numerical	14
Exa 3.9	numerical	15
Exa 3.10	numerical	15
Exa 3.11	numerical	16
Exa 3.12	numerical	16
Exa 3.13	numerical	17
Exa 3.14	numerical	18
Exa 3.15	numerical	19
Exa 3.16	numerical	19
Exa 3.17	numerical	20
Exa 3.18	numerical	21
Exa 3.19	numerical	22
Exa 3.20	numerical	23
Exa 3.21	numerical	23
Exa 4.1	numerical	25
Exa 4.2	numerical	26
Exa 4.3	numerical	26
Exa 5.1	numerical	28
Exa 5.2	numerical	29

Exa 5.3	numerical	29
Exa 5.5	numerical	30
Exa 5.6	numerical	30
Exa 5.7	numerical	31
Exa 5.8	numerical	32
Exa 5.9	numerical	32
Exa 5.10	numerical	33
Exa 5.11	numerical	34
Exa 5.12	numerical	34
Exa 5.13	numerical	35
Exa 5.14	numerical	36
Exa 5.15	numerical	36
Exa 5.16	numerical	37
Exa 5.17	numerical	38
Exa 5.19	numerical	39
Exa 5.20	numerical	39
Exa 5.21	numerical	40
Exa 5.22	numerical	41
Exa 5.23	numerical	42
Exa 5.26	numerical	42
Exa 5.27	numerical	43
Exa 5.28	numerical	44
Exa 5.29	numerical	45
Exa 5.30	numerical	45
Exa 5.31	numerical	46
Exa 5.32	numerical	46
Exa 5.33	numerical	47
Exa 5.34	numerical	47
Exa 5.35	numerical	48
Exa 5.36	numerical	49
Exa 5.38	numerical	49
Exa 5.39	numerical	50
Exa 5.40	numerical	50
Exa 5.41	numerical	51
Exa 5.42	numerical	52
Exa 5.43	numerical	53
Exa 5.44	numerical	54
Exa 6.2	numerical	55

Exa 6.5	numerical	56
Exa 6.10	numerical	56
Exa 6.11	numerical	57
Exa 6.12	numerical	58
Exa 6.14	numerical	58
Exa 6.15	numerical	59
Exa 6.16	numerical	60
Exa 6.17	numerical	60
Exa 6.18	numerical	61
Exa 6.19	numerical	62
Exa 6.20	numerical	62
Exa 6.21	numerical	63
Exa 6.22	numerical	64
Exa 6.23	numerical	65
Exa 7.1	numerical	66
Exa 7.2	numerical	67
Exa 7.3	numerical	68
Exa 7.4	numerical	68
Exa 7.5	numerical	69
Exa 7.6	numerical	70
Exa 7.9	numerical	71
Exa 7.12	numerical	72
Exa 7.14	numerical	72
Exa 7.15	numerical	73
Exa 7.16	numerical	74
Exa 7.17	numerical	75

Chapter 2

Important Termonologies in Thermodynamics

Scilab code Exa 2.1 numerical

```
1 //example 2.1
2
3 clear all;
4 clc;
5
6 //Given:
7 m=25; //weight of water vapour [grams]
8 w=18; //molecular weight of water vapour [grams/mol]
9 T=9.69; //increase in temperature [K]
10 Qp=0.45; //heat supplied at constant pressure [KJ]
11
12
13 //To find the molar constant pressure specific heat
14 n=m/w; //no. of moles of water vapour
15 Cp=Qp/(n*T); //specific heat capacity at constant
    pressure [KJ]
16 printf("The specific heat capacity at constant
    pressure = %f KJ/K/mol", Cp)
```

Scilab code Exa 2.2 numerical

```
1 //example 2.2
2
3 clear all;
4 clc;
5
6 //Given:
7 m=16; //weight of oxygen [grams]
8 w=32; //molecular weight of oxygen [grams/mol]
9 T=300; //Temperature during compression [K]
10 P1=1; //initial pressure of process [atm]
11 P2=100; //final pressure of process [atm]
12 R=8.314; //Universal gas constant [J/K/mol]
13
14 //To find the minimum work of compression
15 n=m/w; //no. of moles of oxygen
16 W=-n*R*T*log(P1/P2);
17 printf("Minimum work done to compress oxygen = %f J
        " , W )
```

Chapter 3

The First Law of Thermodynamics

Scilab code Exa 3.1 numerical

```
1 //example 3.1
2
3 clear all;
4 clc;
5
6 //given:
7 V1=14;//initial volume of cylinder in m3
8 V2=9;//final volume of cylinder in m3
9 P=2000;//pressure during the operation in N/m2
10 U=(-6000);//internal energy of the system in J
11
12 //to find energy transfered in form of heat:
13 W=-P*(V2-V1);//work done during the operation in J
14 Q=U-W;//energy transfered in form of heat in J
15 printf("energy transfered in form of heat is %f J",Q)
    ;
```

Scilab code Exa 3.2 numerical

```
1 //example 3.2
2
3 clear all;
4 clc;
5
6 //given:
7 R=8.314;//universal gas constant [J/K/mol]
8 T=300;//temperture for the process [K]
9 U=0;//change in internal energy [J]
10 V1=2.28;//initial volume [m3]
11 V2=4.56;//final volume [m3]
12
13
14 //to find the heat lost or gained by the system:
15 W=2.303*R*T*log10(V2/V1);//work done during the
    process [J]
16 Q=W;//heat lost or gained by the system [J]
17 printf("the heat gained by the system is %f J",Q);
```

Scilab code Exa 3.3 numerical

```
1 //example 3.3
2
3 clear all;
4 clc;
5
6 //given:
7 H=29.2;//latent heat of vaporisation [KJ/mol]
8 T=332;//temperature of the system [K]
9 R=8.314;//universal gas constant [J/K/mol]
10
11
12 //to find Q,W,U for 1 mole of bromine vaporizes
```

```

13 //where Q is heat absorbed or evolved
14 //W is the work done by system
15 //U is the change in internak energy
16 Qp=H;//at constant pressure [KJ]
17 W=-R*0.001*T;//workdone [KJ]
18 U=Qp+W;//change in internal energy [KJ]
19 printf("heat absorbed by the bromine vapours is %f
      KJ",Qp);
20 printf("\nworkdone during the process is %f KJ",W);
21 printf("\nchange in internal energy of the system is
      %f KJ",U);

```

Scilab code Exa 3.4 numerical

```

1 //example 3.4
2
3 clear all;
4 clc;
5
6 //given:
7 disp("C7H16(l) + 11O2(g) -> 7CO2(g) + 8H2O(l)");
8 n=-4;//change in no. of moles when reaction proceeds
      from reactants to products
9 T=298;//temperature of the process [K]
10 R=8.314;//universal gas constant [J/K/mol]
11 Qv=-4800;//heat energy at constant volume [KJ]
12
13
14 //to find change in enthalpy of the process:
15 U=Qv;//change in internal energy of system [KJ]
16 H=U+n*R*0.001*T;//change in enthalpy of the system [
      KJ]
17 printf("the change in enthalpy of system is %f KJ",H
      );

```

Scilab code Exa 3.5 numerical

```
1 //example 3.5
2
3 clear all;
4 clc;
5
6 //given:
7 n=1;//number of moles of an given ideal gas
8 T=298;//temperature for the process [K]
9 V1=8.3;//initial volume of the ideal gas [m3]
10 V2=16.8;//final volume of the ideal gas [m3]
11 R=8.314;//universal gas constant [J/K/mol]
12
13
14 //to find the Q,W,H
15 //where Q is heat absorbed or evolved by the system
16 //W is the net workdone
17 //H is the change in enthalpy of system
18 W=-2.303*R*T*log10(V2/V1);// [J]
19 Q=-W;// [J]
20 disp("H=U+PV ,where U is change in internal energy
      which is zero due to isothermal process");
21 disp("PV where V is change in volume of system ,PV=
      RT & RT==0 since T i.e change in temp is zero for
      system");
22 disp("Therefore ,the change in enthalpy is 0J");
23 printf("the workdone by system is %f J",W);
24 printf("\nthe heat evolved is %f J",Q);
```

Scilab code Exa 3.6 numerical

```

1 //example 3.6
2
3 clear all;
4 clc;
5
6 //given:
7 T1=323;//intial temperature of water[K]
8 T2=373;//final temperature of water[K]
9 Cp=75.29;//specific heat of water[J/K/mol]
10 w=100;//weight of water[g]
11 mol.wt=18;//molecular weight of water[g/mol]
12
13 //to find the change in enthalpy(H) of water
14 n=w/mol.wt;//no. of moles of water[moles]
15 H=(n*Cp*(T2-T1))*0.001;//change in enthalpy of water
    [J]
16 printf("The change in enthalpy of water is %f J",H);

```

Scilab code Exa 3.7 numerical

```

1 //example 3.7
2
3 clear all;
4 clc;
5
6 disp("SO2 + 0.5O2 -> SO3");
7 //given:
8 U=-97030;//heat of reaction[J]
9 n=1-(1+0.5);//change in no. of moles
10 R=8.314;//universal gas constant[J/K/mol]
11 T=298;//temperature during the reaction[K]
12
13
14 //to find the change in enthalpy of reaction(H)
15 H=U+n*R*T;//change in enthalpy of reaction[J]

```

```
16 printf("The change in enthalpy of reaction is %f J "
    ,H);
```

Scilab code Exa 3.8 numerical

```
1 //example 3.8
2
3 clear all;
4 clc;
5
6 disp("i.C(s) + O2(g) -> CO2(g)");
7 //Given:
8 H1=-393.5;//change in enthalpy [KJ/mol]
9 T1=298;//temperature [K]
10 n1=0;//change in no. of moles in reaction moving in
    forward direction
11 R=0.008314;//universal gas constant [KJ/K/mol]
12
13 //to find the change in internal energy (U) of given
    reaction
14 U1=H1-n1*R*T1;//change in internal energy [KJ]
15 printf("The change in internal energy is %f KJ/mol",
    U1);
16
17 disp("ii.C(s) + 0.5O2 -> CO(g)");
18 //Given:
19 H2=-110.5;//change in enthalpy [KJ/mol]
20 T2=298;//temperature [K]
21 n2=1-0.5;//change in no. of moles in reaction moving
    in forward direction
22 R=0.008314;//universal gas constant [KJ/K/mol]
23
24 //to find the change in internal energy (U) of given
    reaction
25 U2=H2-n2*R*T2;//change in internal energy [KJ]
```

```
26 printf("The change in internal energy is %f KJ/mol",  
    U2);
```

Scilab code Exa 3.9 numerical

```
1 //example 3.9  
2  
3 clear all;  
4 clc;  
5  
6 disp("The standard heat of combustion of");  
7 disp("2C6H6(l)+ 15O2(g)-> 12 CO2(g)+ 6 H2O(l)");  
8 disp("H1(standard heat of combustion)=-6536 KJ/mol")  
    ;  
9 //Given:  
10 H1=-6536; //standard heat of combustion [KJ/mol]  
11  
12 //to find the standard heat of rxn for combustion of  
    1 mole of C6H6  
13 disp("C6H6(l)+ 7.5 O2(g)-> 6 CO2(g)+ 6 H2O(l)");  
14 H2=H1/2; //standard heat of combustion [KJ/mol]  
15 printf(" H2(standard heat of combustion for 1 mole  
    of C6H6)= %f KJ/mol",H2);
```

Scilab code Exa 3.10 numerical

```
1 //example 3.10  
2  
3 clear all;  
4 clc;  
5 disp("N2(g)+3H2(g)-> 2NH3(g)");  
6  
7 //Given:
```



```

8 H=-92.22; //standard heat of reaction [KJ/mol]
9
10 //to find the standard heat of formation of one mole
    of product
11 H1=H/2; //standard heat of formation of 1 mole [KJ/
    mol]
12 printf("H(heat of formation of 1 mole of product)=
    %f KJ/mol" , H1);

```

Scilab code Exa 3.11 numerical

```

1 //example 3.11
2
3 clear all;
4 clc;
5
6 disp("C2H5OH(l)+3O2(g)->2CO2(g)+3H2O(l)");
7 //Given:
8 T=298; //temperature during the reaction [K]
9 Hw=-285.83; //standard heat of formation of liquid
    water [KJ/mol]
10 He=-277.69; //standard heat of formation of liquid
    ethanol [KJ/mol]
11 Hco2=-393.51; //standard heat of formation of carbon
    dioxide [KJ/mol]
12 Ho2=0; //standard heat of formation of oxygen gas [KJ/
    mol]
13
14 //to find the standard heat of reaction
15 H=2*Hco2+3*Hw-He-3*Ho2; //standard heat of reaction
16 printf("H(standard heat of reaction)=%f KJ/mol",H);

```

Scilab code Exa 3.12 numerical

```

1 //example 3.12
2
3 clear all;
4 clc;
5
6 disp("CO(g)+NO(g) ->0.5N2(g)+CO2(g)");
7 //Given:
8 Hrxn=-374; //standard heat of reaction [KJ/mol]
9 Hno=90.25; //standard heat of formation of NO[KJ/mol]
10 Hco2=-393.51; //standard heat of formation of CO2[KJ/
    mol]
11 Hn2=0; //standard heat of formation of N2[KJ/mol]
12 T=298; //temperature of reaction [K]
13
14 //to find the value of standard heat of formation of
    CO
15 Hco=0.5*Hn2+Hco2-Hno-Hrxn; //standard heat of
    formation of CO[KJ/mol]
16 printf("Hco(standard heat of formation)=%f KJ/mol",
    Hco);

```

Scilab code Exa 3.13 numerical

```

1 //example 3.13
2
3 clear all;
4 clc;
5
6 //Given:
7 //("C3H6(g)+H2(g)->C3H8(g);H1=-29.6 Kcal");
8 //("C3H8(g)+5O2(g)->3CO2(g)+4H2O(l) ; H2=-530.6
    Kcal");
9 //("C(s)+O2(g)->CO2(g);H3=-94.0 Kcal");
10 //("H2(g)+0.5O2(g)->H2O(l);H4=-68.3 Kcal. ");
11 H1=-29.6; //the standard heat of hydrogenation of

```

```

    gaseous propylene to propane[Kcal]
12 H2=-530.6; //the heat of combustion of propane[Kcal]
13 H3=-94.0; //the heat of formation of carbon dioxide[
    Kcal]
14 H4=-68.3; //the heat of formation of liquid water[
    Kcal]
15
16
17 //to find the heat of combustion and formation of
    propylene
18 disp("C3H6(g)+4.5O2(g)->3CO2(g)+3H2O(l)");
19 H5=[3*H3+4*H4]-[H1+H2]; // [Kcal]
20 printf("\n H(standard heat of combustion)=%f Kcal.\n
    ",H5);
21 disp("3C(s)+3H2(g)->C3H6(g)");
22 H6=-H5+3*H3+3*H4; // [Kcal]
23 printf("\n H(standard heat of formation)=%f Kcal. \n
    ",H6);

```

Scilab code Exa 3.14 numerical

```

1 //example 3.14
2
3 clear all;
4 clc;
5
6 //Given:
7 H1=-114.1; //standard heat of reaction:2NO(g)+O2(g)
    ->2NO2(g);[KJ/mol]
8 H2=-110.2; //standard heat of reaction:4NO2(g)+O2(g)
    ->2N2O5(g);[KJ/mol]
9 H3=180.5; //standard heat of reaction:N2(g)+O2(g)->2
    NO(g);[KJ/mol]
10
11

```

```

12 //to find the heat of formation of N2O5
13 //reaction:N2(g)+2.5O2(g)->N2O5(g)
14 H4=(2*H1+H2+2*H3)/2; //standard heat of formation of
    N2O5[KJ/mol]
15 printf("H(standard heat of formation of N2O5)=%f KJ/
    mol",H4);

```

Scilab code Exa 3.15 numerical

```

1 //example 3.15
2
3 clear all;
4 clc;
5
6 //Given:
7 Hc=-5645; //standard enthalpy of combustion of
    reaction:C12H22O11(s)+12O2(g)->12CO2(g)+11H2O(l)
    [KJ/mol]
8 Hf1=-393.51; //standard heat of formation of CO2: C(s)
    +O2(g)->CO2(g) [KJ/mol]
9 Hf2=-285.83; //standard heat of formation of H2O: H2(
    g)+0.5O2(g)->H2O(l) [KJ/mol]
10
11
12 //to find the standard heat of formaton of solid
    sucrose
13 //reaction:12C(s)+11H2(g)+5.5O2(g)->C12H22O11(s)
14 Hf=12*Hf1+11*Hf2-Hc; // [KJ/mol]
15 printf("Hf(standard heat of formation of solid
    sucrose)=%f KJ/mol",Hf);

```

Scilab code Exa 3.16 numerical

```

1 //example 3.16
2
3
4 clear all;
5 clc;
6
7 //Given:
8 Hf1=-46.11; //standard heat of formation of NH3 at
   298K //reaction:0.5N2(g)+1.5H2(g)->NH3(g) [KJ/mol
   ]
9 Cp1=29.125; //molar heat capacity at constant
   pressure for N2(g) [J/K/mol]
10 Cp2=28.824; //molar heat capacity at constant
   pressure for H2(g) [J/K/mol]
11 Cp3=35.06; //molar heat capacity at constant pressure
   for NH3(g) [J/K/mol]
12 T1=298; //initial temperature [K]
13 T2=400; //final temperature [K]
14
15
16 //to find the standard heat of formation of NH3 at
   400K for same reaction
17 Cp=Cp3-0.5*Cp1-1.5*Cp2; // [J/K/mol]
18 T=T2-T1; // [K]
19 Hf2=Hf1+Cp*0.001*T; //standard heat of formation for
   NH3 at 400K[KJ/mol]
20 printf("\n Hf2(standard heat of formation for NH3 at
   400K = %f KJ/mol. \n",Hf2);

```

Scilab code Exa 3.17 numerical

```

1 //example 3.17
2
3 clear all;
4 clc;

```

```

5
6 //Given:
7 Cp1=poly([29.07, -0.836*10^-3, 20.1*10^-7], 'T', 'c');
    //heat capacity for H2(g)
8 Cp2=poly([25.72, 12.98*10^-3, -38.6*10^-7], 'T', 'c');
    //heat capacity for O2(g)
9 Cp3=poly([30.36, 9.61*10^-3, 11.8*10^-7], 'T', 'c');//
    heat capacity for H2O(g)
10 Hf1=-241820;//standard heat of formation H2O(g) at
    298K :          H2(g)+ 0.5O2(g)->H2O(g) [J/
    mol]
11 T1=298;//initial temperature [K]
12 T2=1273;//final temperature [K]
13
14 //to find the standard heat of formation of H2O(g)
    at 1273K
15 Cp=Cp3-0.5*Cp2-Cp1;//heat capacity for the formation
    [J/K/mol]
16 i=horner(Cp,1273)-horner(Cp,298);// [J/mol]
17 Hf2=(Hf1-i)*0.001;// [KJ/mol]
18 printf("The heat of formation of H2O at 1273K = %f
    KJ/mol", Hf2);

```

Scilab code Exa 3.18 numerical

```

1 //example 3.18
2
3 clear all;
4 clc;
5
6 //given:
7 H1=435;//bond dissociation energy for: CH4->CH3+H [
    KJ/mol]
8 H2=364;//bond dissociation energy for:CH3->CH2+H [KJ
    /mol]

```

```

9 H3=385; //bond dissociation energy for:CH2->CH+H [KJ/
  mol]
10 H4=335; //bond dissociation energy for:CH->C+H [KJ/
  mol]
11
12
13 //to find the C-H bond energy of CH4 from the above
  bond energies
14 H=(H1+H2+H3+H4)/4; //the bond energy for C-H bond in
  CH4 [KJ/mol]
15 printf("\n H(the C-H bond energy in CH4)=%f KJ/mol.
  \n", H);

```

Scilab code Exa 3.19 numerical

```

1 //example 3.19
2
3 clear all;
4 clc;
5
6 //Given:
7 H1=-84.68; //heat of formation : 2C(s)+3H2(g)->C2H6(g)
  ) [KJ/mol]
8 H2=2*716.68; //heat of formation : 2C(s)->2C(g) [KJ]
9 H3=3*436; //heat of formation : 3H2(g)->6H(g) [KJ]
10 H4=412; //taking it as bond energy for one C-H bond[
  KJ/mol]
11
12
13 //to find the C-C bond energy in ethane bond
14 H=H2+H3-H1; //heat of reaction : C2H6(g)->2C(g)+6H(g)
  [KJ/mol]
15 H5=H-6*H4; //bond energy for one C-C bond in ethane
  bond [KJ/mol]
16 printf("\n Hc-c(bond energy for one C-C bond in

```

```
ethane bond)=%f KJ/mol.\n" ,H5);
```

Scilab code Exa 3.20 numerical

```
1 //example 3.20
2
3 clear all;
4 clc;
5
6 //Given:
7 //MgBr2(s)→Mg(s)+Br2(l)→Mg(g)+Br2(l)→Mg(g)+Br2
   (g)→Mg(g)+2Br(g)→Mg+2(g) + 2e(g) + 2Br(g)→
   Mg+2(g) + 2Br-(g)
8 H1=-524;//enthalpy of formation of MgBr2(s) from its
   element [KJ/mol]
9 H2=148;//enthalpy of sublimation of Mg(s) [KJ/mol]
10 H3=31;//enthalpy of vaporization of Br2(l) [KJ/mol]
11 H4=193;//enthalpy of dissociation Br2 to 2Br(g) [KJ/
   mol]
12 H5=2187;//enthalpy of ionization of Mg(g) to Mg+2(g)
   [KJ/mol]
13 H6=-650;//enthalpy of formation of Br-(g) [KJ/mol]
14
15 //to find the lattice enthalpy of magnesium bromide
16 H=-H1+H2+H3+H4+H5+H6;//lattice enthalpy [KJ/mol]
17 printf("\n H(lattice enthalpy of magnesium bromide)=
   %f KJ/mol.\n" ,H);
```

Scilab code Exa 3.21 numerical

```
1 //example 3.21
2
3 clear all;
```



```

4  clc;
5
6  //Given:
7  Cp1=poly([26.0, 43.5*10^-3, -148.3*10^-7], 'T', 'c');
   //heat capacity for CO2(g) [J/K/mol]
8  Cp2=poly([30.36, 9.61*10^-3, 11.8*10^-7], 'T', 'c');//
   heat capacity for H2O(g) [J/K/mol]
9  Cp3=poly([27.30, -5.23*10^-3, -0.04*10^-7], 'T', 'c');
   //heat capacity for N2(g) [J/K/mol]
10 H1=-881.25;//heat of combustion of methane at 298K[
    KJ/mol]:          CH4 + 2O2(g) --> CO2(g) +
    2H2O(l)
11 H2=43.6;//heat of vaporization of water at 298K[KJ/
    mol]:          H2O(l)-->H2O(g)
12 T1=298;//initial temperature [K]
13
14
15 //to find the maximum flame temperature when one
    mole of methane is burnt completely in
    calculated amount of air(N2 to O2 ratio 4), at
    constant pressure
16 H=(H1+2*H2);//enthalpy of reaction: CH4 + 2O2(g) -->
    CO2(g) + 2H2O(g) [KJ]
17 printf("H(enthalpy of reaction)=%f KJ/mol.\n",H);
18 printf("H(enthalpy of heating)=%f KJ/mol. \n",-H);
19 Cp=Cp1+2*Cp2+8*Cp3;//[J/K/mol]
20 P=poly([0, 305.12, 52.28*10^-3, -41.66*10^-7], '
    T', 'c');
21 i=horner(P,298);
22 c1=-H*1000+i;
23 P1=poly([-c1, 305.12, 52.28*10^-3, -41.66*10^-7]
    , 'T', 'c');
24 T2=roots(P1)
25 printf("T2 (maximum flame temperature)= %f K",T2(1))
    ;

```

Chapter 4

Defining Thermodynamic State The State Postulate

Scilab code Exa 4.1 numerical

```
1 //example 4.1
2
3 clear all;
4 clc;
5
6 //Given:
7 M=2.5; //mass of the substance [Kg]
8 x=0.6; //mass fraction for vapour phase
9 P=7; //pressure [atm]
10 T=438; //temperature [K]
11
12 //to find the mass of substance present in liquid
    and vapour phase
13 Ml=(1-x)*M; //mass fraction of liquid phase [Kg]
14 Mg=x*M; //mass fraction of vapour phase [Kg]
15
16 printf("M(liquid phase)=%f Kg , M(vapour phase)=%f
    Kg" ,Ml ,Mg)
```

Scilab code Exa 4.2 numerical

```
1 //example 4.2
2
3 clear all;
4 clc;
5
6 //Given:
7 V1=0.0177; //specific volume of saturated liquid [m3/
   Kg]
8 Vg=4.43; //specific volume of saturated vapour [m3/Kg]
9 P=7; //pressure [atm]
10 T=438; //temperature [K]
11 x=0.6; //fraction of vapour phase
12 M=2.5; //mass of the substance [Kg]
13
14 //to find the total volume occupied by the mixture
15 V=[(1-x)*V1+x*Vg]*M; //total volume occupied [m3]
16 printf("Total volume occupied =%f m3", V)
```

Scilab code Exa 4.3 numerical

```
1 //example 4.3
2
3 clear all;
4 clc;
5
6 //Given:
7 M=2.5; //mass of a substance [Kg]
8 x=0.6; //fraction of vapour phase
9 Ug=1105; //specific internal energy of saturated
   vapour [J/Kg]
```

```
10 U1=298;//specific internal energy of saturated
    liquid [J/Kg]
11
12
13 //to find the total internal energy of the mixture
14 U=M*[(1-x)*U1+x*Ug];
15 printf("The total internal energy of the mixture =
    %f J", U)
```

Chapter 5

The Second Law of Thermodynamics

Scilab code Exa 5.1 numerical

```
1 //example 5.1
2
3 clear all;
4 clc;
5
6 //given:
7 T1=373;//initial temperature [K]
8 T2=573;//final temperature [K]
9 Q2=750;//Heat absorbed by carnot engine[J]
10
11 //to find the workdone , heat rejected , and
    efficiency of the engine
12 e=(T2-T1)/T2;//efficiency of the engine
13 W=e*Q2;//Workdone by the engine[J]
14 Q1=T1*Q2/T2;//Heat rejected by the engine[J]
15
16 printf("Efficiency of the engine = %f ", e);
17 printf("\n Workdone by the engine = %f J", W);
18 printf("\n Heat rejected by the engine = %f J", Q1);
```

Scilab code Exa 5.2 numerical

```
1 //example 5.2
2
3 clear all;
4 clc;
5
6 //given:
7 T1=250; //temperature of heat rejection [K]
8 T2=1000; //temperature of heat absorption [K]
9
10 //to analyse the efficiency of the engine
11 e=1-(T1/T2);
12 printf("Efficiency of the corresponding carnot
    engine = %f\n",e);
13 disp("Therefore , the inventors claim of 80%
    efficiency is absurd.The patent application
    should be rejected");
```

Scilab code Exa 5.3 numerical

```
1 //example 5.3
2
3 clear all;
4 clc;
5
6 //Given:
7 T1=323; //temperature [K]
8 T2=423; //temperature [K]
9 W=1.3; //work [KJ]
10
```

```

11
12 //To find the minimum heat required from heat source
    to yield the above work
13 e=(T2-T1)/T2;//efficiency
14 Q2=W/e;//minimum heat withdrawal from heat source [KJ
    ]
15 printf("Minimum heat withdrawal from heat source=%f
    KJ",Q2);

```

Scilab code Exa 5.5 numerical

```

1 //example 5.5
2
3 clear all;
4 clc;
5 //Given:
6 T=298;//Temperature [K]
7 n=1;//no. of moles
8 V1=500;//initial volume [cm3]
9 V2=1000;//final volume [cm3]
10 R=8.314;//Universal gas constant [J/mol/K]
11
12 //to find the molar entropy change
13 S=R*log(V2/V1)//molar entropy change at constant
    temperature [J/K]
14 printf("Molar entropy change of argon = %f J/K", S);

```

Scilab code Exa 5.6 numerical

```

1 //example 5.6
2
3 clear all;
4 clc;

```

```

5
6 //Given:
7 W=1728;//Isothermal and reversible work done[J/mol]
8 T=298;//Isothermal temperature[K]
9
10 //to find the change in molar entropy when the gas
    expands isothermally and reversibly
11
12 S=W/T;//change in molar entropy for isothermal and
    reversible process
13 printf("The change in molar entropy = %f J/mol/K", S
    );

```

Scilab code Exa 5.7 numerical

```

1 //example 5.7
2
3 clear all;
4 clc;
5
6 //Given:
7 H=-92.22;//Standard reaction enthalpy [KJ]
8 T=298;//Temperature [K]
9
10
11 //To find the change in entropy of the surroundings
    at 298K
12 //standard reaction enthalpy is H. Therefore , heat
    gained by the surroundings at 298K is -H
13 S=-H*1000/T;//Change in entropy[J/K]
14 printf("Change in entropy of the surroundings at 298
    k = %f J/K",S);

```

Scilab code Exa 5.8 numerical

```
1 //example 5.8
2
3 clear all;
4 clc;
5
6 //Given:
7 T1=298; //Initial Temperature [K]
8 T2=573; //Final Temperature [K]
9 Cv=29.1; //Specific Heat capacity of argon gas [J/K/
    mol]
10 n=1; //no. of moles
11
12
13 //To find the change in entropy for argon gas
14 S=n*Cv*log(T2/T1); //Change in entropy [J/K]
15 printf("The change in entropy of the argon gas is %f
    J/K", S);
```

Scilab code Exa 5.9 numerical

```
1 //example 5.9
2
3 clear all;
4 clc;
5
6 //Given:
7 T1=276; //Initial temperature [K]
8 Tf=278.7; //Freezing point temperature [K]
9 Tb=353.3; //Boiling point temperature [K]
10 T2=373; //Final temperature [K]
11 Hf=9870; //Standard enthalpy of fusion [J/mol]
12 Hv=30800; //Standard enthalpy of vaporization [J/mol]
13 Cp=136.1; //Specific heat capacity of benzene [J/K/mol]
```

```

]
14 mol.wt=78; //molecular weight of benzene[g/mol]
15 mass=200; //weight of solid benzene[g]
16 disp("      Cp doesnot change within this temp limit"
      );
17
18 //To find the total entropy change of solid
19 n=mass/mol.wt; //no. of moles
20 S1=n*Cp*log(Tf/T1); //entropy change in heating [J/K]
21 S2=n*Hf/Tf; //entropy change in melting [J/K]
22 S3=n*Cp*log(Tb/Tf); //entropy change in heating [J/K]
23 S4=n*Hv/Tb; //entropy change in vaporization [J/K]
24 S5=n*Cp*log(T2/Tb); //entropy change in heating [J/K]
25 S=S1+S2+S3+S4+S5; //total entropy change in heating
      from 276 to 373K
26
27 printf("Total entropy change in heating 200g benzene
      from 3 to 100'C is %f J/K",S);

```

Scilab code Exa 5.10 numerical

```

1 //example 5.10
2
3 clear all;
4 clc;
5
6 //Given:
7 mass=32; //weight of methane gas [gm]
8 P1=6*10^5; //Initial temperature [N/m2]
9 P2=3*10^5; //Final pressure [N/m2]
10 mol.wt=16; //molecular weight of methane gas [g/mol]
11 T=298; //Temperature [K]
12 //Isothermal process
13 R=8.314; //Universal gas constant [J/K/mol]
14

```

```
15 //To find the change in entropy of the gas
16 n=mass/mol.wt;//no. of moles
17 S=n*R*log(P1/P2);//change in entropy of gas[J/K]
18 printf("The change in entropy of the gas is %f J/K",
    S);
```

Scilab code Exa 5.11 numerical

```
1 //example 5.11
2
3 clear all;
4 clc;
5
6 //Given:
7 N=3;//Total no. of balls
8 Nb=2;//Black balls
9 Nw=1;//white ball
10
11 //To find the total no. possible configuration
12 w=prod(1:N)/prod(1:Nb)/prod(1:Nw);
13
14 printf("The total no. of possible configurations are
    %f ",w);
```

Scilab code Exa 5.12 numerical

```
1 //example 5.12
2
3 clear all;
4 clc;
5
6 //Given:
7 n=1;//no. of moles
```

```

8 T=273; //temperature [K]
9 Hf=6000; //enthalpy of fusion at 273K [J/mol]
10 k=1.38*(10^-23); //boltzmann constant [J/K]
11
12 //To find the relative number of distinguishable
    quantum states in 1 mole of water and ice at 273K
13
14 p=Hf/(k*T)/2.303;
15 w=10^(p); //w is the relative no. of distinguishable
    quantum states
16 printf("The relative no. of distinguishable quantum
    states in 1 mole of water and ice at 273K is %f",
    w);

```

Scilab code Exa 5.13 numerical

```

1 //example 5.13
2
3 clear all;
4 clc;
5
6 //Given:
7 T=300; //temperature [K]
8 n=4; //no. of moles of an ideal gas
9 P1=2.02*10^5; //initial pressure [N/m2]
10 P2=4.04*10^5; //final pressure [N/m2]
11 R=8.314; //Universal gas constant [J/K/mol]
12
13 //To find the value of Gibb's free energy
14 G=n*R*T*log(P2/P1); // [J]
15 printf(" The change in Gibbs free energy is %f J",G)
    ;

```

Scilab code Exa 5.14 numerical

```
1 //example 5.14
2
3 clear all;
4 clc;
5
6 //Given:
7 n=1; //no. of moles
8 T=300; //temperature [K]
9 V1=2; //initial volume [m3]
10 V2=20; //final volume [m3]
11 R=8.314; //Universal gas constant [J/K/mol]
12
13
14 //To find the work function or Helmholtz free energy
15 A=-n*R*T*log(V2/V1); //Change in work function [J/mol]
16 printf("The change in Helmholtz free energy is %f J/
    mol",A);
```

Scilab code Exa 5.15 numerical

```
1 //example 5.15
2
3 clear all;
4 clc;
5
6 disp("C6H12O6(s) + 6O2(g) --> 6CO2(g) + 6H2O(l)");
7 //Given:
8 T=298; //Temperature [k]
9 R=8.314; //Universal gas constant [J/K/mol]
10 S=182.45; //standard entropy change at 298K [J/K]
11 U=-2808; //change in internal energy at 298K [KJ/mol]
12 //reaction is taking place in bomb calorimeter so no
    volume change
```

```

13 //therefore U=Q at constant volume
14
15 //To find the energy change that can be extracted as
    heat and work
16 A=U-T*S*0.001;//Energy extracted as heat [KJ/mol]
17 Wmax=A;//work done [KJ/mol]
18 dn=6-6;//change in no. of moles
19 H=U+dn*R*T;//Change in enthalpy of the bomb
    calorimeter [KJ]
20 printf("The energy change that can be extracted as
    heat is %f KJ/mol", A);
21 printf("\nThe energy change that can be extracted as
    work is %f KJ/mol", -A);
22 printf("\nThe change in enthalpy of bomb calorimeter
    is %f KJ/mol",H);

```

Scilab code Exa 5.16 numerical

```

1 //example 5.16
2
3 clear all;
4 clc;
5
6 disp("C8H18(g)+12.5O2(g)-->8CO2(g)+9H2O(l)");
7 //Given:
8 T=298;//temperature [K]
9 S=421.5;//change in entropy [J/K]
10 H=-5109000;//Heat of reaction [J]
11 R=8.314;//Universal gas constant [J/K/mol]
12 dn=8-(1+12.5);//change in no. of moles
13
14
15 //To find the helmholts free energy and Gibbs free
    energy
16 U=H;//[J]

```

```

17 A=U-T*S;//Change in helmholts free energy[J]
18 G=A+dn*R*T;//Change in Gibbs free energy[J]
19 printf("The change in Helmholts free energy is %f J"
    ,A);
20 printf("\nThe change in Gibbs free energy is %f J",G
    );

```

Scilab code Exa 5.17 numerical

```

1 //example 5.17
2
3 clear all;
4 clc;
5
6 disp("C3H6(g)+4.5O2(g)-->3CO2(g)+3H2O(1)");
7 //Given:
8 S=-339.23;//standard change in entropy [J/K]
9 T=298;//temperature[K]
10 Hf1=20.42;//enthalpy of formation of C3H6(g)[J]
11 Hf2=-393.51;//enthalpy of formation of CO2(g)[J]
12 Hf3=-285.83;//enthalpy of formation of H2O(1)[J]
13 dn=3-4.5-1;//change in no. of moles
14 R=8.314;//Universal gas constant[J/K/mol]
15
16 //To find the Helmholts free energy and Gibbs free
    energy
17 H=3*Hf2+3*Hf3-Hf1;//Enthalpy of the reaction[J]
18 U=H-dn*R*0.001*T;//Change in internal energy of the
    reaction[J]
19 A=U-T*S*0.001;//Helmholts free energy change[J]
20 G=A+dn*R*0.001*T;//Gibbs free energy change[J]
21 printf("The change in Helmholts free energy is %f J"
    ,A);
22 printf("\nThe change in Gibbs free energy is %f J ",
    G);

```

Scilab code Exa 5.19 numerical

```
1 //example 5.19
2
3 clear all;
4 clc;
5
6 disp("CH4(g)+2O2(g)-->CO2(g)+2H2O(l)");
7
8 //Given:
9 S1=-242.98;//standard entropy change for the
   combustion reaction [J/K]
10 Hf1=-74.81;//Enthalpy of formation of CH4(g) [KJ/mol]
11 Hf2=-393.51;//Enthalpy of formation of CO2(g) [KJ/mol
   ]
12 Hf3=-285.83;//Enthalpy of formation of H2O(l) [KJ/mol
   ]
13 T=298;//temperature [K]
14
15 //To find the total entropy change
16 H=Hf2+2*Hf3-Hf1;//Change in enthalpy of reaction [KJ]
17 S2=-H*1000/T;//Change in entropy of the surrounding [
   J/K]
18 Stotal=(S1+S2)*0.001;//Total entropy change
19 printf("The total change in entropy is %f KJ/K",
   Stotal);
```

Scilab code Exa 5.20 numerical

```
1 //example 5.20
2
```



```

3 clear all;
4 clc;
5
6 disp("2H2(g)+O2(g)-->2H2O(l)");
7 //Given:
8 Hf1=-285.83;//standard enthalpy of formation of H2O(
   l)[KJ/mol]
9 S=-327;//Standard entropy change for the same
   reaction[J/K]
10 T=298;//temperature[K]
11
12
13 //To find the spontaneity of the reaction
14 H=2*Hf1-0-0;//Enthalpy of the reaction[KJ/mol]
15 G=H-T*S*0.001;//Change in Gibbs free energy[KJ]
16 printf("The change in Gibbs free energy is %f KJ\n "
   ,G);
17 disp("As change in Gibbs free energy is negative.
   Therefore ,the reaction is spontaneous")

```

Scilab code Exa 5.21 numerical

```

1 //example 5.21
2
3 clear all;
4 clc;
5
6 disp("CH4(g)+2O2(g)-->CO2(g)+2H2O(l)");
7 //Given:
8 S=-242.98;//standard entropy change for reaction [J/
   K]
9 T=298;//temperature[K]
10 Gf1=-50.72;//standard Gibbs free energy of formation
   for CH4(g)[KJ/mol]
11 Gf2=-394.36;//standard Gibbs free energy of

```

```

    formation for CO2(g) [KJ/mol]
12 Gf3=-237.13; //standard Gibbs free energy of
    formation for H2O(l) [KJ/mol]
13
14
15 //To find the standard enthalpy of reaction
16 G=Gf2+2*Gf3-Gf1; //Standard Gibbs free energy for
    reaction [KJ/mol]
17 H=G+T*S*0.001; //Standard enthalpy of reaction [KJ]
18 printf("The standard enthalpy of reaction is %f KJ",
    H);

```

Scilab code Exa 5.22 numerical

```

1 //example 5.22
2
3 clear all;
4 clc;
5
6 disp("C6H12O6(s)+6O2(g)-->6CO2(g)+6H2O(l)");
7 //Given:
8 mass=25; //mass of glucose for combustion under
    standard condition [gm]
9 T=298; //temperature [K]
10 Gf1=-910; //Standard Gibbs free energy of formation
    for C6H12O6 [KJ/mol]
11 Gf2=-394.4; //Standard Gibbs free energy of formation
    for CO2(g) [KJ/mol]
12 Gf3=-237.13; //Standard Gibbs free energy of
    formation for H2O(l) [KJ/mol]
13 mol.wt=180; //molecular weight of glucose [gm/mol]
14
15 //To find the maximum energy that can be extracted
    as non-expansion work is equal to the change in
    free energy of the system

```

```

16 G=6*Gf2+6*Gf3-Gf1
17 n=mass/mol.wt;//no. of moles
18 Gactual=G*n;//Gibbs free energy for the combustion
    of 0.139mol of glucose
19 printf("The energy that can be extracted as non-
    expansion work is %f KJ",-Gactual);

```

Scilab code Exa 5.23 numerical

```

1 //example 5.23
2
3 clear all;
4 clc;
5
6 //Given:
7 a=1.39*10^-2;//constant for a vanderwaal's gas[lit2.
    atm/mol2]
8 b=3.92*10^-2;//constant for a vanderwaal's gas[lit2.
    atm/mol2]
9 R=0.082;//Universal gas constant[lit.atm/deg/mol]
10
11 //To find the value of the inversion temperature for
    the gas
12 Ti=(2*a)/(R*b);//inversion temperature [K]
13 printf("The inversion temperature for the gas is %f
    K",Ti);

```

Scilab code Exa 5.26 numerical

```

1 //example 5.26
2
3
4 clear all;

```

```

5  clc;
6
7  //Given:
8  T=169.25; //Boiling point [K]
9  R=8.314; //Universal gas constant [J/K/mol]
10 disp(" dlnP/dT=He/R*T^2");
11 disp(" dlnP/dT=(2.303*834.13/T^2)+(1.75/T)
      -(2.30*8.375*10^-3)");
12 disp(" Therefore using these two equations we
      calculate the He(enthalpy) of ethylene");
13
14 //To find the Enthalpy of vaporization of ethylene
15 x=(2.303*834.13/T^2)+(1.75/T)-(2.30*8.375*10^-3); //
      it is dlnP/dT
16 He=R*0.001*T^2*x; //Enthalpy of vaporization [J/mol]
17 printf("\n\nThe Enthalpy of vaporization of ethylene
      at its boiling point is %f KJ/mol",He);

```

Scilab code Exa 5.27 numerical

```

1  //example 5.27
2
3  clear all;
4  clc;
5
6  //Given:
7  P1=101.3; //Initial Pressure [KPa]
8  P2=60; //Final Pressure [KPa]
9  He=31.8; //Enthalpy of vaporization [KJ/mol]
10 R=8.314; //Universal gas constant [J/K/mol]
11 T1=353.2; //boiling point of benzene at 101.3KPa[K]
12
13 //To find the boiling point of benzene at 60KPa
14 x=(T1^-1)-(R*0.001*log(P2/P1)/He);
15 T2=x^-1; //Boiling point of benzene at 60KPa

```

```
16 printf("The boiling point of benzene at 60KPa is %f
    K",T2);
```

Scilab code Exa 5.28 numerical

```
1 //example 5.28
2
3 clear all;
4 clc;
5
6 //Given:
7 P1=0.016; //Vapour pressure of pure ethanol at 273K[
    bar]
8 P2=0.470; //Vapour pressure of pure ethanol at 333K[
    bar]
9 T1=273; //initial temperature [K]
10 T2=333; //final temperature [K]
11 R=8.314; //Universal gas constant [J/K/mol]
12 P=1.01; //vapour pressure at normal boiling point [bar
    ]
13
14
15 //To find the molar enthalpy of vapourization
16 x=(T2^-1)-(T1^-1);
17 He=-R*0.001*log(P2/P1)/x; //molar enthalpy of
    vaporization [J/mol]
18 t=(T2^-1)-(R*0.001*log(P/P2)/He);
19 T=(t^-1)-273; //normal boiling point [C]
20 printf("The molar enthalpy of vapourization is %f J/
    mol",He);
21 printf("\n\nThe normal boiling point for pure
    ethanol is %f C",T);
```

Scilab code Exa 5.29 numerical

```
1 //example 5.29
2
3 clear all;
4 clc;
5
6 //Given:
7 T2=353.2; //normal boiling point of benzene at
   1.01325 bar [K]
8 T1=298; //temperature [K]
9 R=8.314; //Universal gas constant [J/K/mol]
10 P2=1.01325; //Vapour pressure of benzene [bar]
11 //benzene obey's Trouton's rule
12 disp(" from Troutons rule , ");
13 disp(" He/Tb=85J/K/mol");
14
15 //To find the vapour pressure of benzene at 298K
16 He=85*T2; //molar enthalpy of vapourization [J/K/mol]
17 x=(T2^-1)-(T1^-1);
18 t=-He*x/R;
19 P1=P2/exp(t);
20 printf("\nThe vapour pressure of benzene at 298K is
   %f bar",P1);
```

Scilab code Exa 5.30 numerical

```
1 //example 5.30
2
3 clear all;
4 clc;
5
6 //Given:
7 c=1; //no. of components(only CO2)
8 p=2; //no. of phases(liquid + gas)
```

```

9
10 //To find the degrees of freedom
11 F=c-p+2;//degree of freedom
12 printf("Degrees of freedom is %f \n\n",F);
13 disp("Degrees of freedom 1 means that either
      pressure or temperature can be varied
      independently ,i.e.when temperature is fixed ,
      pressure is automatically fixed");

```

Scilab code Exa 5.31 numerical

```

1 //example 5.31
2
3 clear all;
4 clc;
5
6 //Given:
7 c=1;//no. of components
8 p=1;//no. of phases
9
10 //To find the values of degrees of freedom
11 F=c-p+2;//Degrees of freedom
12 printf("Degrees of freedom is %f\n\n",F);
13 disp("Degrees of freedom 2 means both the pressure
      and temperature can be varied independently");

```

Scilab code Exa 5.32 numerical

```

1 //example 5.32
2
3 clear all;
4 clc;
5

```

```

6 //Given:
7 P=1.75*10^-5; //Vapour pressure of pure water at 293K
   [ torr ]
8 dP=1.1*10^-7; //Lowering in vapour pressure of water
9 //To find the mole fraction of sucrose, so that the
   vapour pressure of water will be lowered by dP
10 x=dP/P; //mole fraction of sucrose
11 disp(x, 'The mole fraction of sucrose is ');

```

Scilab code Exa 5.33 numerical

```

1 //example 5.33
2
3 clear all;
4 clc;
5
6 //Given:
7 P=94.6; //The vapour pressure of pure benzene at 298K
   [ torr ]
8 n1=20; //no. of moles of pure benzene
9 n2=5; //no. of moles of pure naphthalene
10
11 //To find the partial vapour pressure of benzene
   over the solution
12 x=n1/(n1+n2); //(mole fraction of benzene)
13 p=x*P; //the partial vapour pressure of benzene [ torr ]
14 printf("The partial vapour pressure of benzene is %f
   torr", p);

```

Scilab code Exa 5.34 numerical

```

1 //example 5.34
2

```



```

3 clear all;
4 clc;
5
6 //Given:
7 x=0.28; //mole fraction of solute
8 R=8.314; //Universal gas constant [J/K/mol]
9 T=298; //temperature [K]
10
11 //To find the reduction in chemical potential
12 du=R*T*log(1-x); //reduction in chemical potential [J/
    mol]
13 printf("The reduction in chemical potential is %f J/
    mol" ,-du);

```

Scilab code Exa 5.35 numerical

```

1 //example 5.35
2
3 clear all;
4 clc;
5
6 //Given:
7 Kb=0.51; //ebullioscopic constant of water [K*Kg/mol]
8 n=155/180; //no. of moles of glucose
9 m=n/1; // [mol/Kg]
10 Ti=373; //Boiling point temperature of water [K]
11
12 //To find the boiling point of the solution which is
    made by dissolving 155g of glucose in 1000g of
    water
13 Tf=(Ti+Kb*m)-273; //boiling point temperature of the
    solution [C]
14 printf("The boiling point of the solution is %f C",
    Tf);

```

Scilab code Exa 5.36 numerical

```
1 //example 5.36
2
3 clear all;
4 clc;
5
6 //Given:
7 Ti=5.44; //freezing point of pure benzene [K]
8 Tf=4.63; //freezing point of solution [K]
9 m1=2.12; //mass of the solute [gm]
10 m2=125; //mass of the benzene [gm]
11 Kf=5.12; //cryoscopic constant of pure benzene [K*Kg/
    mol]
12
13 //To find the molar mass of solute (M2)
14 dTf=Ti-Tf; //depression in freezing point [K]
15 M2=(m1*1000*Kf)/(m2*dTf); //molar mass of solute
16 printf("The molar mass of solute is %f",M2);
```

Scilab code Exa 5.38 numerical

```
1 //example 5.38
2
3 clear all;
4 clc;
5
6 disp("N2(g)+3H2(g)<=>2NH3(g)");
7 //Given:
8 T=298; //Temperature [K]
9 Gf1=-16450; //Gibb's free energy of formation for NH3
    (g) [J/mol]
```

```

10 R=8.314; // Universal gas constant [J/K/mol]
11
12 //To find the Kp value of the above reaction
13 Gf=2*Gf1 //Gibb's free energy for the reaction [KJ]
14 x=Gf/R/T
15 Kp=exp(-x);
16 disp(Kp, 'The Kp for above reaction is ');

```

Scilab code Exa 5.39 numerical

```

1 //example 5.39
2
3 clear all;
4 clc;
5
6 disp(" 0.5N2(g) + 1.5H2(g) <=> NH3(g)");
7 //Given:
8 T=298; //Temperature [K]
9 Kp=900; //Equilibrium constant for above reaction
10 P1=0.32; //partial pressure of N2(g) [bar]
11 P2=0.73; //partial pressure of H2(g) [bar]
12 P3=0.98; //partial pressure of NH3(g) [bar]
13 R=8.314; //Universal gas constant [J/K/mol]
14
15 //To find the reaction Gibb's energy
16 G=-R*T*log(Kp);
17 x=(P1^0.5)*(P2^1.5);
18 p=P3/x;
19 Gr=(G+R*T*log(p))*0.001;
20 printf("The reaction Gibbs free energy is %f KJ/mol
    ",Gr);

```

Scilab code Exa 5.40 numerical

```

1 //example 5.40
2
3 clear all;
4 clc;
5
6 disp("N2(g)+3H2(g)<=>2NH3(g)");
7
8 //Given:
9 Kp1=5.85*10^5; //equilibrium constant at 298K
10 H1=-46.11; //standard enthalpy of formation of NH3(g)
    [KJ/mol]
11 T1=298; //Initial temperature [K]
12 T2=423; //Final temperature [K]
13 R=8.314; //Universal gas constant [J/K/mol]
14
15 //To find the Kp at 423K temperature
16 H=2*H1; //enthalpy for reaction; [KJ]
17 t=(T1^-1)-(T2^-1);
18 x=-H*t/(R*0.001);
19 Kp2=Kp1*exp(x);
20 disp(Kp2, 'The Equilibrium constant for reaction at
    423K is ');

```

Scilab code Exa 5.41 numerical

```

1 //example 5.41
2
3 clear all;
4 clc;
5
6
7 disp("Zn(s) | ZnCl2(aq) || CdSO4(aq) | Cd(s)");
8 //For Zn(s) | ZnCl2(aq) || CdSO4(aq) | Cd(s)
9 //Given:
10 T=298; //Temperature [K]

```

```

11 R=8.314; // Universal gas constant [J/K/mol]
12 E1=-0.7618; // Standard electrode potential for Zn2+/
    Zn [volts]
13 E2=-0.403; // Standard electrode potential for Cd2+/Cd
    [volts]
14 F=96500; // Faraday's constant [coulomb/mol]
15 n=2; // no. of electrons balancing
16 // To find the Standard Free energy and equilibrium
    constant
17 Ei=E2-E1; // Standard potential for the reaction [volts
    ]
18 Gi=-n*F*Ei; // Standard Gibb's Free Energy [KJ/mol]
19 Ki=exp(-Gi/R/T); // Equilibrium constant
20 printf("The Free energy for the rection is %f KJ/mol
    ",Gi*0.001);
21 disp(Ki,'The value of equilibrium constant is ');
22
23
24 disp("Cd(s) | CdSO4(aq) , Hg2SO4(s) | Hg(l)");
25 // For Cd(s) | CdSO4(aq) , Hg2SO4(s) | Hg(l)
26 // Given:
27 E3=0.6141; // Standard electrode potential for Hg2SO4(
    s) , SO4^2- / Hg(l) [volts]
28 // To find the standard free energy and equilibrium
    constant
29 Eii=E3-E2; // Standard potential for the reaction [
    volts]
30 Gii=-n*F*Eii; // Standard Gibb's free energy [KJ/mol]
31 Kii=exp(-Gii/R/T); // Equilibrium constant
32 printf("The Free energy for the rection is %f KJ/mol
    ",Gii*0.001);
33 disp(Kii,'The value of equilibrium constant is ');

```

Scilab code Exa 5.42 numerical

```

1 //example 5.42
2
3 clear all;
4 clc;
5
6 //Given:
7 disp("Zn(s) | ZnCl2(soln) || AgCl(s) | Ag-Ag | AgCl(s) | ZnCl2
      (soln) | Zn(s)");
8 m1=0.02; //concentration [M]
9 Y1=0.65; //mean ionic activity coefficient
10 m2=1.5; //concentration [M]
11 Y2=0.29; //mean ionic activity coefficient
12 R=8.314; //Universal gas constant [J/K/mol]
13 T=298; //Temperature [K]
14 F=96500; //Faraday's constant [coulomb/mol]
15
16 //To find the overall e.m.f of the above cell
17 E=R*T*(log(m2*Y2/m1/Y1))*3/2/F; //[volts]
18 printf("The overall e.m.f of the cell is %f volts",E
        );

```

Scilab code Exa 5.43 numerical

```

1 //example 5.43
2
3 clear all;
4 clc;
5
6 //Given:
7 disp("H2(g, 1 atm) | HCl(aq) | HCl(aq) | H2(g, 1 atm)");
8 m1=0.02; //concentration [M]
9 Y1=0.88; //mean ionic activity coefficient
10 m2=1; //concentration [M]
11 Y2=0.81; //mean ionic activity coefficient
12 R=8.314; //universal gas constant [J/K/mol]

```

```

13 T=298; //Temperature [K]
14 F=96487; //Faraday's constant [coulombs/mol]
15 t=0.178; //Tranference number of Cl-1
16
17 //To find the e.m.f of the cell
18 E=-2*t*R*T*(log(m1*Y1/m2/Y2))/F; //e.m.f of the cell [
    volts]
19 printf("The e.m.f of the cell is %f volts",E);

```

Scilab code Exa 5.44 numerical

```

1 //example 5.44
2
3 clear all;
4 clc;
5
6 disp("The values for reaction that goes on within
    the cadmium cell");
7 //Given:
8 n=2; //no. of moles
9 E=1.01463; //standard cadmium cell potential [volts]
10 d=-5*10^-5; //i.e d=dE/dT [V/K]
11 F=96500; // [coulomb/mol]
12 T=298; //Temperature [K]
13
14 //To find the values of dG,dS and dH
15 dG=-n*E*F; //Change in Gibb's free energy [J]
16 dS=n*F*d; //Change in entropy [J/K]
17 dH=dG+T*dS; //change in enthalpy [J]
18 printf(" dG=%f J",dG);
19 printf("\n dS=%f J/K",dS);
20 printf("\n dH=%f J",dH);

```

Chapter 6

The Question of Ideality

Scilab code Exa 6.2 numerical

```
1 //example 6.2
2
3 clear all;
4 clc;
5
6 //Given:
7 T=500; //Temperature [K]
8 P=100; //Pressure [atm]
9 a=3.61; //van der waals constant for CO2[atm.L^2.mol
    ^-2]
10 b=0.0429; //van der waals constant for CO2[L.mol^-1]
11 R=0.082; //Universal gas constant[atm.K-1.mol^-1]
12
13 //To find the molar volume of CO2
14 x=b+(R*T/P);
15 y=a/P;
16 z=a*b/P;
17 p2 = poly([-z y -x 1], 'Vm', 'c');
18 t=roots(p2);
19 printf("The value of molar volume of CO2 is %f L.mol
    ^-1",t(3));
```

Scilab code Exa 6.5 numerical

```
1 //example 6.5
2
3 clear all;
4 clc;
5
6 //Given:
7 b=0.0391; //Van der waals constant [dm3/mol]
8 R=0.082; //Universal gas constant [dm3*atm/mol]
9 P2=1000; //pressure [atm]
10 P1=0; //pressure [atm]
11 T=1273; //Temperature [K]
12
13 //To find the fugacity and fugacity coefficient
14 x=b*(P2-P1);
15 y=R*T;
16 fc=exp(x/y); //fugacity coefficient
17
18 f=P2*fc; //fugacity [atm]
19 printf("The fugacity is %f atm",f);
20 printf("\n The fugacity coefficient is %f ",fc);
```

Scilab code Exa 6.10 numerical

```
1 //example 6.10
2
3 clear all;
4 clc;
5
6 //Given:
```

```

7 m1=0.03; //mass of CO2(g) [gm]
8 w1=44.01; //molecular weight of CO2(g) [gm/mol]
9 m2=250; //mass of water [gm]
10 w2=18.02; //molecular weight of water [gm/mol]
11 k=1.25*10^6; //Henry's law constant [Torr]
12 T=298; //Temperature [K]
13
14 //To find the partial pressure of CO2 gas
15 n1=m1/w1; //no. of moles of CO2
16 n2=m2/w2; //no. of moles of water
17 x1=n1/(n1+n2); //mole fraction of CO2
18 Pco2=k*x1; //Partial pressure of CO2 [Torr]
19 printf("The partial pressure of CO2 gas is %f Torr",
    Pco2);

```

Scilab code Exa 6.11 numerical

```

1 //example 6.11
2
3 clear all;
4 clc;
5
6 //Given:
7 W=1000; //Total mass of a solution [gm]
8 x1=0.5; //mole fraction of Chloroform
9 x2=0.5; //mole fraction of Acetone
10 V1m=80.235; //Partial molar volume of chloroform [cm3/
    mol]
11 V2m=74.166; //Partial molar volume of Acetone [cm3/mol
    ]
12 M1=119.59; //molecular weight of chloroform [gm/mol]
13 M2=58; //molecular weight of Acetone [gm/mol]
14
15 //To find the Volume of the solution
16 nT=W/(x1*M1+x2*M2); //Total no. of moles

```

```
17 V=nT*(x1*V1m+x2*V2m); //Total volume [cm3]
18 printf("The volume of the solution is %f cm3",V);
```

Scilab code Exa 6.12 numerical

```
1 //example 6.12
2
3 clear all;
4 clc;
5
6 //Given:
7 x1=0.5; //mole fraction of chloroform
8 x2=0.5; //mole fraction of p-xylene
9 T=298; //Temperature [K]
10 //to find the excess volume
11 Ve=x1*x2*[0.585+0.085*(x1-x2)-0.165*(x1-x2)^2]; //
    Excess volume measured by using a dilatometer
12 printf("Ve/(cm3.mol^-1) = %f ",Ve);
```

Scilab code Exa 6.14 numerical

```
1 //example 6.14
2
3 clear all;
4 clc;
5
6 //Given:
7 m1=0.01; //molality [m]
8 v11=1;
9 v12=2;
10 Y1=0.71;
11 m2=0.005; //molality [m]
12 v21=1;
```

```

13 v22=1;
14 Y2=0.53;
15
16
17 //To find the activity , molality of the
    electrolytes
18 v1=(v11)+(v12);
19 v2=(v21)+(v22);
20 a1=(m1^v1)*(v11^v11)*(v12^v12)*(Y1^v1);
21 a2=(m2^v2)*(v21^v21)*(v22^v22)*(Y2^v2);
22 x=1/v1;
23 a1m=a1^x;
24 m1m=m1*(v11^v11*v12^v12)^x; //molality [m]
25 y=1/v2;
26 m2m=m2*(v21*v21*v22^v22)^y; //molality [m]
27 a2m=a2^y;
28 disp(a1,'The activity of the electrolyte ZnCl2 is ');
29 disp(a1m,'The mean activity of ZnCl2 is ');
30 disp(m1m,'The mean molality of ZnCl2 in [m] ');
31 disp(a2,'The activity of the electrolyte CuSO4 is ')
    ;
32 disp(a2m,'The mean activity of CuSO4 is ');
33 disp(m2m,'The mean molality of CuSO4 in [m] ');

```

Scilab code Exa 6.15 numerical

```

1 //example 6.15
2
3 clear all;
4 clc;
5
6 //Given:
7 m2=3; //mass of the sucrose [gm]
8 m1=0.1; //mass of water [Kg]
9 Kf=1.86; //cryoscopic constant of water [K*Kg/mol]

```

```

10 dTf=0.16; //Lowering in freezing point [K]
11
12 //To find the molecular weight of sucrose
13 a=m1*dTf;
14 b=Kf*m2;
15 M2=b/a; //molecular weight
16 printf("M2=molecular weight ,      then M2=%f" ,M2);

```

Scilab code Exa 6.16 numerical

```

1 //example 5.16
2
3 clear all;
4 clc;
5
6 //Given:
7 dTf=0.088; //Lowering in freezing point [K]
8 m2=0.45; //mass of sulphur [gm]
9 m1=0.09955; //mass of benzene [gm]
10 Kf=5.07; //cryoscopic constant for benzene [K*Kg/mol]
11
12 //To find the molecular formula of sulphur
13 a=m1*dTf;
14 b=Kf*m2;
15 M2=b/a; //molecular weight of sulphur
16 printf("The molecular weight of sulphur is %f" ,M2);
17 x=M2/32; //no. of sulphur atoms
18 printf("\n The molecular formula of sulphur is S%f" ,
      x);

```

Scilab code Exa 6.17 numerical

```

1 //example 6.17

```

```

2
3 clear all;
4 clc;
5
6 //Given:
7 m2=1.35; //mass of a macromolecule [gm]
8 V=100; //volume of solution [cm^3]
9 R=82; //Universal gas constant [atm.cm^3.K^-1]
10 T=300; //Temperature [K]
11 II=9.9; //osmotic pressure of the solution [cm]
12 d=1; //density
13 p=1013250; //Atmospheric pressure
14 g=980.67; //gravitational field
15
16
17 //To find the molar mass of macromolecule
18 a=m2*R*T*p;
19 b=V*9.9*d*g;
20 M2=a/b; //molar mass of macromolecule
21 printf(" M2 = molar mass of macromolecule ,
        therefore M2 = %f g.mol^-1",M2);

```

Scilab code Exa 6.18 numerical

```

1 //example 6.18
2
3 clear all;
4 clc;
5
6 //Given:
7 R=82; //Universal gas constant [atm.ml.K^-1.mol^-1]
8 T=298; //Temperature [K]
9 V=250; //volume of water [ml]
10 m2=2.6; //mass of the protein
11 M2=85000; //molar mass of protein [g.mol^-1]

```

```

12
13
14 //To find the osmotic pressure of a solution
15 n2=m2/M2;//no. of moles of protein
16 II=(n2*R*T)/V;//Osmotic pressure of a solution[atm]
17 printf("The osmotic pressure is %f atm ",II);

```

Scilab code Exa 6.19 numerical

```

1 //example 6.19
2
3 clear all;
4 clc;
5
6 //Given:
7 R=8.314;//Universal gas constant [J.K-1.mol-1]
8 Tb=373.15;//Boiling point temperature [K]
9 M1=0.018;// mass of water [kg]
10 Hvap=40.7;//Enthalpy of vaporization [KJ.mol-1]
11
12 //To find the Ebullioscopic constant of water
13 a=R*0.001*Tb2*M1;
14 b=Hvap;
15 Kb=a/b;//Ebullioscopic constant of water [K.Kg.mol-1]
16 printf("The Ebullioscopic constant of water is %f K.
Kg.mol-1",Kb);

```

Scilab code Exa 6.20 numerical

```

1 //example 6.20
2
3 clear all;

```

```

4  clc;
5
6  disp("CaF2(s)<=>CaF2(aq)<=>Ca+2(aq) + 2F-(aq)");
7
8  //Given:
9  Ksp=4*10^-11; //Solubility product of sparingly
    soluble salt CaF2
10
11 //To find the value of activity coefficient
12 x=Ksp/4;
13 Cs=x^0.33; //Solubility
14 y=Cs^2;
15 Y=(x/y)^0.33; //activity coefficient
16 printf("The activity coefficient is %f ",Y);

```

Scilab code Exa 6.21 numerical

```

1  //example 6.21
2
3  clear all;
4  clc;
5
6  //Given:
7  R=8.314; //Universal gas constant [J/K/mol]
8  T=298; //Temperature [K]
9  F=96500; //Faraday's constant
10 Eo=0.98; //Standard e.m.f of the cell [Volts]
11 E=1.16; //e.m.f of the cell [Volts]
12 m=0.01;
13
14 //To find the mean activity coefficient of ZnCl2
    solution
15 a=R*T;
16 b=2*F;
17 x=a/b;

```



```

18 Y=exp((Eo-E-(x*log(4*m*m*m*m)))/(3*x)); //mean activity
    coefficient
19 printf("The mean activity coefficient is %f",Y);

```

Scilab code Exa 6.22 numerical

```

1 //example 6.22
2
3 clear all;
4 clc;
5
6 //Given:
7 M1=0.01; //no. of moles of KCl
8 M2=0.005; //no. of moles of MgCl2
9 M3=0.002; //no. of moles of MgSO4
10 M=0.1; //mass of water [Kg]
11 z11=1;
12 z12=1;
13 z21=2;
14 z22=1;
15 z31=2;
16 z32=2;
17
18 //To find the ionic strength in a solution
19 m1=M1/M; //molality of KCL[m]
20 m2=M2/M; //molality of MgCl2 [m]
21 m3=M3/M; //molality of MgSO4 [m]
22
23 I=0.5*[(m1*z11^2+m1*z12^2+m2*z21^2+2*m2*z22^2+m3*z31
    ^2+m3*z32^2)]; // [mol/Kg]
24
25 printf("The Ionic strength of a solution is %f mol/
    Kg", I);

```

Scilab code Exa 6.23 numerical

```
1 //example 6.23
2
3 clear all;
4 clc;
5
6 //Given:
7 T=298; //Temperature [K]
8 P=1; //pressure [atm]
9 m=0.02; //Ionic strength of HCl solution in CH3OH[mol
    /Kg]
10 E=32.6; //Di-electric constant
11 d=0.787; //Density [gm/cm3]
12
13 //To find the mean activity coefficient
14 I=0.5*(0.02*1*1+0.02*1*1); //Ionic strength of HCl
    solution [mol/Kg]
15 a=I*d;
16 b=(E^3)*(298^3);
17 x=(a/b)^0.5;
18 Y=10^(-1.825*1000000*1*1*x); //mean activity
    coefficient
19 printf("The mean activity coefficient is %f ",Y);
```

Chapter 7

Statistical Thermodynamics

Scilab code Exa 7.1 numerical

```
1 //example 7.1
2
3 clear all;
4 clc;
5
6 //Given:
7 N=20; //no. of particles
8 N1=4; //no. of particles in E1 energy level
9 N2=4; //no. of particles in E2 energy level
10 N3=6; //no. of particles in E3 energy level
11 N4=3; //no. of particles in E4 energy level
12 N5=3; //no. of particles in E5 energy level
13 //To find the number of ways of distributing N
    particles
14 N!=prod(1:N);
15 N1!=prod(1:N1);
16 N2!=prod(1:N2);
17 N3!=prod(1:N3);
18 N4!=prod(1:N4);
19 N5!=prod(1:N5);
20 n=N1!*N2!*N3!*N4!*N5!;
```

```

21 W=N!/n; //no. of ways of distributing
22 disp(W, 'The no. of ways of distributing the
    particles is ');

```

Scilab code Exa 7.2 numerical

```

1 //example 7.2
2
3 clear all;
4 clc;
5
6 //Given:
7 T=298; //Temperature [K]
8 v=6.5*10^13; //Frequency in [sec-1]
9 //Consider zero point energy = 0.
10 h=6.627*10^-34; //planck 's constant [J.s]
11 k=1.381*10^-23; //Boltzmann constant
12 N=1; //Since N=summation(gj*exp(-Ej/kT))
13
14 //To find the fraction of molecules present in
    vibrational level
15 E1=h*v; //for energy level 1[J]
16 E2=2*h*v; //for energy level 2[J]
17 x=k*T;
18 g1=1;
19 g2=1;
20 N1=[g1*exp(-E1/x)]; //molecules present in energy
    level 1
21 N2=[g2*exp(-E2/x)]; //molecules present in energy
    level 2
22 n1=N1/N; //fraction of molecules present in energy
    level 1
23 n2=N2/N; //fraction of molecules present in energy
    level 2
24 printf("The fraction of molecule s present in

```

```

    energy level 1 is %f",n1);
25
26
27 disp(n2,'The fraction of molecules present in energy
    level 2 is ');

```

Scilab code Exa 7.3 numerical

```

1 //example 7.3
2
3 clear all;
4 clc;
5
6 //Given:
7 dE=4.3*10^-20;//difference in energy levels [J]
8 T1=0.000001;//Initial Temperature [K] (approximately
    zero , needed for execution)
9 T2=300;//Final Temperature [K]
10 k=1.381*10^-23;//Boltzmann constant [J/K]
11
12 //To find the ratio of no. of particle per state for
    two states separated by an energy dE
13 x1=k*T1;
14 r1=exp(-dE/x1);
15 x2=k*T2;
16 r2=exp(-dE/x2);
17 disp(r1,'The ratio of no. of particles per state at
    0K is ');
18 disp(r2,'The ratio of no. of particles per state at
    300K is ');

```

Scilab code Exa 7.4 numerical

```

1 //example 7.4
2
3
4 clear all;
5 clc;
6
7 //Given:
8 T1=273; // [K]
9 T2=14273; // [K]
10 E1=-13.6; //Energy of ground state [eV]
11 k=8.617*10^-5; //Boltzmann constant [eV/K]
12 g2=8; //total no. of states with energy E2
13 g1=2; //total no. of states with energy E1
14
15
16 //To find the no. of atoms in first-excited singlet
    state
17 E2=E1/(2^2); //Energy for n=2 (i.e.E2=E1/n2)
18 x1=k*T1;
19 r1=(g2/g1)*exp(-(E2-E1)/x1);
20 x2=k*T2;
21 r2=(g2/g1)*exp(-(E2-E1)/x2);
22 disp(r1, 'The fraction of atoms present in level n=2
    at 273K is ');
23 disp("Therefore total 3*10^25 atoms we say that
    all are present at ground state");
24 printf("\n\nThe fraction of atoms present in level n
    =2 at 14273 is %f\n",r2);
25 x=r2*3*10^25;
26 disp(x, 'Therefore no. of atoms in level n=2 is ');

```

Scilab code Exa 7.5 numerical

```

1 //example 7.5
2

```

```

3 clear all;
4 clc;
5
6 //Given:
7 r1=0.001;//the population of the states at a higher
   energy to that at a lower energy
8 dE=8*10^-20;//The difference in energy [J]
9 k=1.381*10^-23;//Boltzmann constant [J/K]
10
11
12 //To find the Temperature at this condition
13 x=k*log(r1);
14 T=-dE/x;// [K]
15 printf("The Temperature at this condition is %f K",T
   );

```

Scilab code Exa 7.6 numerical

```

1 //example 7.6
2
3 clear all;
4 clc;
5
6
7 //section (1)
8 //Given:
9 //The energy levels are not degenerate
10 w=1;//no. of ways of distributing the molecules
11 k=1.381*10^-23;//Boltzmann constant [J/K]
12 //To find the entropy of the system
13 S1=k*log(w);//Entropy of system at 0K
14 printf("The Entropy of System at 0K and non-
   degenerate eng level is %f J/K/mol",S1);
15
16 //section (2)

```

```

17 //Here the energy levels are degenerate
18 n=2;
19 R=8.314; //Universal gas constant [J/K/mol]
20
21 //To find the entropy of the system
22 //S=klog(n^N)=>S=R*log(n)
23 S2=R*log(n); //Entropy of the system [J/K/mol]
24 printf("\nThe Entropy of system at 0K and degenerate
        eng level is %f J/K/mol",S2);

```

Scilab code Exa 7.9 numerical

```

1 //example 7.9
2
3 clear all;
4 clc;
5
6 //Given:
7 V=0.001; //Volume of vessel [m3]
8 T=300; //Temperature [K]
9 k=1.381*10^-23; //Boltzmann constant [J/K]
10 mol.wt=32; //molecular mass of oxygen molecule
11 h=6.626*10^-34; //planck's constant [J.s}
12
13
14
15 //To find the Transitional partition function of an
        oxygen molecule confined in a 1-litre vessel at
        300K
16 m=32*1.66*(10^-27); //mass of oxygen molecule [Kg]
17 x=[(2*3.14*m*k*T)^(3/2)]*V;
18 y=h^3;
19 zt=x/y; //Transitional partition function of an
        oxygen molecule
20 disp(zt,'The Transitional partition function of an

```


oxygen molecule confined in a 1-litre vessel at
300K is ')

Scilab code Exa 7.12 numerical

```
1 //example 7.12
2
3 clear all;
4 clc;
5
6 //Given:
7 R=1.99; //Universal gas constant [cal/K]
8 e=2.718;
9 V=22414; //volume [cm3]
10 L=6.023*10^23;
11 h=6.626*10^-27; //Planck's constant [erg.sec]
12 m=6.63*10^-23; //mass [gm]
13 k=1.381*10^-16; //Boltzmann constant [erg/K]
14 T=273.2; //Temperature [K]
15
16 //To find the Entropy of argon at 273K and 1
    atmospheric pressure
17 x=V*(e^2.5);
18 y=L*(h^3);
19 z=(2*3.14*m*k*T)^1.5;
20 S=R*log(x*z/y); //Entropy [cal/degree/mol]
21 printf("The Entropy of argon at 273K and 1 atm is %f
    cal/degree/mol",S);
```

Scilab code Exa 7.14 numerical

```
1 //example 7.14
2
```

```

3 clear all;
4 clc
5
6 //Given:
7 T=298; //Temperature [K]
8 I=1.9373*10^-46; //moment of inertia of O2 gas [Kg/m2
    ]
9 h=6.626*10^-34; //Planck 's constant [J.s]
10 k=1.381*10^-23; //Boltzmann constant [J/K]
11 R=8.314; //Universal gas constant [J/K/mol]
12 u=2; //Homonuclear diatomic molecule
13
14
15 //To find the rotational entropy and free energy for
    O2 gas
16 Sr=R+R*log(8*3.14*3.14*I*k*T/(u*h*h)); // [J/K/mol]
17 Gr=-R*0.001*T*log(8*3.14*3.14*I*k*T/(u*h*h)); // [KJ/
    mol]
18 printf("The rotational entropy for O2 gas is %f J/K/
    mol",Sr);
19 printf("\nThe rotational free energy for O2 gas is
    %f KJ/mol",Gr);

```

Scilab code Exa 7.15 numerical

```

1 //example 7.15
2
3 clear all;
4 clc;
5
6 //Given:
7 T=298; //Temperature [K]
8 v=892.1*3*10^10; //frequency [sec -1]
9 h=6.626*10^-27; //Planck 's constant [J.s]
10 k=1.381*10^-16; //Boltzmann constant [erg/K]

```

```

11 e=2.718;
12 R=1.998; // Universal gas constant [cal/K]
13 //To find the vibrational contribution to the
    entropy of F2 at 298K
14 x=h*v/(k*T);
15 a=R*x*e^-x/(1-e^-x); //a=E-Eo/T
16 b=R*log(1-e^-x); //b=G-Eo/T
17 S=a-b; // [cal/deg]
18 printf("The vibrational contribution to the entropy
    of F2 is %f cal/deg",S);

```

Scilab code Exa 7.16 numerical

```

1 //example 7.16
2
3 clear all;
4 clc;
5
6 //Given:
7 T=1273; //Temperature [K]
8 h=6.26*10^-27; //Planck's constant [J.s]
9 k=1.381*10^-16; //Boltzmann constant [erg/K]
10 T=1000; //Temperature [degrees]
11 m=3.82*10^-23; //mass of Na [gm]
12 I=(1.91*10^-23)*(3.078*10^-8)^2; //moment of inertia [
    gm.cm2]
13 dE=0.73*1.602*10^-12; // [erg]
14 v=159.23*(3*10^10); //frequency [s-1]
15 R=82; //universal gas constant [cm3.atm/deg]
16 u=2; //symmetry number
17 L=6.023*10^23; //avogadro's number
18
19 //To find the equilibrium constant
20 p=((3.14*m*k*T)^1.5)/h/h/h;
21 s=R*u*h*h/L/8/3.14/3.14/I/k;

```

```

22 q=1-(exp(-h*v/k/T));
23 r=exp(-dE/k/T);
24 Kp=p*s*q*r;//Equilibrium constant
25 printf("The equilibrium constant is %f",Kp);

```

Scilab code Exa 7.17 numerical

```

1 //example 7.17
2
3 clear all;
4 clc;
5
6 //Given:
7 T=298;//Temperature [K]
8 m1=32;
9 m2=36;
10 m3=34;
11 u1=8;
12 u2=9;
13 u3=16*18/34;
14 z1=0.99924;
15 z2=0.99951;
16 z3=0.99940;
17 h=6.26*10^-27;//Planck's constant [J.s]
18 c=3*10^10;//Speed of light [m/s]
19 k=1.38*10^-16;//Boltzman's constant [erg/K]
20 vo1=1535.8;//vibration frequency of 16O18O [cm-1]
21 vo2=1580.4;//vibration frequency of 16O2 [cm-1]
22 vo3=1490.0;//vibration frequency of 18O2 [cm-1]
23 dE=0.5*h*c*(2*vo1-vo2-vo3);// [erg]
24 r=dE/k/T;
25
26
27 //To find the equilibrium constant for isotopic
    exchange reaction

```

```
28 a=m3^3/m2^1.5/m1^1.5;
29 b=(u3^2)*4/u2/u1;
30 c=z3^2/z2/z1;
31 Kp=a*b*c*exp(-r);
32 printf("The value of equilibrium constant for
        isotopic exchange reaction is %f",Kp);
```
