Scilab Textbook Companion for Basic Principles And Calculations In Chemical Engineering

by D. M. Himmelblau And J. B. Riggs[1]

Created by
Amarjeet Kumar Singh
B.Tech
Chemical Engineering
IT BHU
College Teacher
Prakash Kotecha
Cross-Checked by

July 31, 2019

---

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

Book Description

Title: Basic Principles And Calculations In Chemical Engineering

Author: D. M. Himmelblau And J. B. Riggs

Publisher: PHI Learning Private Limited, New Delhi

Edition: 7

Year: 2004

ISBN: 978-81-203-3839-5

Scilab numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

AP Appendix to Example(Scilab Code that is an Appednix to a particular Example of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means a scilab code whose theory is explained in Section 2.3 of the book.

Contents

List of Scilab Codes

Chapter 1

Dimensions units and their conversion

Scilab code Exa 1.1 `Dimensions and Units`

```
clear;
clc;

// Example 1.1
printf('Example 1.1\n\n');
//Page no. 13
// Solution

//(a)
printf('(a) We cannot add the two terms since both
   have different dimensions. 1 foot has the
   dimension of the length , whereas 3 seconds has
   the dimension of time.\n');

//(b)
// Converting all terms to same unit
hp = 746;//[watts]
total = 1*hp+300;//[watts]

```

```
printf(' (b) Answer is  %i watts.',total);
```

Scilab code Exa 1.2 Conversion of Units

```
clear ;
clc;

// Example 1.2
printf('Example 1.2\n\n');
//Page no. 17
// Solution

// (a)
// Converting all terms to same unit
ml = 1.61;//[km]
km = (2*1)/(ml);//[miles]
printf('(a) 2 kilometers is equal to %.2f miles.\n',
   km);

//(b)
in = 2.54;//[cm]
dy = 24*60;//[min]
nw_unit = (400*(in)^3*1)/(1*dy);//[cubic centimetre/
   min]
printf(' (b) 400 cubic in./day is equal to %.2f
   cubic centimetre/min.',nw_unit);
```

Scilab code Exa 1.3 Nanotechnology

```
clear;
clc;

// Example 1.3
```

```
printf('Example 1.3\n\n');
//Page no. 17
// Solution

//(a)
// Converting all terms to same unit
nm = 10^(-9);//[meters]
m1 = 10;//[decimeters]
dm = (1.8*nm*m1)/(1*1);//[decimeters]
printf('(a) 1.8 nanometers is equal to %.2e dm.\n',
   dm);

//(b)
m2 = 39.37;//[inches]
in = (1.8*nm*m2)/(1*1);//[inches]
printf(' (b) 1.8 nanometers is equal to %.2e in.\n',
   in);
```

Scilab code Exa 1.4 `A conversion involving lbm and lbf`

```
clear ;
clc;

// Example 1.4
printf('Example 1.4\n\n');
//Page no. 19
// Solution

// Potential Energy = mgh
// Assume 100 lb means 100 lb mass
m = 100;//[lb]
g = 32.2 ;//[ft/second square]
h = 10 ;//[ft]
gc = 32.174 ;//[(ft*lbm)/(second square/lbf)]
pe = (m*(g/gc)*h) ;//[ft*lbf]
```

```
printf('Potential Energy  is equal to %i (ft)(lbf).\n',pe);
```

Scilab code Exa 1.5 Conversion associated with Biological Materials

```
clear;
clc;

// Example 1.5
printf('Example 1.5\n\n');
//Page no.20
// Solution

// Basis 1 min
// Assume 100lb means 100 lb mass
g = 10^6 ;//[ug mol]
lb = 454 ;//[g mol]
ml = .001 ;//[L]
L = 3.531*10^(-2);//[ft^3]
hr = 60 ;//[min]
dy = 24 ;//[hr]
pr_rate = (0.6*1*1*1*hr*dy/(g*lb*ml*L)); //[ft*lbf]

printf('Production rate of glucose is %.4f lb mol/(cubic feet*day).\n',pr_rate);
```

Scilab code Exa 1.6 Dimensional Consistency

```
clear;
clc;

// Example 1.6
printf('Example 1.6\n\n');
```

```
//Page no. 22
// Solution

// using suitable conversion factors inside front cover of book
printf('By analysing dimensionally both sides of equation you can say that both values of 16.2 must have the units of microns(10^-6 m).\n');
printf(' The exponential must be dimensionless so that 0.021 must have the associated units of s^(-1).\n');

m = 39.27 ;//[inches]
um = 10^(-6) ;//[meters]
c1 = 16.2*m*um ;//[inches]
mn = 60 ;//[seconds]
c2 = 0.021*60;//[min^(-1)]
printf('\n New modified expression so that we can put t in minutes and get d in inches is as follows,\n d(in) = %.2e(1-e^(-%.2f*t(min))) \n',c1,c2);
```

Scilab code Exa 1.7 `Dimensional Consistency`

```
clear;
clc;

// Example 1.7
printf('Example 1.7\n\n');
// Page no. 23
// Solution

printf('By analysing dimensionally , both a and x have same units(from 1+(x^2/a^2)),thus left hand side of equation has units of 1/x(from d/dx),\
```

```
nAnd right hand side has units of x^2 (from product a.x).\n');
printf('\nTherefore something is wrong as the equation is not dimensionally consistent.\n');
```

Scilab code Exa 1.8 `Retention of significant Figures`

```
clear ;
clc;

// Example 1.8
printf('Example 1.8\n\n');
// Page no. 28
// Solution

// Using Scientifc notation
x = 2.24 * 10^4 ; //[kg]
y = 2.01 * 10^4 ;//[kg]
D = x - y ;// Difference obtained by using scientific notation //[kg]

printf(' Difference obtained by using scientific notation is %.4e kg.\n  Hence answer is good to 2 decimal places. \n',D);
```

Scilab code Exa 1.9 `Micro dissection of DNA`

```
clear;
clc;

// Example 1.9
printf('Example 1.9\n\n');
//Page no. 29
```

```
// Solution

um = 3 ;//[kb]
kb = 1000 ;//[bp]
bs_prs = (3*um*kb)/(1*1);
printf('The number of base pairs are %i bp. \n', bs_prs);
```

Chapter 2

Moles Density and Concentration

Scilab code Exa 2.1 `Calculation of Molecular Weight`

```
clear ;
clc;

// Example 2.1
printf('Example 2.1\n\n');
//Page no. 45
// Solution

// Count the number of each element from fig. E2.1.
// Look for the atomic weights of elements from
   Appendix B
// Assume the one cell is a molecule
n_Ba = 2 ;// Number of atoms of Ba
n_Cu = 16 ;// Number of atoms of Cu
n_O = 24 ;// Number of atoms of O
n_Y = 1 ;// Number of atoms of Y
m_Ba = 137.34 ;//Atomic wt. -[g]
m_Cu = 63.546 ;//Atomic wt.-[g]
m_O = 16.00 ;//Atomic wt.-[g]
```

```
m_Y = 88.905; //Atomic wt.-[g]
mol_wt =  n_Ba*m_Ba + n_Cu*m_Cu + n_O*m_O + n_Y*m_Y ;//The molecular weight of given material-[g]

printf('The molecular weight of given material is %1.1f g/g mol.\n',mol_wt);
```

Scilab code Exa 2.2 `Use of Molecular Weight to Convert Mass to Moles`

```
clear ;
clc;

// Example 2.2
printf('Example 2.2\n\n');
// Page no. 46
// Solution

//(a)
m_NaOH = 40.0 ;//[lb]
pnd_mol = 2*1/m_NaOH ;//[lb mol]
printf('(a) The number of pound moles of NaOH in 2.00 lb NaOH is %.2f lb mol.\n',pnd_mol);

//(b)
grm_mol = pnd_mol*454 ;//[g mol]
printf(' (b) The number of gram moles of NaOH in 2.00 lb NaOH is %.2f g mol.\n',grm_mol);
```

Scilab code Exa 2.3 `Use of Molecular Weight to Convert Moles to Mass`

```
clear;
clc;

```

```
// Example 2.3
printf('Example 2.3\n\n');
//Page no. 46
// Solution

//Basis 7.5 g mol of NaOH
m_NaOH = 40.0 ;//[lb]
lb = 454 ;//[g mol]
n = 7.50*1*m_NaOH/(lb*1);
printf('Number of pounds of NaOH is %.3f lb.\n',n);
```

Scilab code Exa 2.4 `Calculation of Density`

```
clear ;
clc;

// Example 2.4
printf('Example 2.4\n\n');
//Page no. 53
// Solution

// (a)
d_w1=1000 ;//[kg/cubic metre]

d1=(1.184*d_w1*1000)/(10^6) ;//[g/cubic centimetre]
printf('(a) Density in g/cubic centimetre is %.4f g/cubic centimetre.\n',d1);

// (b)
d_w2= 62.4 ;//[lbm/cubic feet]

d2=1.184*d_w2/1 ;//[lbm/cubic feet]

printf(' (b) Density in lbm/cubic feet is %.1f lbm/cubic feet.\n',d2);
```

```

// (c)
d3=1.184*d_w1 ;//[kg/cubic metre]

printf(' (c) Density in kg/cubic metre is %.1f kg/cubic metre.\n',d3);
```

Scilab code Exa 2.5 `Calculation of Mass and Moles`

```
clear ;
clc;

// Example 2.5
printf('Example 2.5\n\n');
//Page no. 54
// Solution

m_wt=192 ;//[kg]
d_sol=1.024*1000 ;//[kg/cubic metre]
// 1000L=1 cubic metre
c_sol=d_sol/1000 ;//[kg/L]
c_drug=c_sol*.412 ;//[kg/L]
printf('Concentration of drug in solution is %.3f kg/L .\n',c_drug);

Q=10.5 ;//[L/min]
Qmol=10.5*c_drug/m_wt ;//[kg mol/min]
printf(' Flow rate of drug is %.3f kg mol/min. \n',Qmol);
```

Scilab code Exa 2.6 `Conversion between Mass Fraction and Mole Fraction`

```
clear ;
```

```
clc;

// Example 2.6
printf('Example 2.6\n\n');
//Page no.57
// Solution

// Let component 1 be water and component 2 be NaOH
// Basis 10 kg total solution
m1 = 5.0 ;//[kg]
m2 = 5.0; //[kg]
total = m1  + m2 ;//[kg]
m_fr1 = m1/total ;//mass fraction of water
m_fr2 = m2/total ;//mass fraction of NaOH
mw1 = 18.0 ;//molecular weight of water
mw2 = 40.0 ;//molecular weight of NaOH
mol1 = m1/mw1;
mol2 = m2/mw2;
mol_fr1 = mol1/(mol1  + mol2) ;//mol fraction of water
mol_fr2 = mol2/(mol1  + mol2) ;//mol fraction of NaOH
printf(' Component     kg          Mass fraction      Mol.Wt.      kg mol     Mole fraction\n');
printf('n Water          %.2f        %.3f                %.1f         %.3f        %.2f\n',m1,m_fr1,mw1,mol1,mol_fr1);
printf(' NaOH           %.2f        %.3f                %.1f         %.3f        %.2f\n',m2,m_fr2,mw2,mol2,mol_fr2);
printf(' Total          %.2f       %.3f                          %.3f        %.2f',m1  + m2,m_fr1  + m_fr2,mol1  + mol2,mol_fr1  + mol_fr2);
```

Scilab code Exa 2.7 `Nitrogen Requirements for the Growth of Cells`

```
clear ;
clc;

// Example 2.7
printf('Example 2.7\n\n');
//Page no.58
// Solution

// Basis 500 L solution containing 35g/L
// (NH4)2SO4 is the only nitrogen source
cn = 35 ;//[g/L]
wt = 9  ;//[wt % N]
m_wt1 = 132 ;//[g]
m_wt2 = 14 ;//[g]
amt = (500*(35)*.09*1*1*m_wt1)/(1*m_wt2*1*1);
printf('Total amount of (NH4)2SO4 consumed  is  %.1f
    g.',amt);
```

Scilab code Exa 2.8 Use of ppm

```
clear ;
clc;

// Example 2.8
printf('Example 2.8\n\n');
//Page no. 63
// Solution

// 1 kg of the air/HCN mixture
// (a)
m1 = 27.03 ;//[g]
m2 = 29.0 ;//[g]
cn = (10*m1*1000*1000)/(10^6*m2) ;//[mgHCN/kg air]
printf('(a) 10.0 ppm HCN is %.2f mg HCN/kg air.\n',
   cn);
```

```

// (b)
ld = 300 ;//[mg/kg air]
fr = cn/ld;
printf(' (b) Fraction of lethal dose is 10.0 ppm is %.3f.',fr);
```

Scilab code Exa 2.9 `Calculation of Mole Fraction and ppm from a Concentration`

```
clear ;
clc;

// Example 2.9
printf('Example 2.9\n\n');
//Page no. 64
// Solution

// Let component 1 be water and component 2 be HNO3
// Basis 1L solution
c = 15 ;//[g/L]
sg = 1.10 ;
L = 1000 ;//[cubic centimetre]
m1 = 18.0 ;//[g]
m2 = 63.02 ;//[g]
cn2 = (15*1)/(L*sg) ;//[gHNO3/g soln]
// Basis 1g soln
cn1 = 1-cn2 ;// Mass of water in 1 g soln
mg1 = cn1/m1;
mg2 = cn2/m2;
ml_fr1 = mg1/(mg1+mg2);
ml_fr2 = mg2/(mg1+mg2);
printf(' (a) Component   g(per 1g soln)   Mol.Wt.    g mol              Mole fraction\n')
printf('     Water       %.4f                                          %.2f
```

```
          %.3f                    %.2f\n',cn1,m1,mg1,ml_fr1);
printf('      HNO3          %.4f                          %.2f          %e          %e\n',cn2,m2,mg2,ml_fr2);
// (b)
cpm = cn2*10^6 ;//[ppm]
printf('\n (b)Ppm of HNO3 in soln. is %.2f ppm.',cpm);
```

Scilab code Exa 2.10 `Evaluation of Alternate Processes for the Production of Methy`

```
clear;
clc;

// Example 2.10
// Page no. 64
// Solution

// Given
// Process a
// Let us take array of given values for compounds in following order 1- acetone, 2 - Hydrogen cyanide, 3- methanol, 4-Sulphuric acid , 5 - Methyl methacrylate
Lb1  =  [0.68,0.32,0.37,1.63,1]  ;// Mass of compounds -[lb]
Value1  =  [0.43, 0.67,0.064,0.04,0.78] ;// Cost of compounds -[$/lb]
TLV1  =  [750,10,200,2,100] ;// TLV value of compounds -[ppm]
OITF1  =  [0,1000,10,10000,10] ;// Note : (?) mark values are neglected as they are nearly equal to zero

// Process b
```

```
// Let us take array of given values for compounds in following order 1- Isobutylene, 2 - Methanol, 3- Pentane, 4-Sulphuric acid , 5 - Methyl methacrylate
Lb2 = [1.12,0.38,0.03,0.01,1.00] ;// Mass of compounds -[lb]
Value2 = [0.31,0.64,0.112,0.04,0.78] ;// Cost of compounds -[$/lb]
TLV2 = [200,200,600,2,100] ;// TLV value of compounds -[ppm]
OITF2 = [0,10,0,10000,10] ;// Note : (?) mark values are neglected as they are nearly equal to zero

NetV1 = Lb1(5)*Value1(5) - Lb1(2)*Value1(2) - Lb1(3)*Value1(3) - Lb1(4)*Value1(4) - Lb1(1)*Value1(1); // Net Value for process (a) -[$]
NetV2 = Lb2(5)*Value2(5) - Lb2(2)*Value2(2) - Lb2(3)*Value2(3) - Lb2(4)*Value2(4) - Lb2(1)*Value2(1) ;// Net Value for process (b) -[$]

printf('1.With respect to cost criteria\n');
printf(' Net value for process (a) is %.2f $ and for process (b) is %.2f $.\n Hence based on net value both process are equivalent. \n',NetV1,NetV2);

// With respect to two environmental criteria
TLV_index1 = Lb1(1)/TLV1(1) + Lb1(2)/TLV1(2) + Lb1(3)/TLV1(3) + Lb1(4)/TLV1(4) + Lb1(5)/TLV1(5) ;// TLV index for process a
OITF_index1 = OITF1(1)*Lb1(1) +OITF1(2)*Lb1(2) + OITF1(3)*Lb1(3) + OITF1(4)*Lb1(4) + OITF1(5)*Lb1(5) ;// OITF index process a

TLV_index2 = Lb2(1)/TLV2(1) + Lb2(2)/TLV2(2) + Lb2(3)/TLV2(3) + Lb2(4)/TLV2(4) + Lb2(5)/TLV2(5) ;// TLV index for process b
```

```
OITF_index2 = OITF2(1)*Lb2(1) +OITF2(2)*Lb2(2) + OITF2(3)*Lb2(3) + OITF2(4)*Lb2(4) + OITF2(5)*Lb2(5) ;// OITF index process b

printf('\n 2.With respect to  two environmental criteria\n');
printf('   Process (a)\n   TLV index for process a is %.2f .\n   OITF index process a is %.2f .  \n',TLV_index1,OITF_index1);
printf('   \n   Process (b)\n   TLV index for process b is %.2f .\n   OITF index process b is %.2f .  \n',TLV_index2,OITF_index2);
```

Chapter 3

Choosing a Basis

Scilab code Exa 3.1 `Choosing a Basis`

```
clear ;
clc;

// Example 3.1
printf('Example 3.1\n\n');
//Page no. 79
// Solution

// Let component 1 be Ce and component 2 be O
// Basis 2kg mol CeO
mol1 = 1.0 ;//[kg mol]
mol2 = 1.0 ;//[kg mol]
total = mol1+mol2 ;//[kg mol]
mol_fr1 = mol1/total ;//mole fraction of Ce
mol_fr2 = mol2/total ;//mole fraction of O
mw1 = 140.12; //molecular weight of Ce
mw2 = 16.0  ;//molecular weight of O
m1 = mw1*mol1;
m2 = mw2*mol2;
m_fr1 = m1/(m1+m2) ;//mass fraction of Ce
m_fr2 = m2/(m1+m2) ;//mass fraction of O
```

```

printf('Component      kg mol        Mole fraction      Mol.Wt.       kg.          Mass fraction\n')
printf('\n Ce                %.2f           %.3f              %.2f        %.3f        %.2f\n',mol1,mol_fr1,mw1,m1,m_fr1);
printf(' O                %.2f           %.3f         %.2f           %.3f          %.2f\n',mol2,mol_fr2,mw2,m2,m_fr2);
printf(' Total             %.2f           %.3f         %.2f        %.3f        %.2f',mol1+mol2,mol_fr1+mol_fr2,mw1+mw2,m1+m2,m_fr1+m_fr2);
```

Scilab code Exa 3.2 `Choosing a Basis`

```
clear ;
clc;

// Example 3.2
printf('Example 3.2\n\n');
//Page no. 80
// Solution

// Basis 100kg mol gas
ml1 = 20.0 ;//[kg mol]
ml2 = 30.0 ;//[kg mol]
ml3 = 40.0 ;//[kg mol]
ml4 = 10.0 ;//[kg mol]
mw1 = 44.0  ;//molecular weight of CO2
mw2 =  28.0  ;//molecular weight of CO
mw3 =  16.04 ; //molecular weight of CH4
mw4 =  2.02  ;//molecular weight of H2
m1 = mw1*ml1;
m2 = mw2*ml2;
m3 = mw3*ml3;
```

```
m4 = mw4*ml4;
printf(' Component        kg mol          Mol.Wt.      kg.
         \n')
printf(' CO2              %.2f            %.2f         %.0f
       \n',ml1,mw1,m1);
printf(' CO               %.2f            %.2f         %.0f
       \n',ml2,mw2,m2);
printf(' CH4              %.2f            %.2f         %.0f
       \n',ml3,mw3,m3);
printf(' H2               %.2f            %.2f          %.0f
        \n',ml4,mw4,m4);
printf('\n Total            %.2f            %.2f         %.0f
          \n',ml1+ml2+ml3+ml4,mw1+mw2+mw3+mw4,m1+m2+
   m3+m4);
av_m = (m1+m2+m3+m4)/100 ;//[kg]
printf('\nAverage molecular mass of gas is %.1f kg.\
   n',av_m);
```

Scilab code Exa 3.3 Choosing a Basis for Cell Growth

```
clear ;
clc;

// Example 3.3
printf('Example 3.3\n\n');
//Page no. 81
// Solution

// Basis 1 hour
rc = 5000 ;//[cpm-counts per minute]
cg = 10000/24 ;//[cells/hr]
k = cg/rc ;//[cells/cpm]
n_rc = 8000 ;//[cpm]
n_cg = k*n_rc ;//[cells/hr]
printf('New average cell growth rate is %.0f cells/
```

```
hr.\n',n_cg);
in_p = ((n_cg-cg)/cg)*100 ;//[increase percent]
printf(' Increase percent of cell growth rate is %.1f %% .\n',in_p);
```

Scilab code Exa 3.4 Calculation of the Mass Fraction of the Components in Nanopart

```
clear;
clc;

// Example 3.4
printf('Example 3.4\n\n');
//Page no. 82
// Solution

// Basis 100 g mol of Nd(4.5)Fe(77)B(18.5)
//(a)
n_Fe = 77-0.2;
printf('(a) Molecular formula after adding Cu is Nd(4.5)Fe(%.1f)B(18.5)Cu(.2).\n',n_Fe);

//(b)
o_ml1 = 4.5 ;//[kg mol]
o_ml2 = 77.0 ;//[kg mol]
o_ml3 = 18.5 ;//[kg mol]
o_ml4 = 0.0 ;//[kg mol]
f_ml1 = 4.5 ;//[kg mol]
f_ml2 = 77.0-0.2 ;//[kg mol]
f_ml3 = 18.5 ;//[kg mol]
f_ml4 = 0.2 ;//[kg mol]
mw1 = 144.24  ;//molecular weight of Nd
mw2 =  55.85  ;//molecular weight of Fe
mw3 =  10.81 ; //molecular weight of B
mw4 =  63.55  ;//molecular weight of Cu
m1 = mw1*f_ml1;
```

```
m2 = mw2*f_ml2;
m3 = mw3*f_ml3;
m4 = mw4*f_ml4;
f1 = f_ml1/100;
f2 = f_ml2/100;
f3 = f_ml3/100;
f4 = f_ml4/100;
tf = f1+f2+f3+f4;
printf('\n (b) Component      Original g mol      Final g mol        Mol.Wt.          g.              Mass fraction\n')
printf('     Nd              %.2f               %.2f               %.2f         %.2f          %.3f\n',o_ml1,f_ml1,mw1,m1,f1);
printf('     Fe              %.2f              %.2f              %.2f         %.2f          %.3f\n',o_ml2,f_ml2,mw2,m2,f2);
printf('     B               %.2f              %.2f              %.2f         %.2f          %.3f\n',o_ml3,f_ml3,mw3,m3,f3);
printf('     Cu              %.2f               %.2f               %.2f          %.2f          %.3f\n',o_ml4,f_ml4,mw4,m4,f4);
printf('\n      Total          100.0              100.0                          %.2f         %.3f\n',m1+m2+m3+m4,tf);
```

Scilab code Exa 3.5 `Changing Bases`

```
clear ;
clc;

// Example 3.5
printf('Example 3.5\n\n');
//Page no. 84
```

```
// Solution

// Basis 100 kg coal
ml_r = 9;
wt_r = (9*1.008)/(1*12) ;//conversion of mole ratio to wt.ratio
m1 = 2 ;//[kg] wt.of sulphur
m2 = 1 ;//[kg] wt. of nitrogen
m3 = 6 ;//[kg] wt. of oxygen
m4 = 11 ;//[kg] wt. of ash
m5 = 3 ;//[kg] wt. of water
m6 = (1*77)/(wt_r+1) ;//[kg] wt. of carbon
m7  = wt_r*m6 ;//[kg] wt. of hydrogen
wc = 100-(m4+m5)  ;//[kg] wt. of coal excluding ash and water
wf1 = m1/wc;
wf2 = m2/wc;
wf3 = m3/wc;
wf4 = m4/wc;
wf6 = m6/wc;
wf7 = m7/wc;
tf = wf1+wf2+wf3+wf6+wf7;
printf(' Component        kg.              Mass fraction');
printf('\n C               %.2f            %.2f\n',m6,wf6);
printf(' H               %.2f            %.2f\n',m7,wf7);
printf(' S               %.2f             %.2f\n',m1,wf1);
printf(' N               %.2f             %.2f\n',m2,wf2);
printf(' O               %.2f             %.2f\n',m3,wf3);
printf('\n Total           %.2f            %.2f\n',wc,tf);
```

Chapter 4

Temperature

Scilab code Exa 4.1 `Temperature Conversion`

```
clear ;
clc;

// Example 4.1
printf('Example 4.1\n\n');
//Page no. 92
// Solution

//(a)
Temp_c=100 ;//[degree Celsius]
Temp_k=Temp_c+273 ;//[K]
printf('(a) Temperature in kelvin is %.2f K\n',
   Temp_k);

//(b)
Temp_f=(100*(1.8/1)) +32 ;//[degree Fahrenheit]
printf(' (b) Temperature in degree Fahrenheit is %.2
   f \n',Temp_f);

//(c)
Temp_r= Temp_f + 460 ;//[degree Rankine ]
```

```
printf(' (c) Temperature in degree Rankine is %.2f ',Temp_r);
```

Scilab code Exa 4.2 Temperature Conversion

```
clear ;
clc;

// Example 4.2
printf('Example 4.2\n\n');
// Page no. 93
// Solution

// Given
// Heat capacity = 139.1 + (1.56*10^-1)Tc J/(g mol*degree C), T is in degree C
// First convert Tc (Temperature in degree celsius) to TR (in degree R) to get c + dTR, where
c = 139.1 + (1.56*10^-1)*(-460-32)/1.8 ;
d = (1.56*10^-1)/1.8;

//Now convert c +dTR to (Btu/lb mol*degree R) to get answer of form a + bTR, where
a = c*(454/(1055*1.8)) ;
b = d*(454/(1055*1.8)) ;

printf('The required answer is %.2f + (%.2e)T Btu/(lb mol*degree R) , where T is in degree R .  \n',a,b);
```

Chapter 5

Pressure

Scilab code Exa 5.1 `Pressure Conversion`

```
clear ;
clc;
// Example 5.1
printf('Example 5.1\n\n');
//Page no.109
// Solution

P = 60 ;//[Gpa]

//(a)
p_atm = (P*(10^6))/101.3 ;//[atm]
printf('(a) Pressure in atmospheres is %.2e atm\n',p_atm);

//(b)
p_s = (P*(10^6)*14.696)/101.3 ;//[psia]
printf(' (b) Pressure in psia is %.2e psia\n',p_s);

// (c)
p_in = (P*(10^6)*29.92)/101.3 ;//[inches of Hg]
printf(' (c) Pressure in inches of Hg is %.2e in. Hg
```

```
   \n',p_in);

// (d)
p_mm = (P*(10^6)*760)/101.3 ;//[mm of Hg]
printf(' (d) Pressure in mm of Hg is %.2e mm Hg\n',
   p_mm);
```

Scilab code Exa 5.2 `Pressure Conversion`

```
clear;
clc;
// Example 5.2
printf('Example 5.2\n\n');
//Page no. 110
// Solution

b_rd = 28.0 ;//[in. Hg]
p_rd = 51.0 ;//[psia]
p_atm = b_rd*14.7/29.92 ;//[psia]
p_tnk  =  p_atm+p_rd ;//[psia]
printf(' Pressure in tank in psia is %.1f psia\n',
   p_tnk);
```

Scilab code Exa 5.3 `Vacuum Pressure Reading`

```
clear ;
clc;
// Example 5.3
printf('Example 5.3\n\n');
//Page no. 111
// Solution

b_rd = 100.0 ;//[kPa]
```

```
gp =  64.5*101.3/76.0 ;//[kPa]
p_tnk = b_rd-gp ;//[kPa]
printf(' Absolute Pressure in tank in is %.1f kPa\n',p_tnk);
printf('  Since absolute pressure in tank(%.1f kPa) is less than 20 kPa , the mice probably will not survive. \n',p_tnk);
```

Scilab code Exa 5.4 `Calculation of Pressure Difference`

```
clear ;
clc;
// Example 5.4
printf('Example 5.4\n\n');
//Page no. 115
// Solution

df = 1.10*10^3 ;//[kg/m^3]
d = 1.0*10^3 ;//[kg/m^3]
g = 9.8 ;//[m/s^2]
h = 22.0 ;//[mm]
dP = (df-d)*g*(h*10^(-3)) ;//[Pa]
printf('Pressure difference across the orifice plate is %.1f Pa.\n',dP);
```

Scilab code Exa 5.5 `Pressure Conversion`

```
clear ;
clc;
// Example 5.5
printf('Example 5.5\n\n');
//Page no. 117
// Solution
```

```

p_atm=730.0*29.92/760.0 ;//[in. Hg]
gp= (4.0*29.92)/(2.54*12*33.91) ;//[in. Hg]
p_air=p_atm-gp ;//[in. Hg]
printf(' Pressure of the air is %.1f in. Hg.\n',
   p_air);
```

Chapter 6

Introduction to Material Balances

Scilab code Exa 6.1 `A Material Balance for the blending of Gasoline`

```
clear ;
clc;
// Example 6.1
printf('Example 6.1\n\n');
// Page no. 142
// Solution

// Given
P_O = 89 ;// Premium octane -[octane/gal]
S_O = 93 ;// Supereme octane - [octane/gal]
R_O = 87 ;// Regular octane - [octane/gal]
CP = 1.269 ;// Cost of premium octane -[$/gal]
SP = 1.349 ;// Cost of supereme octane -[$/gal]
RP = 1.149 ;// Cost of regular octane -[$/gal]

// Let x and y fraction of regular octane and
   supreme octane is blended respectively,therefore:
    x + y = 1 ...(a)
// and 89 = 87x + 93y ...(b)
```

```
// Solve equations (a) and (b) simultaneously
a = [1 1;87 93] ;// Matrix of coefficients of unknown
b = [1;89] ;// Matrix of constant
c = a\b ;// Matrix of solutons- x = c(1) , y = c(2)
cost =  c(1)*RP + c(2)*SP ;// Cost after blending - [$/gal]
sv = CP - cost ;// Save on blending - [$/gal]

// Check whether there is loss or save
if (sv<0)
   then
    printf('We will not save money by blending.');

  else
      printf('We will save money by blending, and save is %.3f $/gal.',sv);
```

Scilab code Exa 6.2 Concentration of cells using a Centrifuge

```
clear ;
clc;
// Example 6.2
printf('Example 6.2\n\n');
//Page no. 147
// Solution

// Basis 1 hour
fd= 1000.0 ;//feed rate-[L/hr]
cfd= 500.0;//Weight of cells per litre- [mg/L]
dn= 1.0 ;//Density of feed-[g/cm^3]
wp= 50.0 ;// Weight percent of cells in product stream
Pg=(fd*cfd*dn)/(1000*wp*.01) ;// Mass balance for cells
```

```
printf(' Product flow(P) per hour is %.1f g\n',Pg);
Dg= (fd*dn*1000) - Pg*(wp*.01) ;// Mass balance for the fluid
printf('  Discharge flow per hour is %.3e g\n',Dg);
```

Scilab code Exa 6.3 `Discharge of Tank Residuals to the Environment`

```
clear ;
clc;
// Example 6.3
printf('Example 6.3\n\n');
//Page no. 154
// Solution

//Basis 10000 gal motor oil at an assumed 77 degree fahrenheit
dn =  0.80 ;//Density of motor oil -[g/cm^3]
in_ms = (10000*(0.1337)*62.4*dn) ;// Initial mass of motor oil in the tank -[lb]
printf(' Initial mass of motor oil in the tank is %.1f lb\n',in_ms);
m_fr = .0015 ;//Mass fractional loss
printf('  Mass fractional loss is %.4f \n',m_fr);
Dsg = m_fr*in_ms ;// Mass balance for the fluid
printf('  Discharge of motor oil on flushing flow for 10000 gal motor oil is %.1f lb\n',Dsg);
```

Chapter 7

A General Strategy for Solving Material Balance Problems

Scilab code Exa 7.1 `Understanding the Problem`

```
clear ;
clc;
// Example 7.1
printf('Example 7.1\n\n');
//Page no.169
// Solution

v_ts = 105.0 ;// velocity of train wrt station -[cm/s]
v_mt = 30.0 ;// velocity of man wrt train -[cm/s]
v_hm = 2.0 ;// velocity of hot dough wrt man-[cm/s]
v_am = 1.0 ;// velocity of ant wrt man- [cm/s]
// By careful reading of problem you can see that ant is moving away from man's mouth at 1 cm/s , so ant's velocity wrt station is say v_as
v_as  =  v_ts + v_mt + v_am;
printf(' The ant is moving towards station at the rate of %.1f cm/s.\n',v_as);
```

Scilab code Exa 7.2 `Drawing a Sketch of a Mixing Process`

```
clear ;
clc;
// Example 7.2
printf('Example 7.2\n\n');
// Page no. 169
// Solution Fig. E7.2

printf("Drawing as in fig E7.2 is not possible with
    scilab.")
```

Scilab code Exa 7.3 `Placing the unknown Information on the Diagram`

```
clear ;
clc;
// Example 7.3
printf('Example 7.3\n\n');
// Page no. 171
// Solution Fig. E7.3

printf("Drawing as in fig E7.3 is not possible with
    scilab.")
```

Scilab code Exa 7.4 `Analyisis of the Degrees of Freedom`

```
clear ;
clc;
// Example 7.4
```

```
printf('Example 7.4\n\n');
//Page no. 180
// Solution

n_un= 7 ;// Number of unknowns in the given problem- 3 values of xi and 4 values Fi
n_ie = 5 ;// Number of independent equations
// Summary of independent equations
// Three material balances:CH4,C2H6 and N2
// One specified ratio: moles of CH4 to C2H6 equal 1.5
// One summation of mole fraction in mixture equals 1
d_o_f = n_un-n_ie ;// No. of degree of freedom

printf('Number of degree of freedom for the given system is  %i .\n',d_o_f);
```

Scilab code Exa 7.5 `Analysis of Degree of Freedom in the Production of Biomass`

```
clear ;
clc;
// Example 7.5
printf('Example 7.5\n\n');
//Page no. 182
// Solution

n_un=8 ;// Number of unknowns in the given problem- 8 values of mole fractions
n_ie =6 ;// Number of independent equations- six elemental balances
d_o_f= n_un-n_ie ;// Number of degree of freedom
printf('Number of degree of freedom for the given system is %i .\n',d_o_f);
//Note: Experiments show that the change in CH1.8O.5
```

N.16S.0045P.0055 and the change in C(alpha)H(beta)O(gamma) prove to be related by amount of biomass present and the maintenance coefficient(the moles of substrate per mole of biomass per second) so the respective quantities cannot be chosen independently.Consequently with this extra constraint,only one degree of freedom remains to be specified, the basis

Chapter 8

Solving Material Balance Problems for Single Units without Reaction

Scilab code Exa 8.1 `Extraction of Streptomycin from a Fermentation Broth`

```
clear ;
clc;
// Example 8.1
printf('Example 8.1\n\n');
//Page no. 197
// Solution

// Basis : 1 min
d_w =  1.0 ;// Density of aqueous solution -[g/cubic metre]
d_sol =  0.6 ;// Density of organic solvent -[g/cubic metre]

n_un = 8 ;// Number of unknowns in the given problem
n_ie  = 8 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('Number of degree of freedom for the given
```

```
   system is %i .\n',d_o_f);

// Material balance of Strep.
x =  (200*10+10*0-200*0.2)/10;//[g]
printf('Strep per litre of solvent is %.1f g .\n',x
   );

cnc = x/(1000*d_sol) ;//[g Strep/g of S]
printf('Strep per gram of solvent is %.4f g Strep/g
    of S .\n',cnc);

m_fr = cnc/(1+cnc) ;//Mass fraction
printf('Mass fraction of Strep is %.3f g .\n',m_fr)
   ;
```

Scilab code Exa 8.2 Separation of gases Using a Membrane

```
clear ;
clc;
// Example 8.2
printf('Example 8.2\n\n');
// Page no. 199
// Solution Fig. E8.2b

F_O2 = 0.21 ;// fraction of O2 in feed(F)
F_N2 = 0.79 ;// fraction of N2 in feed(F)
P_O2 = 0.25 ;// fraction of O2 in product(P)
P_N2 = 0.75 ;// fraction of N2 in  product(P)
F = 100 ;// Feed - [g mol]
w = 0.80 ;// Fraction of waste
W = w*F ;// Waste -[g mol]

// By analysis for degree of freedom , DOF comes to
   be zero
P = F - W ;// By overall balance - [g mol]
```

```
W_O2 = (F_O2*F - P*P_O2)/100 ;// Fraction of O2 in waste stream by O2 balance
W_N2 = (W - W_O2*100)/100 ;//Fraction of N2 in waste stream

printf('Composition of Waste Stream\n');
printf('\n  Component                  Fraction in waste stream\n');
printf('  O2                         %.2f\n',W_O2);
printf('  N2                         %.2f\n',W_N2);
```

Scilab code Exa 8.3 Overall analysis for a continuous Distillation Column

```
clear;
clc;
// Example 8.3
printf('Example 8.3\n\n');
// Page no. 202
// Solution

// Basis : 1 hr so F  = 1000 kg
F = 1000 ;// feed rate -[kg/hr]
P = F/10 ;// product mass flow rate -[kg/hr]

n_un = 9 ;// Number of unknowns in the given problem
n_ie  = 9 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('Number of degree of freedom for the given system is  %i .\n',d_o_f);

//  Overall mass balance: F  =  P+B
B =  F-P ;// bottom mass flow rate -[kg/hr]
printf('\n Bottom mass flow rate -               %.1f kg \n',B);

```

```
// Composition of bottoms by material balances
m_EtOH = 0.1*F-0.6*P ;// By EtOH balance-[kg]
m_H2O = 0.9*F - 0.4*P ;// By H2O balance-[kg]
total  = m_EtOH+m_H2O ;//[kg]
f_EtOH = m_EtOH/total ;// Mass fraction of EtOH
f_H2O = m_H2O/total ;// Mass fraction of H2O

printf(' Mass of EtOH in bottom -              %.1f kg \n',m_EtOH);
printf(' Mass of H2O in bottom -               %.1f kg \n',m_H2O);
printf(' Mass fraction of EtOH in bottom -     %.3f \n',f_EtOH);
printf(' Mass fraction of H2O in bottom -      %.3f \n',f_H2O);
```

Scilab code Exa 8.4 `Mixing of Battery Acid`

```
clear ;
clc;
// Example 8.4
printf('Example 8.4\n\n');
// Page no. 205
// Solution Fig E8.4

// Given
A = 200 ;// Mass of added solution [kg]
P_H2SO4 = .1863 ;//Fraction of H2SO4 in P(Final solution)
P_H2O = .8137 ;//Fraction of H2O in P(Final solution)
A_H2SO4 = .777 ;//Fraction of H2SO4 in A(Added solution)
A_H2O = .223 ;//Fraction of H2O in A(Added solution)
F_H2SO4 = .1243 ;//Fraction of H2SO4 in F(Original
```

```
solution)
F_H2O = .8757 ;//Fraction of H2O in F(Original solution)

// By analysis for degree of freedom , DOF comes to be zero
// Solve following equations simultaneously for F and P,
// P*P_H2O - F*F_H2O = A*A_H2O - By H2O balance
// P - F = A - By overall balance
a = [P_H2O -F_H2O;1 -1] ;// Matrix of coefficient
b = [A*A_H2O;A] ;// Matrix of contants
x = a\b ;// Matrix of solutions- P = x(1) and F = x(2)

printf(' Original solution taken-                              %.0i kg\n',x(2) );
printf('  Final solution or kilograms of battery acid formed-              %.0i kg\n',x(1) );
```

Scilab code Exa 8.5 `Drying`

```
clear ;
clc;
// Example 8.5
printf('Example 8.5\n\n');
// Page no. 207
// Solution Fig E8.5

// Given
W = 100 ;// Water removed - [kg]
A_H2O = 0.80 ;// Fraction of water in A(intial fish cake)
A_BDC = 0.20 ;//  Fraction of BDC(bone dry cake) in
```

```
    B(final dry fish cake)
B_H2O = 0.40 ;// Fraction of water in A(intial fish
    cake)
B_BDC = 0.60 ;//  Fraction of BDC(bone dry cake) in
    B(final dry fish cake)

// By analysis for degree of freedom , DOF comes to
    be zero
// Solve following equations simultaneously for A
    and B,
// A*A_H2O = B*B_H2O + W - By H2O balance
// A = B + W - By overall balance
a = [A_H2O -B_H2O;1 -1] ;// Matrix of coefficient
b = [W;W] ;// Matrix of contants
x = a\b; // Matrix of solutions- A = x(1) and B = x
    (2)

printf('Weight of the fish cake originally put into
    dryer -     %.0i kg\n',x(1) );
```

Scilab code Exa 8.6 `Crystallizaton`

```
clear ;
clc;
// Example 8.6
printf('Example 8.6\n\n');
// Page no. 209
// Solution

// Composition of initial solution at 30 degree C
s_30 = 38.8 ;//  solublity of Na2CO3 at 30 degree C,
     by using the table for solublity of Na2CO3-[g
    Na2CO3/100 g H2O]
If_Na2CO3 =  s_30/(s_30+100) ;// Initial mass
    fraction of Na2CO3
```

```
If_H2O = 1-If_Na2CO3 ;// Initial mass fraction of H2O

// Composition of crystals
// Basis : 1g mol Na2CO3.10H2O
n_mol_Na2CO3 = 1 ;// Number of moles of Na2CO3
n_mol_H2O = 10 ;// Number of moles of H2O
mwt_Na2CO3 = 106 ;// mol. wt of Na2CO3
mwt_H2O = 18 ;// mol. wt of H2O
m_Na2CO3 = mwt_Na2CO3*n_mol_Na2CO3 ;// Mass of Na2CO3
m_H2O = mwt_H2O*n_mol_H2O ;// Mass of H2O
Cf_Na2CO3 =  m_Na2CO3/(m_Na2CO3+m_H2O) ;// mass fraction of Na2CO3
Cf_H2O = 1-Cf_Na2CO3 ;// mass fraction of H2O

n_un = 9 ;// Number of unknowns in the given problem
n_ie  = 9 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('Number of degree of freedom for the given system is  %i .\n',d_o_f);

// Final composition of tank
//Basis :I = 10000 kg
// Material balance reduces to Accumulation  = final -initial = in-out(but in = 0)
I = 10000 ;//initial amount of saturated solution -[kg]
amt_C = 3000 ;// Amount of crystals formed -[kg]
Fm_Na2CO3 = I*If_Na2CO3-amt_C*Cf_Na2CO3 ;// Mass balance of Na2CO3
Fm_H2O = I*If_H2O-amt_C*Cf_H2O ;// Mass balance of H2O

//To find temperature ,T
s_T =  (Fm_Na2CO3/Fm_H2O)*100 ;// Solublity of Na2CO3 at temperature T
s_20 = 21.5 ;//Solublity of Na2CO3 at temperature 20
```

```
degree C ,from given table -[g Na2CO3/100 g H2O]
// Find T by interpolation
T =  30-((s_30-s_T)/(s_30-s_20))*(30-20) ;// Temperature -[degree C]
printf(' Temperature to which solution has to be cooled to get 3000 kg crystals is %.0f degree C.\n',T);
```

Scilab code Exa 8.7 `Hemodialysis`

```
clear ;
clc;
// Example 8.7
printf('Example 8.7\n\n');
// Page no. 213
// Solution

// Write given data
B_in =  1.1 ;// Flow rate in  of blood -[L/min]
B_out = 1.2;// Flow rate out of blood -[L/min]
S_in = 1.7;// Flow rate in  of solution -[L/min]

// Composition of input blood
B_in_CR = 2.72 ;//[g/L]
B_in_UR = 1.16 ;//[g/L]
B_in_U = 18 ;//[g/L]
B_in_P = 0.77 ;//[g/L]
B_in_K = 5.77 ;//[g/L]
B_in_Na = 13.0 ;//[g/L]
B_in_water = 1100 ;//[mL/min]

// Composition of output blood
B_out_CR = 0.120 ;//[g/L]
B_out_UR = 0.060;//[g/L]
B_out_U = 1.51 ;//[g/L]
```

```
B_out_P = 0.040 ;//[g/L]
B_out_K = 0.120 ;//[g/L]
B_out_Na = 3.21 ;//[g/L]
B_out_water = 1200 ;//[mL/min]

n_un = 7 ;// Number of unknowns in the given problem
n_ie  = 7 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('Number of degree of freedom for the given system is  %i .\n\n',d_o_f);

// Water balance in grams, assuming 1 ml is equivalent to 1 g
S_in_water = 1700 ;//[ml/min]
S_out_water =  B_in_water+ S_in_water - B_out_water;
S_out = S_out_water/1000 ;//[L/min]
printf(' Flow rate of water in output solution is  %.2f L/min.\n\n',S_out);

// The component balance in grams for CR,UR,U,P,K and Na are
S_out_CR  =  (B_in*B_in_CR - B_out*B_out_CR)/S_out;
S_out_UR  =  (B_in*B_in_UR - B_out*B_out_UR)/S_out;
S_out_U  =  (B_in*B_in_U - B_out*B_out_U)/S_out;
S_out_P  =  (B_in*B_in_P - B_out*B_out_P)/S_out;
S_out_K  =  (B_in*B_in_K - B_out*B_out_K)/S_out;
S_out_Na  =  (B_in*B_in_Na - B_out*B_out_Na)/S_out;
printf(' Component     Concentration(g/L) in output Dialysis solution   \n');
printf(' UR             %.2f     \n',S_out_UR);
printf(' CR             %.2f     \n',S_out_CR);
printf(' U              %.2f     \n',S_out_U);
printf(' P              %.2f     \n',S_out_P);
printf(' K              %.2f     \n',S_out_K);
printf(' Na             %.2f     \n',S_out_Na);
```

Chapter 9

The Chemical Reaction Equation and Stoichiometry

Scilab code Exa 9.1 `Balancing a reaction for a Biological Reaction`

```
clear;
clc;
// Example 9.1
printf('Example 9.1\n\n');
// Page no. 228
// Solution

// Given
//Main eqn. C6H12O6 + aO2 ---> bCO2 + cH2O
// By carbon balance
b = 6 ;

//By hydrogen balance
c = 6;

//Balancing oxygen in reaction
a = (c*1+b*2-6)/2;
printf('Value of a is  %i\n',a);
printf('Value of b is  %i\n',b);
```

```
printf('Value of c is %i\n',c);
```

Scilab code Exa 9.2 Use of Chemical Reaction to Calculate the Mass of Reactants gi

```
clear ;
clc;
// Example 9.2
printf('Example 9.2\n\n');
// Page no. 229
// Solution

m_CO2 = 44.0 ;//molecular wt-[g]
m_C7H16 = 100.1 ;//molecular wt-[g]
p_con = 50 ;// percentage conversion of CO2 to dry ice
amt_di = 500 ;// amount of dry ice to be produce per hour-[kg]
// By using the given equation
amt_C7H16 = (amt_di*m_C7H16)/((p_con/100)*m_CO2*7) ; // [kg]
printf('Amount of heptane required per hour to produce 500kg dry ice per hour is %.1f kg.\n',amt_C7H16);
```

Scilab code Exa 9.3 Application of Stoichiometry when more than one Reaction occur

```
clear ;
clc;
// Example 9.3
printf('Example 9.3\n\n');
// Page no. 230
// Solution

```

```
m_CaCO3 = 100.1 ;//molecular wt-[g]
m_MgCO3 = 84.32 ;//molecular wt-[g]
m_CaO = 56.08 ;//molecular wt-[g]
m_MgO = 40.32 ;//molecular wt-[g]
m_CO2 = 44.0 ;//molecular wt-[g]


// Limestone analysis
p_CaCO3 = 92.89 ;// percentage of CaCO3
p_MgCO3 = 5.41 ;//  percentage of MgCO3
inrt = 1.7 ;//percentage of inert

//(a)
amt_CaO  = (((p_CaCO3/100)*m_CaO)/m_CaCO3)*2000 ;// Pounds of CaO produced from 1 ton(2000lb) of limestone
printf(' Amount of CaO produced from 1 ton(2000lb) of limestone is  %.0f lb.\n',amt_CaO);

//(b)
mol_CaCO3 = (p_CaCO3/100)/m_CaCO3 ;// lb mol of CaCO3
mol_MgCO3 = (p_MgCO3/100)/m_MgCO3 ;// lb mol of MgCO3
total_mol = mol_CaCO3+mol_MgCO3;
amt_CO2 = total_mol*m_CO2 ;// Amount of CO2 recovered per pound of limestone-[lb]
printf('  Amount of CO2 recovered per pound of limestone is  %.3f lb.\n',amt_CO2);

//(c)
amt_CaO = m_CaO*mol_CaCO3 ;// since lb mol of CaO  = CaCO3
amt_MgO = m_MgO*mol_MgCO3 ;// since lb mol of MgO  = MgCO3
total_lime =  amt_CaO+amt_MgO+(inrt)/100 ;// total amount of lime per pound limestone
amt_lmst = 2000/total_lime ;// Amount of limestone
```

```
   required to make 1 ton(2000lb) of lime
printf('  Amount of limestone required to make 1 ton(2000lb) of lime   %.1f lb.\n',amt_lmst);
```

Scilab code Exa 9.4 Calculation of extent of Reaction

```
clear ;
clc;
// Example 9.4
printf('Example 9.4\n\n');
// Page no. 235
// Solution

f_NH3 = 5 ;// NH3 in feed -[g]
f_N2 =  100 ;// N2 in feed -[g]
f_H2 =  50 ;// H2 in feed -[g]
p_NH3 = 90 ;// NH3 in product -[g]
m_NH3  = 17 ;// Molecular wt. of NH3-[g]
m_N2  = 28 ;// Molecular wt. of N2-[g]
m_H2  = 2 ;// Molecular wt. of H2-[g]

// Extent of reaction can be calculated by using eqn . 9.3
// For NH3
ni = p_NH3/m_NH3 ;//[g mol NH3]
nio = f_NH3/m_NH3 ;//[g mol NH3]
vi = 2 ;// coefficint of NH3
ex_r =  (ni-nio)/vi ;// Extent of reaction - moles reacting

//Determine H2 and N2 in product of reaction by Eqn. 9.4
// For N2
nio_N2 = f_N2/m_N2 ;//[g mol N2]
vi_N2 = -1 ;// coefficint of N2
```

```
ni_N2 = nio_N2 + vi_N2*ex_r ;//N2 in product of reaction -[g moles ]
m_N2 = ni_N2*m_N2 ;// mass of N2 in product of reaction -[g]
printf('  N2 in product of reaction is  %.2f g moles \n',ni_N2);
printf('   Mass of N2 in product of reaction is  %.2f g   \n',m_N2);
// For H2
nio_H2 = f_H2/m_H2 ;//[g mol H2]
vi_H2 = -3 ;// coefficint of H2
ni_H2 = nio_H2 + vi_H2*ex_r ;//H2 in product of reaction -[g moles ]
m_H2 = ni_H2*m_H2 ;// mass of H2 in product of reaction -[g]
printf(' \n  H2 in product of reaction is  %.2f g moles  \n',ni_H2);
printf('  Mass of H2 in product of reaction is  %.2f g   \n',m_H2);

// ARP
m_SO2 = 64 ;// Molecular wt.of SO2-[g]
mol_SO2 =  2 ;// moles of SO2
ARP = (1/m_NH3)/(mol_SO2/m_SO2);
printf(' \n ARP is  %.2f    \n',ARP);
```

Scilab code Exa 9.5 `Calculation of Limiting and Excess Reactants`

```
clear ;
clc;
// Example 9.5
printf('Example 9.5\n\n');
// Page no. 238
// Solution

```

```
f_N2 = 10 ;// N2 in feed -[g]
f_H2 = 10 ;// H2 in feed -[g]
m_NH3 = 17.02;// Molecular wt. of NH3-[g]
m_N2 = 28 ;// Molecular wt. of N2-[g]
m_H2 = 2 ;// Molecular wt. of H2-[g]

// Extent of reaction can be calculated by using eqn. 9.3
// Based on N2
nio_N2 = f_N2/m_N2 ;//[g mol N2]
vi_N2 = -1 ;// coefficint of N2
ex_N2 = -(nio_N2)/vi_N2 ;// Max. extent of reaction based on N2

// Based on H2
nio_H2 = f_H2/m_H2 ;//[g mol H2]
vi_H2 = -3 ;// coefficint of H2
ex_H2 = -(nio_H2)/vi_H2 ;// Max. extent of reaction based on H2

//(a)
vi_NH3 = 2 ;// coefficint of NH3
mx_NH3 = ex_N2*vi_NH3*m_NH3 ;// Max. amount of NH3 that can be produced
printf(' (a) Max. amount of NH3 that can be produced is %.1f g\n',mx_NH3);

//(b) and (c)
if (ex_H2 > ex_N2 )
 printf('  (b) N2 is limiting reactant  \n');
 printf('  (c) H2 is excess reactant  \n');
 ex_r = ex_N2;
  else
printf('  (b) H2 is limiting reactant  \n');
 printf('  (c) N2 is excess reactant  \n');
 ex_r = ex_H2 ;
 end
```

Scilab code Exa 9.6 `Yeilds in the Reaction of Glucose to produce Ethanol`

```
clear;
clc;
// Example 9.6
printf('Example 9.6\n\n');
// Page no. 242
// Solution

//(a)
mol_bms =  0.59 ;// Biomass produced per g mol of
   glucose -[g mol biomass/ g mol glucose]
mw_bms =  23.74 ;// molecular wt. of biomass -[g]
mw_gls =  180.0 ;// molecular wt. of glucose -[g]
ms_bms =  (mol_bms*mw_bms)/mw_gls ;// Biomass
   produced per gram of glucose -[g biomass/ g
   glucose]
printf('(a) Biomass produced per gram of glucose is
  %.4f g biomass/ g glucose.',ms_bms);

//(b)
mol_etol = 1.3 ;//Ethanol produced per g mol of
   glucose -[g mol ethanol/ g mol glucose]
mw_etol =  46.0 ;// molecular wt. of ethanol -[g]
ms_etol =  (mol_etol*mw_etol)/mw_gls ;// Ethanol
   produced per gram of glucose -[g ethanol/ g
   glucose]
printf('\n (b) Ethanol produced per gram of glucose
   is %.3f g ethanol/ g glucose.',ms_etol);
```

Scilab code Exa 9.7 `Selectivity in the Production of Nanotubes`

```
clear ;
clc;
// Example 9.7
printf('Example 9.7\n\n');
// Page no. 243
// Solution

//Basis: 3 g mol H2 by reaction (a)
// 0.50 g mol C2H4 by reaction (b)
// by analysing reaction (a)  0.50 g mol C2H4
   corresponds to 0.50 g mol H2 produced in reaction
    (b)
// By using reaction (a)
H2_a = 3-0.50 ;// H2 produced in reaction (a)
C_a = (2/3)*H2_a ;// Nanotubes(the C) produced by
   reaction (a)
sel = C_a/0.50 ;// Selectivity of C reletive to C2H4
   -[g mol C/ g mol C2H4]
printf('Selectivity of C reletive to C2H4 is %.2f g
   mol C/ g mol C2H4.\n',sel)
```

Scilab code Exa 9.8 Calculation of various terms Pertaning to Reaction

```
clear;
clc;
// Example 9.8
printf('Example 9.8\n\n');
// Page no. 244
// Solution

m_C3H6 = 42.08;// molecular wt. of propene-[g]
m_C3H5Cl = 76.53 ;// molecular wt. of C3H5Cl-[g]
m_C3H6Cl2 = 112.99 ;// molecular wt. of C3H6Cl2-[g]
// Product analysis
pml_Cl2 = 141.0 ;// [g mol]
```

```
pml_C3H6 = 651.0 ;//[g mol]
pml_C3H5Cl = 4.6 ;// [g mol]
pml_C3H6Cl2 = 24.5 ;// [g mol]
pml_HCL = 4.6 ;//[g mol]

//(a)
a_Cl = pml_C3H5Cl; // Chlorine reacted by eqn.(a)
b_Cl = pml_C3H6Cl2 ;// Chlorine reacted by eqn.(b)
fed_Cl = pml_Cl2+a_Cl+b_Cl ;// Total chlorine fed to reactor -[g mol]
//by analysing reaction (a) and (b)
a_C3H6 = a_Cl+b_Cl ;// C3H6 reacted by reaction (a)
fed_C3H6 = pml_C3H6+a_C3H6 ;//Total C3H6 fed to reactor -[g mol]
printf('(a) Total chlorine fed to reactor is %.2f  g mol \n',fed_Cl);
printf('     Total C3H6 fed to reactor is %.2f  g mol \n',fed_C3H6);

//(b) and (c)
// Extent of reaction can be calculated by using eqn. 9.3
// Based on C3H6
nio_C3H6 = fed_C3H6 ;//[g mol C3H6]
vi_C3H6 = -1 ;// coefficint of C3H6
ex_C3H6 = -(nio_C3H6)/vi_C3H6 ;// Max. extent of reaction based on C3H6

// Based on Cl2
nio_Cl2 =  fed_Cl; //[g mol Cl2]
vi_Cl2 = -1 ;// coefficint of Cl2
ex_Cl2 = -(nio_Cl2)/vi_Cl2 ;// Max. extent of reaction based on Cl2

if (ex_Cl2 > ex_C3H6 )
 printf(' \n (b) C3H6 is limiting reactant  \n');
 printf('    (c)Cl2 is excess reactant  \n');
 ex_r = ex_C3H6;
```

```
  else
printf(' \n (b) Cl2 is limiting reactant  \n');
 printf(' (c) C3H6 is excess reactant  \n');
 ex_r = ex_Cl2;
 end

//(d)
fr_cn = pml_C3H5Cl/fed_C3H6 ;//Fractional conversion of C3H6 to C3H5Cl
printf(' \n (d) Fractional conversion of C3H6 to C3H5Cl is %.2e  \n',fr_cn);

//(e)
sel = pml_C3H5Cl/pml_C3H6Cl2 ;// Selectivity of C3H5Cl relative to C3H6Cl2
printf(' \n (e) Selectivity of C3H5Cl relative to C3H6Cl2 is %.2f g mol C3H5Cl/g mol C3H6Cl2  \n',sel);

//(f)
yld = (m_C3H5Cl*pml_C3H5Cl)/(m_C3H6*fed_C3H6) ;// Yield of C3H5Cl per g C3H6 fed to reactor
printf(' \n (f) Yield of C3H5Cl per g C3H6 fed to reactor is %.3f g C3H5Cl/g C3H6  \n',yld);

//(g)
vi_C3H5Cl = 1  ;// coefficint of C3H5Cl
vi_C3H6Cl2 = 1  ;// coefficint of C3H6Cl2
ex_a = (pml_C3H5Cl-0)/vi_C3H5Cl ;// Extent of reaction a as C3H5Cl is produced only in reaction a
ex_b = (pml_C3H6Cl2-0)/vi_C3H6Cl2 ;// Extent of reaction b as C3H6Cl2 is produced only in reaction b
printf(' \n (g) Extent of reaction a as C3H5Cl is produced only in reaction a is %.1f   \n',ex_a);
printf('     Extent of reaction b as C3H6Cl2 is produced only in reaction b %.1f   \n',ex_b);
```

```

//(h)
in_Cl = fed_Cl*2 ;//Entering Cl -[g mol]
out_Cl  = pml_HCL ;// Exiting Cl in HCl-[g mol]
ef_w = out_Cl/in_Cl ;// Mole efficiency of waste
ef_pr =  1-ef_w ;// Mole efficiency of product
printf('\n (h) Mole efficiency of product is %.3f \n
   ',ef_pr);
```

Chapter 10

Material Balances for Processes Involving Reaction

Scilab code Exa 10.1 `Reaction in which Fraction Conversion is specified`

```
clear;
clc;
// Example 10.1
printf('Example 10.1\n\n');
//Page no. 264
// Solution

F = 100 ;// feed to the reactor -[g mol]
// Composition of feed
CH4 = 0.4*F ;// [g mol]
Cl2 = 0.5*F ;// [g mol]
N2= 0.1*F ;//[g mol]

// Extent of reaction can be calculated by using eqn . 9.3
// Based on CH4
nio_CH4 = CH4 ;//[g mol CH4]
vi_CH4 = -1 ;// coefficint of CH4
ex_CH4 = -(nio_CH4)/vi_CH4 ;// Max. extent of
```

```
   reaction based on CH4

// Based on Cl2
nio_Cl2 = Cl2 ;//[g mol Cl2]
vi_Cl2 = -1 ;// coefficint of Cl2
ex_Cl2 = -(nio_Cl2)/vi_Cl2 ;// Max. extent of
   reaction based on Cl2

if (ex_Cl2 > ex_CH4 )
 printf(' \n CH4 is limiting reactant  \n');
  else
printf(' \n (b) Cl2 is limiting reactant  \n');
end
// By execution of above block its clear that CH4 is
    limiting reactant , therefore extent of reaction
   is
cn_CH4 = 67/100 ;// percentage conversion of CH4
ex_r = (-cn_CH4)*CH4/vi_CH4 ;// extent of reaction
printf(' extent of reaction is %.1f g moles reacting
    \n',ex_r);

n_un = 11 ;// Number of unknowns in the given
   problem
n_ie = 11 ;// Number of independent equations
d_o_f = n_un-n_ie ;// Number of degree of freedom
printf(' Number of degree of freedom for the given
   system is  %i \n',d_o_f);

// Product composition using species balance using
   eqn.10.2
vi_CH3Cl = 1;
vi_HCl = 1;
vi_N2 = 0;
p_CH4 = CH4+(vi_CH4*ex_r);// [g mol]
p_Cl2 = Cl2+(vi_Cl2*ex_r);// [g mol]
p_CH3Cl = 0+(vi_CH3Cl*ex_r);// [g mol]
p_HCl = 0+(vi_HCl*ex_r);// [g mol]
p_N2 =  N2+(vi_N2*ex_r);// [g mol]
```

```
// As we have taken F = 100 so answers we are getting can be directly used as percentage composition
printf('\n\nComposition of product stream in %% g mol of products\n');
printf('\nProduct                 Percentage g mol\n');
printf('\nCH4                     %.1f%% g mol\n',p_CH4);
printf('\nCl2                     %.1f%% g mol\n',p_Cl2);
printf('\nCH3Cl                   %.1f%% g mol\n',p_CH3Cl);
printf('\nHCl                     %.1f%% g mol\n',p_HCl);
printf('\nN2                      %.1f%% g mol\n',p_N2);
```

Scilab code Exa 10.2 A reaction in which Fraction Conversion is to be Calculated

```
clear ;
clc;
// Example 10.2
printf('Example 10.2\n\n');
// Page no. 266
// Solution

S = 5000 ;// Sulphur [lb]
// Composition of feed
CH4 =  80 ;// [%]
H2S =  20 ;// [%]

n_un = 11 ;// Number of unknowns in the given problem
n_ie  = 11 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('Number of degree of freedom for the given system is  %i \n',d_o_f);
m_S = 32.0 ;//molecular wt. of S -[lb]
mol_S = S/32.0;
```

```
// Extent of reaction can be calculated by using eqn. 9.3
// Based on S
nio_S = 0 ;//[g mol S]
ni_S = mol_S ;//[g mol S]
vi_S = 3 ;// coefficint of S -from given reaction
ex_r = (ni_S-nio_S)/vi_S ;//  Extent of reaction based on S
printf(' Extent of reaction is %.1f g moles reacting\n',ex_r);

// Product composition
vi_H2O = 2 ;// coefficint of H2O
vi_H2S = -2 ;// coefficint of H2S
vi_SO2 = -1 ;//coefficint of SO2
vi_CH4 = 0 ;//coefficint of CH4
P_H2O = 0+(vi_H2O*ex_r);// [lb mol]
P_H2S = P_H2O/10 ;//[lb mol]
P_SO2 = 3*P_H2S ;//[lb mol]

F = (P_H2S-vi_H2S*ex_r)/(H2S/100) ;// total feed -[lb mol]
F_SO2 = P_SO2-(vi_SO2*ex_r);// feed rate of SO2- [lb mol]
F_CH4 = (CH4/100)*F+vi_CH4*ex_r ;//feed rate of CH4-[lb mol]
F_H2S = ((H2S/100)*F) ;// feed rate of H2S-[lb mol]

// We can see from situation that H2S is limiting reagent as ratio of SO2 to H2S in the product gas (3/1) is greater than their molar ratio in chemical reaction(2/1)
f_cn = -(vi_H2S*ex_r)/((H2S/100)*F) ;// Fractional conversion of limiting reagent

printf('\n(1)Feed rate of H2S- %.1f lb mol\n',F_H2S);
printf('(2)Feed rate of SO2- %.1f lb mol\n',F_SO2);
```

```
printf('(3)Fractional conversion of limiting reagent - %.2f \n',f_cn);
```

Scilab code Exa 10.3 Material Balances for Process in which two Simultaneous React

```
clear ;
clc;
// Example 10.3
printf('Example 10.3\n\n');
// Page no. 270
// Solution

F = 1 ;//CH3OH -[gmol]
// Extent of reactions can be calculated by using eqn. 10.5
// For reaction 1 based on CH3OH is limiting reagent
f_cn = 90 ;//[%]
vi_CH3OH = -1 ;//coefficint of CH3OH
ex_r1 = (-90/100)/vi_CH3OH ;// Extent of reaction based on CH3OH
printf(' Extent of reaction 1 is %.2f g moles reacting \n',ex_r1);
//For reaction 2
yld = 75 ;//[%]
ex_r2 = ex_r1-(F*(yld/100));
printf(' Extent of reaction 2 is %.2f g moles reacting \n',ex_r2);

// For amount of air
// Entering O2 is twice the O2 required by reaction 1,therefore
f_O2 = 0.21 ;// mol. fraction of O2
f_N2 = 0.79 ;// mol. fraction of N2
n_O2 = 2*((1/2)*F) ;// entering oxygen -[g mol]
air =  n_O2/f_O2 ;// Amount of air entering
```

```
n_N2 = air-n_O2  ;// entering nitrogen -[g mol]

// Degree of freedom analysis
n_un = 11 ;// Number of unknowns in the given problem
n_ie  = 11 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('  Number of degree of freedom for the given system is  %i \n',d_o_f);

// Reaction 1
v1_CH3OH = -1 ;//coefficint of CH3OH
v1_O2 = -1/2 ;//coefficint of O2
v1_CH2O = 1 ;//coefficint of CH2O
v1_H2O = 1 ;//coefficint of H2O
v1_CO = 0  ;//coefficient of CO
//Reaction 2
v2_O2 = -1/2 ;//coefficint of O2
v2_CH2O = -1 ;//coefficint of CH2O
v2_H2O = 1 ;//coefficint of H2O
v2_CO = 1 ;//coefficient of CO
P = F+air +(v1_CH3OH+v1_O2+v1_CH2O+v1_H2O)*ex_r1 +(v2_O2+v2_CH2O+v2_H2O+v2_CO)*ex_r2 ;// Product -[g mol]

no_CH3OH = F+(v1_CH3OH*ex_r1)+0 ;// [g mol]
no_O2 = n_O2+(v1_O2*ex_r1)+v2_O2*ex_r2 ;// [g mol]
no_CH2O = 0 + v1_CH2O*ex_r1 +v2_CH2O*ex_r2 ;//[g mol]
no_CO =  0+v1_CO*ex_r1 +v2_CO*ex_r2 ;//[g mol]
no_H2O = 0+v1_H2O*ex_r1+v2_H2O*ex_r2 ;// [g mol]
no_N2 =  n_N2-0-0 ;// [g mol]

// Composition of product
y_CH3OH = (no_CH3OH/P )*100 ;// mole %
y_O2 = (no_O2/P)*100 ;// mole %
y_CH2O = (no_CH2O/P)*100 ;// mole %
y_CO = (no_CO/P)*100 ;// mole %
```

```
y_H2O = (no_H2O/P)*100 ;// mole %
y_N2 = (no_N2/P )*100;// mole %

printf('\nComposition of product\n');
printf('Component          mole percent\n');
printf(' CH3OH             %.1f %%\n',y_CH3OH);
printf(' O2                %.1f %%\n',y_O2);
printf(' CH2O              %.1f %%\n',y_CH2O);
printf(' CO                %.1f %%\n',y_CO);
printf(' H2O               %.1f %%\n',y_H2O);
printf(' N2                %.1f %%\n',y_N2);
```

Scilab code Exa 10.4 `Analysis of Bioreactor`

```
clear;
clc;
// Example 10.4
printf('Example 10.4\n\n');
// Page no. 273
// Solution

F = 4000 ;//[kg]
m_H2O = 18.02 ;// molecular masss of water
m_C6H12O6 = 180.1 ;// molecular mass of glucose
m_CO2 = 44 ;//molecular mass of CO2
m_C2H3CO2H = 72.03 ;// molecular mass of C2H3CO2H
m_C2H5OH = 46.05 ;// molecular mass of ethanol

p_H2O = 88 ;// [%]
p_C6H12O6 = 12;// [%]
ni_H2O = (F*p_H2O/100)/m_H2O ;// initial moles of water
ni_C6H12O6 = (F*(p_C6H12O6/100))/m_C6H12O6 ;// initial moles of glucose

```

```
// Degree of freedom analysis
n_un = 9 ;// Number of unknowns in the given problem
n_ie  = 9 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('Number of degree of freedom for the given system is  %i \n',d_o_f);

ur_C6H12O6 =  90 ;//[kg]
pr_CO2 = 120 ;//[kg]
nf_C6H12O6 = ur_C6H12O6/m_C6H12O6 ;// [kmoles]
nf_CO2 = pr_CO2/m_CO2 ;// [kmoles]

// solve following eqn. (b) and (e)simultaneously
//(b):  nf_C6H12O6 = ni_C6H12O6+-1*ex_r1+-1*ex_r2
//(e):  nf_CO2 = 0+2*ex_r1+ 0*ex_r2
a = [-1 -1;2 0];// matrix formed by coefficients of unknowns
b = [(nf_C6H12O6-ni_C6H12O6);nf_CO2];//matrix formed by constant
x = a^(-1)*b;//matrix formed by solution
printf(' Extent of reaction 1 is %.3f kg moles reacting \n',x(1));
printf(' Extent of reaction 2 is %.3f kg moles reacting \n',x(2));

nf_H2O = ni_H2O+0*x(1) +2*x(2);// from eqn. (a)-[kmoles]
nf_C2H5OH = 0+2*x(1)+0*x(2);// from eqn.(c)-[kmoles]
nf_C2H3CO2H = 0+0*x(1)+2*x(2) ;//from eqn.(d)-[kmoles]
total_wt = m_H2O*nf_H2O + m_C6H12O6*nf_C6H12O6 + m_CO2*nf_CO2 + m_C2H3CO2H*nf_C2H3CO2H + m_C2H5OH*nf_C2H5OH;
mp_C2H5OH = (m_C2H5OH*nf_C2H5OH*100)/total_wt ;// Mass percent of ethanol -[%]
mp_C2H3CO2H = (m_C2H3CO2H*nf_C2H3CO2H*100)/total_wt ;//Mass percent of propenoic acid -[%]

```

```
printf(' \n Mass percent of ethanol in broth at end of fermentation process is %.1f %%\n',mp_C2H5OH);
printf(' Mass percent of propenoic acid  in broth at  end of fermentation process is %.1f %%\n',mp_C2H3CO2H);
```

Scilab code Exa 10.5 `Solution using Elemental Balances`

```
clear ;
clc;
// Example 10.5
printf('Example 10.5\n\n');
// Page no. 279
// Solution

//(a)Solution of Example 10.1 using element balance
printf('(a)Solution of Example 10.1 using element balance\n');
F = 100 ;// feed to the reactor -[g mol]
// Composition of feed
CH4 =  0.4*F ;// [g mol]
Cl2 =  0.5*F ;// [g mol]
N2 =  0.1*F ;//[g mol]

n_un = 10 ;// Number of unknowns in the given problem(excluding extent of reaction)
n_ie  = 10 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('    Number of degree of freedom for the given system is  %i \n',d_o_f);

// Extent of reaction can be calculated by using eqn. 9.3
// Based on CH4
```

```
nio_CH4 = CH4 ;//[g mol CH4]
vi_CH4 = -1; // coefficint of CH4
ex_CH4 = -(nio_CH4)/vi_CH4 ;// Max. extent of reaction based on CH4

// Based on Cl2
nio_Cl2 =  Cl2 ;//[g mol Cl2]
vi_Cl2 = -1 ;// coefficint of Cl2
ex_Cl2 = -(nio_Cl2)/vi_Cl2 ;// Max. extent of reaction based on Cl2

if (ex_Cl2 > ex_CH4 )
 printf('    CH4 is limiting reactant  \n');
  else
printf(' \n (b) Cl2 is limiting reactant  \n');
end
// By execution of above block its clear that CH4 is limiting reactant,therefore
cn_CH4 =  67/100 ;// percentage conversion of CH4(limiting reagent)
no_CH4 = CH4-(cn_CH4*CH4) ;//CH4 in product -[g mol]

// Product composition using element balance
// By N2 balance
no_N2 = N2;//N2 in product -[g mol]

C = CH4 ;//moles of CH4  =  moles of C (by molecular formula)
H = 4*CH4 ;// moles of H  =  4*moles of CH4 (by molecular formula)
Cl = 2*Cl2 ;// moles of Cl = 2* moles of Cl2 (by molecular formula)
// Solving following 3 eqn. obtained from balance of C,H,Cl for 3 unknowns
//1.  C-no_CH4*1 =  1*no_CH3Cl
//2. H-4*no_CH4 = 3*no_CH3Cl+no_HCl*1
//3. Cl = no_Cl2*2 + no_HCl*1+1*no_CH3Cl
a = [0 0 1;0 1 3;2 1 1] ;// matrix formed by
```

```
coefficients of unknowns
b = [C-no_CH4*1;H-4*no_CH4;Cl] ;//matrix formed by constant
x = a^(-1)*b ;// matrix of solution

// As we have taken F = 100 so answers we are getting can be directly used as percentage composition
printf('\nComposition of product stream in %% g mol of products\n');
printf('Product              Percentage g mol\n');
printf('\nCH4                    %.1f%% g mol\n',no_CH4);
printf('\nCl2                    %.1f%% g mol\n',x(1));
printf('\nCH3Cl                  %.1f%% g mol\n',x(3));
printf('\nHCl                    %.1f%% g mol\n',x(2));
printf('\nN2                     %.1f%% g mol\n',no_N2);

//(b)Solution of Example 10.3 using element balance
printf('
--------------------------------------------------------------------------------
');
printf('\n\n(b)Solution of Example 10.3 using element balance\n');
F = 1 ;//CH3OH -[gmol]
yld = 75 ;//[%]
cnv = 90 ;//conversion of methanol-[%]

// For amount of air
// Entering O2 is twice the O2 required by reaction 1,therefore
f_O2 = 0.21 ;// mol. fraction of O2
f_N2 = 0.79 ;// mol. fraction of N2
n_O2 = 2*((1/2)*F) ;// entering oxygen -[g mol]
air =  n_O2/f_O2 ;// Amount of air entering
n_N2 = air-n_O2  ;// entering nitrogen -[g mol]

// Degree of freedom analysis
```

```
n_un = 9 ;// Number of unknowns in the given problem (excluding extent of reactions)
n_ie  = 9 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf(' Number of degree of freedom for the given system is  %i \n',d_o_f);

// Product composition using element balance
// By N2 balance
no_N2 = n_N2 ;// inert ,terefore input  =  output
C = 1*F ;//moles of C  =  moles of CH3OH (by molecular formula)
H = 4*F ;//moles of H  =  4*moles of CH3OH (by molecular formula)
O =  1*F +2*n_O2;// moles of O =  1*moles of CH3OH + O in air
no_CH2O = yld/100 ;//[g mol]
no_CH3OH = F-((cnv/100)*F);// [g mol]

// Solving following 3 eqn. obtained from balance of C,H,O for 3 unknowns
a = [0 0 1;0 2 0;2 1 1] ;// matrix formed by coefficients of unknowns
b = [(C-(no_CH3OH*1+no_CH2O*1));(H-(4*no_CH3OH+2*no_CH2O));(O-(no_CH3OH*1+no_CH2O*1))] ;//matrix formed by constant
x = a\b ;// matrix of solution

P = no_CH2O+no_CH3OH+no_N2+x(1)+x(2)+x(3);

// Composition of product
y_CH3OH = (no_CH3OH/P )*100;// mole %
y_O2 = ((x(1))/P)*100;// mole %
y_CH2O = (no_CH2O/P)*100 ;// mole %
y_CO = (x(3)/P)*100 ;// mole %
y_H2O = (x(2)/P)*100 ;// mole %
y_N2 = (no_N2/P )*100;// mole %

```

```

printf('\nComposition of product\n');
printf('Component          mole percent\n');
printf(' CH3OH             %.1f %%\n',y_CH3OH);
printf(' O2                %.1f %%\n',y_O2);
printf(' CH2O              %.1f %%\n',y_CH2O);
printf(' CO                %.1f %%\n',y_CO);
printf(' H2O               %.1f %%\n',y_H2O);
printf(' N2                %.1f %%\n',y_N2);
```

Scilab code Exa 10.6 Use of Elemental Balance to Solve a Hydrocracking Problem

```
clear ;
clc;
// Example 10.6
printf('Example 10.6\n\n');
// Page no. 281
// Solution

// Basis: P=100 // Product from the reactor -[g mol]
P=100 ;//Product from the reactor -[g mol]
// Composition of product
C3H8 = 0.195*P ;// [g mol]
C4H10 = 0.594*P ;// [g mol]
C5H12 = 0.211*P;// [g mol]

n_un = 3 ;// Number of unknowns in the given problem (excluding extent of reaction)
n_ie  = 3 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('Number of degree of freedom for the given system is  %i \n',d_o_f);

C = C3H8*3+C4H10*4+C5H12*5 ;// moles of C on product side
```

```
H = C3H8*8+C4H10*10+C5H12*12 ;// moles of H on product side
// Solve following eqn.( obtained by element balance of C & H) for F and G
//8F+0G = C
//18F+2G = H
a = [8 0;18 2] ;// matrix formed by coefficients of unknowns
b = [C;H] ;//matrix formed by constant
x = a\b ;// matrix of solution

R = x(2)/x(1) ;// Ratio of H2 consumed to C8H18 reacted = G/F
printf(' Molar ratio of H2 consumed to C8H18 reacted is %.3f \n',R);
```

Scilab code Exa 10.7 Excesss Air

```
clear ;
clc;
// Example 10.7
printf('Example 10.7\n\n');
// Page no. 286
// Solution

C3H8 = 20 ;//C3H8 burned in a test -[kg]
m_C3H8 = 44.09 ;// mol. wt . of 1 kmol C3H8
cf_O2 = 5 ;// coefficient of O2 in given reaction
air = 400 ;// Air given -[kg]
m_air = 29 ;// molecular wt. of 1kmol air -[kg]
O2p = 21 ;// percentage of O2 in air -[%]
p_CO2 = 44 ;// CO2 produced -[kg]
p_CO = 12 ;// CO produced -[kg]
O2 = (air*O2p/100)/(m_air) ;// amount of entering O2 -[k mol]
```

```
rqO2 = (C3H8*cf_O2)/(m_C3H8) ;// Required O2 for given reaction
ex_air = ((O2-rqO2)*100)/rqO2 ;// Excess air percent -[%]
printf('Excess air percent is %.0f %%.\n',ex_air);
```

Scilab code Exa 10.8 `A Fuel Cell to generate Electricity from Methane`

```
clear;
clc;
// Example 10.8
printf('Example 10.8\n\n');
// Page no. 287
// Solution

F = 16 ;// feed of CH4 -[kg]
CH4p = 100 ;//[%]
m_CH4 = 16 ;// mass of kmol of CH4-[kg]
mol_CH4 = (F*CH4p/100)/m_CH4;//k moles of CH4 in feed -[kmol]
air = 300 ;// Air given -[kg]
m_air = 29 ;// molecular wt. of 1kmol air -[kg]
mol_air = air/m_air ;// kmoles of air -[kmol]
O2p = 21 ;// percentage of O2 in air -[%]
O2 = (mol_air*O2p/100)  ;// amount of entering O2-[k mol]
N2 = mol_air-O2 ;// amount of entering N2-[k mol]

// Degree of freedom analysis
n_un = 8 ;// Number of unknowns in the given problem (excluding extent of reactions)
n_ie  = 8 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('Number of degree of freedom for the given system is  %i \n',d_o_f);
```

```

// Product composition analysis using element balance of C,H,O and N
p_N2 = N2 ;// inert
C_in = 1*mol_CH4 ;// kmoles of carbon in input-[kmol]
H_in = 4*mol_CH4 ;// kmoles of hydrogen in input-[kmol]
O_in = 2*O2 ;// kmoles of oxygen in input-[kmol]
p_CO2 = C_in/1 ;//kmoles of  CO2 in product obtained by carbon balance-[kmol]
p_H2O = H_in/2 ;//kmoles of  H2O in product obtained by hydrogen balance-[kmol]
p_O2 = (O_in-(2*p_CO2+p_H2O))/2 ;//kmoles of  O2 in product obtained  by oxygen balance-[kmol]
p_CH4 = 0 ;// Complete reaction occurs
P = p_CH4 + p_N2+  p_CO2 + p_H2O + p_O2;

y_N2 = p_N2*100/P ;//[mol %]
y_CO2 = p_CO2*100/P ;//[mol %]
y_H2O = p_H2O*100/P ;//[mol %]
y_O2 = p_O2*100/P ;//[mol %]
y_CH4 = p_CH4*100/P ;//[mol %]

printf('\nComposition of product\n');
printf('Component           mole percent\n');
printf(' CH4                %.1f %%\n',y_CH4);
printf(' O2                 %.1f %%\n',y_O2);
printf(' CO2                %.1f %%\n',y_CO2);
printf(' H2O                %.1f %%\n',y_H2O);
printf(' N2                 %.1f %%\n',y_N2);
```

Scilab code Exa 10.9 `Combustion of Coal`

```
clear;
```

```
clc;
// Example 10.9
printf('Example 10.9\n\n');
// Page no. 290
// Solution

F = 100  ;// feed of coal -[lb]
// given coal composition-given
C = 83.05 ;//[%]
H = 4.45 ;//[%]
O = 3.36 ;// [%]
N = 1.08 ;// [%]
S = 0.70  ;//[%]
ash = 7.36;//[%]
H2O = 3.9 ;//[%]
w_C = 12 ;// mol. wt. of C
w_H =  1.008;//mol. wt. of H
w_O = 16 ;// mol. wt. of O
w_N = 14 ;// mol. wt. of N
w_S = 32  ;//mol. wt. of S

//given stack gas analysis-given
CO2 =  15.4 ;//[%]
CO = 0.0 ;//[%]
O2 = 4.0 ;// [%]
N2 = 80.6 ;//[%]
//given refuse analysis
ash_R = 86 ;//[%]
odr = 14 ;//[%]

H2O_air =  .0048 ;// [lb H2O/lb dry air]
m_air = 29 ;// mol. wt. of air
mf_O2 = 0.21 ;// mole fraction of  O2 in air
mf_N2 = 0.79 ;//mole fraction of  N2 in air
m_H2O =  18 ;// mol. wt. of H2O

H_cl = (H2O*2)/m_H2O ;// lb mol of H in coal moisture
```

```
O_cl = H_cl/2 ;// lb mol of O in coal moisture

H_air = (H2O_air*m_air )/m_H2O;// lb mol of H per lb mol air
O_air = H_air/2 ;// lb mol of O per lb mol  air

// Ash balance to get refuse(R)
R = ash/(ash_R/100) ;// Refuse -[lb]
//refuse composition
pub_cl = 14 ;// percentage of unburned coal in refuse -[%]
ub_cl =  (14/100)*R ;// amount of unburned coal in refuse
C_p = (C/(100-ash))*ub_cl ;//  C in unburned coal -[lb]
H_p = (H/(100-ash))*ub_cl ;//  H in unburned coal -[lb]
O_p =  (O/(100-ash))*ub_cl ;//  O in unburned coal -[lb]
N_p =  (N/(100-ash))*ub_cl ;//  N in unburned coal -[lb]
S_p =  (S/(100-ash))*ub_cl ;//  S in unburned coal -[lb]
mol_C =  C_p/w_C;// lb mol of C
mol_H = H_p/w_H ;// lb mol of H
mol_N = N_p/w_N ;// lb mol of N
mol_O = O_p/w_O ;// lb mol of O
mol_S = S_p/w_S ;// lb mol of S

// Degree of freedom analysis
n_un = 4 ;// Number of unknowns in the given problem (excluding extent of reactions)
n_ie  = 4 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('Number of degree of freedom for the given system is  %i \n\n',d_o_f);

//Using element balance of C+S, N& H
```

```
P = (C/w_C + S/w_S - (mol_C+mol_S ))/.154 ;// mol of stack gas-[lb mol]
A = (2*P*.806 +2*mol_N-N/w_N)/(2*mf_N2) ;// mol of air -[lb mol]
W = (H/w_H +H_cl+H_air*A-mol_H)/2 ;// moles of exit water-[lb mol]
printf(' Moles of stack gas(P)    -      %.1f  lb mol \n',P);
printf(' Moles of air (A)         -      %.1f lb mol \n',A);
printf(' Moles of exit water(W)   -      %.1f lb mol \n',W);
// by using P,W , A and O2 balance we get 19.8 = 20.3 , therefore difference is about 1%

//Calculation of excess air
// For O2 required
C_req =  (C/w_C)/1 ;// O2 required by entering C given by reaction C+O2--->CO2 -[lb mol]
H_req = (H/w_H)/4 ;// O2 required by entering H by given reaction H2+(1/2)*O2--->H20-[lb mol]
N_req = 0 ;// inert
O_req = (O/w_O)/2 ;// O2 required by entering O-[lb mol]
S_req = (S/w_S)/1 ;// O2 required by entering S-given by S+O2--->SO2 -[lb mol]
total_O2_req =  C_req+H_req+N_req+O_req +S_req ;// Total oxygen required -[lb mol]
O2_in = A*mf_O2 ;// O2 entering in air
ex_air = 100*((O2_in-total_O2_req)/total_O2_req) ;// [% of excess air]
printf('\n Excess air is %.1f %%.\n',ex_air);
```

Chapter 11

Material Balance Problems involving Multiple Units

Scilab code Exa 11.1 `Determination of the Number of Independent Material Balance i`

```
clear ;
clc;
// Example 11.1
printf('Example 11.1\n\n');
// Page no. 311
// Solution

// Composition of each stream
w_A1 = 1 ;//concentration of A in 1
w_B2 = 1 ;//  concentration of B in 2
w_A3 = 0.8 ;// concentration of A in 3
w_B3 = 0.2 ;// concentration of B in 3
w_C4 = 1 ;// concentration of C in 4
w_A5  = 0.571 ;//concentration of A in 5
w_B5  = 0.143 ;//concentration of B in 5
w_C5  =  0.286 ;//concentration of C in 5
w_D6 = 1;// concentration of D in 6
w_A7 = 0.714 ;// concentration of A in 7
w_B7 = 0.286 ;// concentration of B in 7
```

```
w_B8 = 0.333 ;//concentration of B in 8
w_C8 = .667 ;//concentration of C in 8

us1 = 2 ;// Species involved in unit 1
us2 = 3 ;// Species involved in unit 2
us3 = 4 ;// Species involved in unit 3
total_sp = us1+us2+us3 ;//  Total species in system

// Element balance of all systems
printf('Number of possible equations are 9, they are as follows- \n');
printf(' Subsystem 1\n');
printf('        A:  F1*w_A1+F2*0  =  F3*w_A3      (a)\n');
printf('        B:F1*0 + F2*w_B2 = F3*w_B3       (b)\n');
printf(' Subsystem 2\n');
printf('        A:  F3*w_A3+F4*0  =  F5*w_A5      (c)\n');
printf('        B:F3*w_B3 + F4*0 = F5*w_B5       (d)\n');
printf('        C:  F3*0+F4*w_C4 = F5*w_C5       (e)\n');
printf(' Subsystem 3\n');
printf('        A:  F5*w_A5+F6*0  =  F7*w_A7+F8*0   (f)\n');
printf('        B:F5*w_B5 + F6*0 = F7*0+F8*w_B8   (g)\n');
printf('        C:  F5*w_C5+F6*0 = F7*0+F8*w_C8   (h)\n');
printf('        D:F5*w_C5+F6*0  = F7*0+F8*w_C8   (i)\n');
printf('\n The above equations do not form a unique set\n');

// By inspection we can see that only 7 equations are independent
//Independent Equations are:
```

```
// Subsystem 1
//A: F1*w_A1+F2*0 = F3*w_A3 (a)
//B:F1*0 + F2*w_B2 = F3*w_B3 (b)
//Subsystem 2
//A: F3*w_A3+F4*0 = F5*w_A5 (c)
// C: F3*0+F4*w_C4 = F5*w_C5 (e)
// Subsystem 3
//A: F5*w_A5+F6*0 = F7*w_A7+F8*0 (f)
//B:F5*w_B5 + F6*0 = F7*0+F8*w_B8 (g)
//D:F5*w_C5+F6*0 = F7*0+F8*w_C8 (i)

printf('\n Number of independent equations are 7 \n');
```

Scilab code Exa 11.2 Material Balances for Multiple Units in which no Reaction occ

```
clear;
clc;
// Example 11.2
printf('Example 11.2\n\n');
// Page no.315
// Solution

//Basis:1 hr
G = 1400 ;//[kg]
//Unit 1
// Degree of freedom analysis
n_un = 16 ;// Number of unknowns in the given problem(excluding extent of reactions)
n_ie  = 16 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf('For unit 1 number of degree of freedom for the given system is  %i .\n',d_o_f);
//Given
o1_air = 0.995 ;// Mass fraction of air at out of
```

```
unit 1 in A
i1_air = 0.95 ;// Mass fraction air at in of unit 1 in G
i1_wtr = 0.02;// Mass fraction water at in of unit 1 in G
F1_wtr = 0.81 ;// Mass fraction of water at out of unit 1 in F
o1_wtr = 0.005 ;// Mass fraction of water at out of unit 1 in A
o2_wtr = 0.96 ;// Mass fraction of water at out of unit 2 in B
o3_wtr = 0.01;// Mass fraction of water at out of unit 3 in D
i1_act = 0.03 ;// Mass fraction of acetone at in of unit 1 in G
F1_act = 0.19 ;//  Mass fraction of acetone at out of unit 1 in F
o3_act = 0.99 ;//  Mass fraction of acetone at out of unit 3 in D
o2_act  = 0.04 ;//  Mass fraction of acetone at out of unit 2 in B

//Mass balance to get A ,W & F
A = G*i1_air/o1_air ;//air -[kg]
F = G*i1_act/F1_act ;//[kg]
W = (F*F1_wtr+A*o1_wtr-G*i1_wtr)/1 ;//Pure water in -[kg]
// unit 2 and 3
// Degree of freedom analysis
n_un = 9 ;// Number of unknowns in the given problem (excluding extent of reactions)
n_ie  = 9 ;// Number of independent equations
d_o_f =  n_un-n_ie ;// Number of degree of freedom
printf(' For unit 2 and 3 number of degree of freedom for the given system is  %i .\n',d_o_f);
// Mass balance
// solving eqn (d)& (e) simultaneously for D and B
a = [o3_act o2_act;o3_wtr o2_wtr];// Matrix formed
```

```
   by coefficients of unknown
b = [F*F1_act;F*F1_wtr];// Matrix formed by constant
x = a\b ;// Solution matrix-x(1) = D and x(2) = B
printf('\n W-Pure water in to unit 1  -  %.2f kg/hr\
   n',W);
printf(' A-Air out of unit 1        -  %.2f kg/hr\n'
   ,A);
printf(' F-out of unit 1            -  %.2f kg/hr\n'
   ,F);
printf(' B-out of unit 2            -  %.2f kg/hr\n'
   ,x(2));
printf(' D-out of unit 3            -  %.2f kg/hr\n'
   ,x(1));
```

Scilab code Exa 11.3 `Material Balance for problems Involving Multiple units and Re`

```
clear ;
clc;
// Example 11.3
printf('Example 11.3\n\n');
//Page no. 318
// Solution

P = 6205 ;//[lb mol/hr]
//Given
amt_F = 560 ;//[bbl]
// Fuel oil(F) analysis
C_F = 0.50 ;// [mol fraction]
H2_F = 0.47 ;//[mol fraction]
S_F = 0.03 ;//[mol fraction]
// Natural Gas(G) analysis
CH4_G = 0.96 ;//[mol fraction]
C2H2_G = 0.02 ;//[mol fraction]
CO2_G = 0.02 ;//[mol fraction]
// Analysis of air into Gas furnace(A)
```

```
O2_A = 0.21 ;//[mol fraction]
N2_A = 0.79 ;//[mol fraction]
// Analysis of air into Oil furnace(A1)
O2_A1 = 0.20  ;//[mol fraction]
N2_A1 = 0.76  ;//[mol fraction]
CO2_A1 = 0.04  ;//[mol fraction]
//Stack gas(P) analysis
N2_P = .8493  ;//[mol fraction]
O2_P = .0413  ;//[mol fraction]
SO2_P = .0010 ;// [mol fraction]
CO2_P = .1084  ;//[mol fraction]

// Degree of freedom analysis
n_un = 5;// Number of unknowns in the given problem(excluding extent of reactions)
n_ie  = 5 ;// Number of independent equations
d_o_f =  n_un-n_ie; // Number of degree of freedom
printf('Number of degree of freedom for the given system is  %i .\n',d_o_f);

// Elemental mole balance for 2N,2H,2O,S and C
// Use S balance to get F
F = P* SO2_P/S_F ;// [lb mol/hr]
//Solve other four balances to get G
//2H:  G*(2*CH4_G+C2H2_G)+F*H2_F-W*1
//2N:  A*N2_A+A1*N2_A1 = P*N2_P
//2O:  A*(O2_A)+A1*(O2_A1+CO2_A1)+G*CO2_G-W*(1/2) = P*(O2_P+CO2_P+SO2_P)
//C: G*(CH4_G+2*C2H2_G+CO2_G)+F*C_F+A1*CO2_A1 = P*CO2_P
//Solving above eqns. by matrix method[G W A A1]
a = [2*CH4_G+C2H2_G -1 0 0;0 0 N2_A N2_A1;CO2_G -.5 O2_A O2_A1+CO2_A1;CH4_G+2*C2H2_G+CO2_G 0 0 CO2_A1];// matrix of coefficients
b = [-F*H2_F;P*N2_P;P*(O2_P+CO2_P+SO2_P);(P*CO2_P-F*C_F)];// matrix of constants
x = a\b ;// matrix of solutions x(1) = G,x(2) = W,x(3) = A & x(3) = A1
```

```
G = x(1);//[lb mol/hr]
m_F = 7.91 ;// Molecular wt. of fuel oil -[lb]
Fc = (F*m_F)/(7.578*42);// Fuel gas consumed -[bbl/hr]
time = amt_F/Fc ;// Time for which available fuel gas lasts -[hr]
printf('(1) Fuel gas consumed(F) is  %.2f bbl/hr .\n',Fc);
printf('(2) Time for which available fuel gas lasts is  %.0f hr .\n',time);

// For increase in arsenic and mercury level
F_oil = Fc*42; //[gal/hr]
Em_ars2 =  (3.96 *10^(-4))/1000 ;// [lb/gal]
Em_Hg2 =  (5.92 *10^(-4))/1000 ;// [lb/gal]
ars_F = F_oil*Em_ars2 ;// Arsenic produced on burning oil -[lb]
Hg_F = F_oil*Em_Hg2 ;//Mercury produced on burning oil -[lb]
G_gas = G*359 ;//[ft^3/hr]
Em_ars1 =  (2.30 *10^(-4))/10^6 ;// [lb/ft^3]
Em_Hg1 =  (1.34 *10^(-4))/10^6 ;// [lb/ft^3]
ars_G = G_gas*Em_ars1; // Arsenic produced on burning Natural gas-[lb]
Hg_G = G_gas*Em_Hg1 ;//Mercury produced on burning Natural Gas-[lb]
in_ars = ((ars_F-ars_G)/ars_G)*100   ;//[% increase in Arsenic emission]
in_Hg = ((Hg_F-Hg_G)/Hg_G)*100  ; //[% increase in Mercury emission]
printf('(3) Increase in Arsenic emission is  %.1f %%.\n',in_ars);
printf('(4) Increase in Mercury emission is  %.1f %%.\n',in_Hg);
```

Scilab code Exa 11.4 Analysis of Sugar Recovery Process involving Multiple Serial

```
clear ;
clc;
// Example 11.4
printf('Example 11.4\n\n');
// Page no. 322
// Solution fig E11.4

//Basis : M = 1000 lb
M = 1000 ;//[lb]
//Given
F_s = 16/100 ;// Fraction of sugar in F
F_w = 25/100  ;// Fraction of  water in F
F_p = 59/100 ; // Fraction of pulp in F
D_p = 80/100 ; // Fraction of pulp in D
E_s = 13/100 ;// Fraction of sugar in E
E_p = 14/100 ;// Fraction of pulp in E
G_p = 95/100 ;// Fraction of pulp in G
H_s = 15/100 ;// Fraction of sugar in H
K_s = 40/100 ;// Fraction of sugar in K

// For crystallizer equations are
K_w = 1 - K_s ;// summation of wt. fraction is 1
K = M/K_s ;// By sugar balance -[lb]
L = K_w*K ;// By water balance -[lb]

// For evaporator equations are
H_w = 1- H_s ;//summation of wt. fraction is 1
H = K_s*K/H_s ;//By sugar balance -[lb]
J = H - K; //By overall balance -[lb]

// For screen equations are
E_w = 1 - (E_p + E_s) ; // summation of wt. fraction
    is 1
// solve E and G by simultaneous eqn. obtained by
   overall and pulp balance
a1 = [1 -1;E_p -G_p] ;// Matrix of coefficients of
```

```
unknown
b1 = [H;0] ;//Matrix of constants
x1 = a1\b1 ;// Matrix of solutions ,x1(1) = E, x1(2) = G
E = x1(1) ;//[lb]
G = x1(2) ;//[lb]
G_s = (E_s*E - H_s *H )/G ;// By sugar balance
G_w = 1 -(G_s + G_p) ;// summation of wt. fraction is 1

// For mill equations are
// solve F and D by simultaneous eqn. obtained by overall and pulp balance
a2 = [1 -1;F_p -D_p] ;// Matrix of coefficients of unknown
b2 = [E;E_p*E] ;//Matrix of constants
x2 = a2\b2 ;// Matrix of solutions ,x2(1) = F, x2(2) = D
F = x2(1) ;//[lb]
D = x2(2) ;//[lb]
D_s = (F_s*F - E_s *E )/D ;// By sugar balance
D_w = 1 -(D_s + D_p) ; // summation of wt. fraction is 1

S_rec = M/(F*F_s) ; // Fraction of sugar recovered

printf('\nFlow streams and their respective compositions.\n');
printf('\n M = %.0f lb  \n',M);
printf('  Sugar: %.2f \n',1);

printf('\n L = %.0f lb  \n',L);
printf('  Water: %.2f\n',1);

printf('\n K = %.0f lb  \n',K);
printf('  Sugar: %.2f\n',K_s);
printf('  Water: %.2f\n',K_w);

```

```
printf('\n J = %.0f lb   \n',J);
printf('   Water: %.2f \n',1);

printf('\n H = %.0f lb   \n',H);
printf('   Sugar: %.2f\n',H_s);
printf('   Water: %.2f\n',H_w);

printf('\n G = %.0f lb   \n',G);
printf('   Sugar: %.3f\n',G_s);
printf('   Water: %.3f\n',G_w);
printf('   Pulp : %.2f\n',G_p);

printf('\n E = %.0f lb   \n',E);
printf('   Sugar: %.2f\n',E_s);
printf('   Water: %.2f\n',E_w);
printf('   Pulp : %.2f\n',E_p);

printf('\n D = %.0f lb   \n',D);
printf('   Sugar: %.3f\n',D_s);
printf('   Water: %.3f\n',D_w);
printf('   Pulp : %.2f\n',D_p);

printf('\n F = %.0f lb   \n',F);
printf('   Sugar: %.2f\n',F_s);
printf('   Water: %.2f\n',F_w);
printf('   Pulp : %.2f\n',F_p);
```

Scilab code Exa 11.5 `Production of a Hormone in connected Reactor`

```
clear ;
clc;
// Example 11.5
printf('Example 11.5\n\n');
// Page no.324
// Solution
```

```

// Option 1
F = 15 ;//[L/hr]
cs_in = 10 ;//Nutrient conc. input vessel − [g nutrient/L substrate]
V1 = 100 ;// [L]
V2 = 50 ;//[L]
Yxs = 0.2 ;// [cells/g]
umax = 0.4 ;//[hrˆ − 1]
Ks = 2 ;//[g/L] − Monod constant
// Use eqn. 10.1 for balances and Monod eqns. applies to each vessel
//Cells: 0 − F/V ∗ x_out + u ∗ x_out − 0 = 0....(a)
//Nutrient: F/V ∗ cs_in − F/V ∗ cs_out + 0 − (u ∗ x_out)/(Yxs) = 0.....(b)
//From eqn.(a) F/V = u(dilution rate)...(c)
// From eqn. (b) x_out = Yxs(cs_in − cs_out)....(d)
u1 = F/V1 ;//[hrˆ − 1] //[hrˆ − 1]
cs_out = (Ks * u1/umax)/(1 - (u1/umax)) ;//Nutrient conc. output vessel − [g nutrient/L substrate]
// Find x_out by eqn. (d)
x_out = Yxs * (cs_in - cs_out) ;//[g cells / L substrate]

//Option 2
//For vessel 1
u2 = F/V2;
cs_out1 = (Ks * u2/umax)/(1 - (u2/umax)) ;//Nutrient conc. output vessel − [g nutrient/L substrate]
x_out1 = Yxs * (cs_in - cs_out1) ;//[g cells / L substrate]
// For vessel 2
// Eqn. (a) is now F/V ∗ x_out1 − F/V ∗ x_out2 + u3 ∗ x_out2 = 0...(e)
// Eqn. (b) is now F/V ∗ cs_out1 − F/V ∗ cs_out2 + (u3 ∗ x_out2)/Yxs = 0...(f)
// u3 = (umax ∗ cs_out2) / (Ks + cs_out2).. Monod
```

```
Eqn...(g)
// (e),(f) and (g) form a non - linear set of equations , solving them we get cs_out2 = 1.35 g nutrient/L substrate and x_out2  = 1.73 g cells/L
x_out2 =  1.73 ;// From eqn. (e),(f) and (g) - [g cells / L substrate]

printf('g cells/L from option 1 is %.2f.\n',x_out);
printf(' g cells/L from option 2 is %.2f.\n',x_out2);
printf(' By comparing option 1 and option 2 the respective answers are essentially the same.\n');
```

Chapter 12

Recycle Bypass Purge and the Industrial Application of Material Balance

Scilab code Exa 12.1 `A Continuous Crystallizer involving a Recycle Stream`

```
clear;
clc;
//Page No.349
// Example 12.1
printf('Example 12.1\n\n');
// Solution

//(a) fig.E12.1a
F = 10000 ;//[lb/hr]
//Given
NaOH_F = 40/100 ;//[wt. fraction]
NaOH_P1 = 95/100 ;//[wt. fraction of NaOH filter
   cake]
NaOH_P2 = (0.05 *  45)/100 ;//[wt. fraction of NaOH
   in NaOH soln.]
H2O_P2 = (0.05 *  55)/100 ;//[wt. fraction of H2O in
    NaOH soln.]
```

```
NaOH_R = 45/100;//[wt. fraction]
NaOH_G = 50/100;//[wt. fraction]
//Get P from overall NaOH balance
P = (NaOH_F *  F)/[NaOH_P1 + NaOH_P2] ;//[lb/hr]
// Get W from overall total balance
W = F-P ;// [lb/hr]

// Solve following eqn. simultaneously to get G & R
// NaOH_G *  G = F *  NaOH_F + NaOH_R *  R (NaOH balance on crystallizer)
//G = R + P (overall balance)
a = [NaOH_G -NaOH_R;1 -1] ;// matrix formed of coefficients of unknown
b = [F *  NaOH_F;P];// matrix formed by constant
x = a\b ;// matrix of solutions . x(1) = G, x(2) = R
G = x(1) ;// [lb/hr]
R = x(2) ;// [lb/hr]
printf('(a)  Flow rate of water removed by evaporator is  %.1f lb/hr\n',W);
printf('      The recycle rate of the process is %.1f lb/hr\n',R);

// (b) fig.E12.1b
//given
NaOH_H = 45/100 ;//[wt fraction]
H2O_H = 55/100 ;//[wt fraction]
// Get H & G by solving following eqn. simultaneously
//NaOH_G *  G = [NaOH_P1 + NaOH_P2] *  P  + NaOH_H *  H (NaOH balance on crystallizer)
//H2O_G *  G = H2O_P2 *  P  +  H2O_H *  H (H2O balance on crystallizer)
a1 = [NaOH_G -NaOH_H;NaOH_G -H2O_H] ;// matrix formed of coefficients of unknown
b1 = [((NaOH_P1 + NaOH_P2) *  P);(H2O_P2) *  P];// matrix formed by constant
x1 = ((a1)^-1) *  b1 ;// matrix of solutions nw_G = x1(1);H = x1(2)
```

```
nw_G1 = x1(1) ;// [lb/hr]
H = x1(2);// [lb/hr]
// By overall NaOH balance
nw_F = (NaOH_H *  H + (NaOH_P1 + NaOH_P2) *  P)/ NaOH_F ;//[lb/hr]
printf(' (b)  Total feed rate when filterate is not recycled is %.1f lb/hr\n',nw_F);
```

Scilab code Exa 12.2 `Recycle in a Process in Which Reaction Occurs`

```
clear ;
clc;
//Page No.357
// Example 12.2
printf('Example 12.2\n\n');
// Solution fig.E12.2

// Given
// Main reaction - C6H6 + 3H2 --> C6H12
F_Bz = 100 ;// Fresh benzene feed / basis - [mol]
con_Bz = .95 ;// Fraction of conversion of benzene
sp_con = .20 ;// Fraction of single pass conversion
ex_H2 = .20 ;// Fraction of exces H2 used in fresh feed
R_Bz = 22.74 ;// Benzene in Recycle stream - [mol %]
R_H2 = 78.26 ;// H2 in Recycle stream - [mol %]
TLV_Bz = 0.5 ;// TLV value of benzene -[ppm]
TLV_C6H12 = 300 ;// TLV value of cyclohexane -[ppm]
TLV_H2 = 1000 ;// TLV value of H2 -[ppm]

// Feed composition
F_H2 = F_Bz*3*(1+ex_H2) ;// H2 in Feed - [mol]
F = F_Bz + F_H2 ;// Total feed - [mol]

// Use Eqn. 12.1 to get extent of reaction -(ex_r)
```

```
ex_r = con_Bz*F_Bz/(-(-1)) ;// Extent of reaction

// get composition of P by using overall species balances
P_Bz = F_Bz -1*(ex_r) ;// Benzene in P ,by benzene balance - [mol]
P_H2 = F_H2 + -3*(ex_r) ;// H2 in P ,by H2 balance - [mol]
P_C6H12 = 0 + 1*(ex_r) ;// Cyclohexane in P ,by cyclohexane balance - [mol]
P = P_Bz + P_H2 + P_C6H12 ;// Total Product - [ mol]

// Use single pass conversion information to get recyle stream(R)
R = ((-(-ex_r))/(sp_con) - F_Bz)/(R_Bz/100) ;// Recycle stream - [mol]
R_by_F = R/F ;// Ratio of R to F

printf('Ratio of R to F is %.2f .\n',R_by_F);

TLV = (P_Bz/P)*(1/TLV_Bz) + (P_H2/P)*(1/TLV_H2) + (P_C6H12/P)*(1/TLV_C6H12) ;// TLV (environmental index)

printf('\n TLV (environmental index) is %.3f .\n',TLV);
```

Scilab code Exa 12.3 Recycle in a Process with a Reaction Occurring

```
clear;
clc;
//Page No.359
// Example 12.3
printf('Example 12.3\n\n');
// Solution fig.E12.3a and fig.E12.3b
```

```

// Given
// Main reaction - C6H12O6(d-glucose) --> C6H12O6(d-fructose)

RR = 8.33 ;// Recycle ratio
F = 100 ;// Overall feed/basis - [lb]
F_g = 0.40 ;// Fraction of glucose in overall feed
F_w = 0.60 ;// Fraction of water in overall feed
F_dash_f = 0.04 ; // Fraction of fructose in feed to reactor
P = F ;// By overall balance -[lb]
R = P/RR ;// Recycle stream - [lb]
P_w = (F_w * F)/ P ;// Fraction of water in product(P), by overall water balance
R_w = P_w ;//Fraction of water in recycle (R), since both R and P has same composition

// Mixing point 1
F_dash = F +R ;// Feed to reactor ,by total balance -[lb]
R_f = (F_dash*F_dash_f)/R ;// Fraction of fructose in recycle stream
R_g = 1 - (R_f + R_w) ;// Fraction of glucose in recycle stream
F_dash_g = (F*F_g + R*R_g)/F_dash ;// Fraction of glucose i feed to reactor

// Make glucose balance in reactor to get fraction of conversion (f_con)
f_con = ((F_dash*F_dash_g) - (R + P)*R_g)/(F_dash*F_dash_g) ;// Fraction of conversion of glucose in reactor

printf('Fraction of conversion of glucose in reactor is %.2f .\n',f_con);
```

Scilab code Exa 12.4 `A Bioreactor with Recycle`

```
clear ;
clc;
//Page No.362
// Example 12.4
printf('Example 12.4\n\n');
// Solution fig.E12.4

// Given
F = 100 ;// Overall feed/basis - [kg]
F_com = 0.10 ;// Mass fraction of component in fresh feed
F_w = 0.90 ;// Mass fraction of water in fresh feed
P_w = 0.10 ;// Mass fraction of water in product
P_com = 0.90 ;//Mass fraction of component in product
F_dash_com = 0.03 ;//Mass fraction of component in feed to reactor
W_w = 1 ;// Mass fraction of water in W(waste)
C_con = .40 ;// Fraction of conversion of component in reactor

// By analysis DOF is zero
// Take overall process as system
P = F_com*F/P_com ;//By component balance- Product - [kg]
W = F - P ;// By overall balance - waste(W)- [kg]

//Take reactor plus product recovery unit as system
// Use Eqn. 10.1 for component balance
Rw = (F*F_com - F*F_com*C_con)/C_con ;// Mass of component in recycle(R) - [kg]

```

```
// Take mixer a system
F_dash = ( F*F_com + Rw  )/F_dash_com ;// By component balance - feed to reactor(F') -[kg]
R = F_dash - F ;// Recycle(R) - By total balance -[kg]
w = Rw/R ;// Mass fraction of component in recycle(R)

printf('Recycle(R) stream-  %.0f kg \n',R);
printf(' Mass fraction of component in recycle(R)-  %.4f .\n',w);
```

Scilab code Exa 12.5 `Bypass Calculation`

```
clear ;
clc;
//Page No.367
// Example 12.5
printf('Example 12.5\n\n');
// Solution fig.E12.5

// Given
F = 100 ;// Overall feed/basis - [kg]
F_n_C5H12 = 0.80 ;// Fraction of n_C5H12 in overall feed
F_i_C5H12 = 0.20 ;// Fraction of i_C5H12in overall feed
S_i_C5H12 = 1 ;// Fraction of i_C5H12 in isopentane stream
P_n_C5H12 = .90 ;// Fraction of n_C5H12 in overall product
P_i_C5H12 = .10 ;// Fraction of i_C5H12 in overall product

// Overall Balances
```

```
P = (F*F_n_C5H12)/P_n_C5H12 ;//Product Material Balance of n_C5H12 -[kg]
S = F - P ;// Isopentane stream (S) from overall material balance - [kg]

// Balance around isopentane tower
// Let x be kg of butane free gas going to isopentane tower , y be the n-C5H12 stream leaving the isopentane tower
// Solve following Equations by Matrix method
// x = S + y - By Total materal balance
// x*F_n_C5H12 = y
a = [1 -1;F_n_C5H12 -1] ;// Matrix of coefficients of unknown
b = [S;0] ;// Matrix of constants
x = a\b ;// Matrix of solutions , x(1) = x , x(2) = y

xf = x(1)/F ;// Fraction of butane-free gas going to isopentane tower

printf('Fraction of butane-free gas going to isopentane tower is %.3f .\n',xf);
```

Scilab code Exa 12.6 Purge

```
clear;
clc;
//Page No.369
// Example 12.6
printf('Example 12.6\n\n');
// Solution fig.E12.6

// Given
F = 100 ;// Overall feed/basis - [mole]
F_H2 = 0.673 ;// Mole fraction of H2 in overall feed
```

```
F_CO = 0.325 ;// Mole fraction of i_C5H12in overall feed
F_CH4 = .002 ;// Mole fraction of CH4 in overall feed
E_CH3OH = 1 ;//  Mole fraction of CH3OH in Exit(E)
// Let x , y and z be Mole fraction of H2,CO and CH4 respectively in recycle(R) and purge(P)
z = .032 ;
CO_con = .18 ;// Fraction of conversion of CO in reactor

// Following eqn. are obtained by Materal balances
// x + y + z = 1                    eqn.(a)
// F_H2*F + F_CH4*F*2 = E*2 + P*(x + 2z)    - By H2 balance     eqn.(b)
// F_CO*F + F_CH4*F = E + P*(y + z)    -By C balance     eqn.(c)
// F_CO*F   = E + P*y     - By O balance   eqn.(d)
// F_CO*F + Ry - Ry - Py = (F_CO*F + Ry)*CO_con    - By CO balance    eqn.(e)

//By using eqn.(c) and (d)
P = F_CH4*F/z ;// Purge stream - [mole]

// Using eqn.(a) , (b) and (c)
x_plus_y = 1 - z ;// x + y
E = (F_H2*F + F_CO*F + 3*F_CH4*F - P*(x_plus_y + 3*z))/3 ;// Exit stream - [mole]

// By using eqn. (d)
y = ( F_CO*F - E )/P ;// Mole fraction of CO

// By using eqn. (a)
x = 1 - z - y ;// Mole fraction of H2

// Lastly by using eqn.(e)
R = ( F_CO*F - P*y - F_CO*F*CO_con )/(y*CO_con) ;// Recycle steam - [mole]
```

```

printf('Moles of recycle(R) per mole of feed(F) -
                        %.4f \n',R/F);
printf(' Moles of CH3OH(E) per mole of feed(F)   -
                        %.4f \n',E/F);
printf(' Moles of Purge(P) per mole of feed(F)   -
                        %.4f \n',P/F);
printf('\n Composition of Purge \n');
printf('  Component                  Mole fraction \n
   ');
printf('  H2                         %.3f \n',x);
printf('  CO                         %.3f \n',y);
printf('  CH4                        %.3f \n',z);
```

Chapter 13

Ideal Gases

Scilab code Exa 13.1 `Use of Standard Conditions to Calculate Volume from Mass`

```
clear;
clc;
// Example 13.1
printf('Example 13.1\n\n');
//Page No. 404
// Solution

m_CO2 = 40 ;// Mass of CO2-[kg]
mol_wt_CO2 = 44 ;// Molecular mass of 1kmol CO2 -[kg
    ]
mol_V = 22.42 ;// Molar of ideal gas at standard
    condition -[cubic metre/kg mol]
V_CO2 = (m_CO2 * mol_V)/(mol_wt_CO2);// volume of
    CO2-[cubic metre]
 printf('Volume occupied by 40 kg CO2 at standard
     condition is %.1f cubic metre.',V_CO2);
```

Scilab code Exa 13.2 `Calculation of R using the Standard Conditions`

```
clear ;
clc;
// Example 13.2
printf('Example 13.2\n\n');
//Page No. 405
// Solution

p =1 ;// Pressure -[atm]
V = 22415 ;// Molar valume -[cubic centimetre/g mol]
T = 273.15 ;// Temperature-[K]
R = (p*V/T);// Universal gas constant -[(cubic centimetre.atm)/(K.g mol)]
 printf('Universal gas constant is %.2f (cubic centimetre*atm)/(K*g mol). ',R);
```

Scilab code Exa 13.3 `Application of Ideal Gas Law to Calculate Volume`

```
clear ;
clc;
 //    Example 13.3
 printf('Example 13.3\n\n');
 //    Page No.406
 //    Solution

m_CO2 = 88 ;//    Mass of CO2-[lb]
mol_wt_CO2 = 44  ;//    Molecular mass of 1 lb mol CO2 -[lb]
mol_V = 359 ; //    Molar volume-[cubic feet]

 //   State 1-standard condition
P1 = 33.91 ; //    Pressure -[ft of water]
T1 = 273   ;//   Temperature-[K]

 //    State 2
P2 = 32.2  ;//    Pressure -[ft of water]
```

```
Tc = 15  ;//   Temperature-[degree C]
T2 = Tc+273  ;//  Temperature-[K]

 //   Use eqn. 13.2 to get final volume
V1 = (m_CO2 * mol_V) / (mol_wt_CO2);
V2 = (V1 * T2 * P1) / (T1 * P2);
 printf('The volume occupied 88 lb of CO2 at given condition is  %.0f cubic feet.',V2);
```

Scilab code Exa 13.4 `Calculation of Gas Density`

```
clear ;
clc;
// Example 13.4
printf('Example 13.4\n\n');
//Page No. 408
// Solution

mol_wt_N2 = 28 ;// Molecular mass of 1 kg mol N2 -[kg]
mol_V = 22.42 ;// Molar of ideal gas at standard condition -[cubic metre/kg mol]
Tc =  27 ;// Temperature-[degree C]
T = Tc + 273 ;//Temperature-[K]
P = 100 ;//Pressure-[kPa]

//Standard condition
Ps = 101.3 ;// Pressure -[kPa]
Ts = 273 ;//Temperature-[K]

V = (T *  Ps *  mol_V)/(Ts *  P) ;// Volume occupied by N2-[cubic metre]
D_N2 = mol_wt_N2/V ;// Density of N2 at given condition -[kg/cubic metre]
 printf(' Density of N2 at given condition is  %.3f
```

```
kg/cubic metre.',D_N2);
```

Scilab code Exa 13.5 `Calculation of Specific Gravity of a Gas`

```
clear ;
clc;
// Example 13.5
printf('Example 13.5\n\n');
//Page No. 409
// Solution

mol_wt_N2 = 28 ;// Molecular mass of 1 lb mol N2 -[lb]
mol_wt_air = 29 ;// Molecular mass of 1 lb mol air -[lb]
mol_V = 359 ;// Molar volume of ideal gas-[cubic feet]
//Given condition
Tf =  80 ;// Temperature-[degree F]
T = Tf + 460 ;//Temperature-[degree Rankine]
P = 745 ;//Pressure-[mm of Hg]

//Standard condition
Ps = 760 ;// Pressure -[mm of Hg]
Ts = 492 ;//Temperature-[degree Rankine]

D_air = (Ts *  P *  mol_wt_air)/(T *  Ps *  mol_V) ;// Density of air at given condition-[lb/cubic feet]
D_N2 = (Ts *  P *  mol_wt_N2)/(T *  Ps *  mol_V) ;// Density of N2 at given condition-[lb/cubic feet]
sg_N2 = D_N2/D_air ;// Specific gravity of N2 compared to air at given condition
 printf(' Specific gravity of N2 compared to air at given condition  is  %.3f .',sg_N2);
```

Scilab code Exa 13.6 `Calculation of the Partial Pressures of the Components in a G`

```
clear ;
clc;
// Example 13.6
printf('Example 13.6\n\n');
//Page No. 414
// Solution

F_gas =  1 ;// Flue gas [kg mol]
mf_CO2 = 14/100 ;// [mol fraction]
mf_O2 = 6/100 ;// [mol fraction]
mf_N2 = 80/100 ;// [mol fraction]
P = 765 ;//Pressure-[mm of Hg]
T =  400 ;// Temperature-[degree F]
p_CO2 = P * mf_CO2 ;// Partial pressure of CO2-[mm of Hg]
p_O2 = P * mf_O2 ;// Partial pressure of O2-[mm of Hg]
p_N2 = P * mf_N2 ;// Partial pressure of N2-[mm of Hg]

 printf(' Component                pi(Partial pressure-[mm of Hg]) \n');
 printf('  CO2                      %.1f mm of Hg\n ',p_CO2);
 printf(' O2                       %.1f mm of Hg\n ',p_O2);
 printf(' N2                       %.1f mm of Hg\n ',p_N2);
```

Scilab code Exa 13.7 `Material Balance for a Process Involving Combustion`

```
clear ;
clc;
// Example 13.7
printf('Example 13.7\n\n');
//Page no. 416
// Solution fig E13.7

G = 100 ;// Basis: Pyrolysis Gas-[lb mol]
ub_CO = 10/100 ;// fraction of CO left unburnt
ex_air = 40/100 ;;// fraction of excess air
m_vol = 359 ;// molar volume of gas at std. cond.-[cubic feet]
Ts = 492 ;// Standard temperature -[degree Rankine]
Ps = 29.92 ;//Standard pressure -[in. Hg]

// Analysis of entering gas of entering gas
Tf1 =  90 ;// Temperature of gas-[degree F]
T_gas = Tf1 +  460 ;//Temperature of gas-[degree Rankine]
P_gas = 35 ;//Pressure-[in. Hg]
CO2 = 6.4/100 ;// mol fraction of CO2
O2 = 0.1/100 ;// mol fraction of O2
CO = 39/100 ;// mol fraction of CO
H2 = 51.8/100 ;// mol fraction of H2
CH4 = 0.6/100 ;// mol fraction of CH4
N2 = 2.1/100 ;// mol fraction of N2

// Analysis of entering air
Tf2 = 70 ;// Temperature of air -[degree F]
T_air = Tf2 +  460 ;//Temperature of air-[degree Rankine]
P_air = 29.4 ;//Pressure of air [in. Hg]
f_N2 = 79/100 ;// mol fraction of N2
f_O2 =  21/100 ;// mol fraction of O2

// Get O2 required for combustion of CO,H2 & CH4 according to the following  equation
// CO +  1/2O2-->CO2
```

```
//H2 +  1/2O2-->H20
//CH4 +  2O2--> CO2 +  2H2O
O2r_O2 = O2 * G ;//  O2 required by O2-[lb mol]
O2r_CO = CO * G/2 ;// O2 required by CO-[lb mol]
O2r_H2 = H2 * G/2 ;// O2 required by H2-[lb mol]
O2r_CH4 = G * CH4 * 2 ;// O2 required by CH4-[lb mol]
O2r_total = O2r_O2 +  O2r_CO +  O2r_H2 +  O2r_CH4 ; // Total O2 required -[lb mol]
ex_O2 = ex_air * O2r_total ;// Excess O2-[lb mol]
total_O2 = ex_O2 +  O2r_total ;// Total amt of O2 in air -[lb mol]
total_N2 = total_O2 * (f_N2/f_O2);// Total amt of in air -[lb mol]
air = total_O2 +  total_N2 ;// Total air entering -[lb mol]

// Product analysis
P_CO = ub_CO * CO * G ;//Unburnt CO in P-[lb mol]
//Element balance of 2N
P_N2 = N2 * G +   total_N2 ;//  N2 in P-[lb mol]
//Element balance of C
P_CO2 =  (CO2 +  CO +  CH4) * G - 1 * P_CO;//CO2 in P-[lb mol]
// Element balance of 2H
P_H2O = (H2 +  2 * CH4) * G ;// H2 in P-[lb mol]
//  Element balance of 2O
P_O2 = (CO2 +  O2 +  0.5 * CO) * G +  total_O2 - P_CO2-0.5 * (P_H2O +  P_CO);// O2 in P-[lb mol]
P = P_CO +  P_N2 +  P_CO2 +  P_H2O +  P_O2 ;// Product -[lb mol]
Tf3 =  400  ;// Temperature of product -[degree F]
T_prod = Tf3 +  460 ;//Temperature of product -[degree Rankine]
P_prod =  35 ;// Pressure of product -[in.Hg]
V_gas = (G * m_vol * T_gas * Ps)/(Ts * P_gas);
V_air = (air * m_vol * T_air * Ps)/(Ts * P_air);
V_prod = (P * m_vol * T_prod * Ps)/(Ts * P_prod);
```

```
air_ft3 = V_air/V_gas ;//Air supplied per ft^3 of gas entered -[cubic feet]
P_ft3 =  V_prod/V_gas ;//Product gas produced per ft^3 of gas entered -[cubic feet]

printf(' Air supplied per ft^3 of gas entered %.2f cubic feet.\n ',air_ft3);
printf(' Product gas produced per ft^3 of gas entered %.2f cubic feet.\n',P_ft3);
```

Scilab code Exa 13.8 `Material Balance without Reaction`

```
clear ;
clc;
//Page No. 419
// Example 13.8
printf('Example 13.8\n\n');
// Solution fig E13.8

T1c = 15 ;// Temperature of F & P -[degree C]
T1 =  273  +  T1c ;// Temperature of F & P -[K]
P1 =  105 ;// Pressure of F & P -[kPa]
// F analysis
F_CO2 = 1.2/100 ;// Volume fraction
F_odr = 98.8/100 ;// Volume fraction

// P analysis
P_CO2 = 3.4/100 ;// Volume fraction
P_odr = 96.6/100 ;// Volume fraction

Tc_CO2 =  7 ;//Temperature CO2 -[degree C]
T_CO2 =  273 +  Tc_CO2 ;// Temperature CO2 -[K]
P_CO2 =  131 ;// Pressure of CO2 -[kPa]
CO2 = 0.0917 ;// Volume flow rate of CO2-[cubic metre/min]
```

```
// Convert given volume flow rate of CO2 at temperature of F & P
nw_CO2 = (CO2 *  T1 *  P_CO2)/(T_CO2 *  P1) ;// volume flow rate of CO2 at temperature of F & P-[cubic metre]
// Solve P & F by following eqns. obtained by component balance of CO2 and total balance
// F(F_odr) = P(P_odr) - others balance
// F +  nw_CO2 = P - Total balance
// Solving by matrix method
a = [F_odr -P_odr;1 -1];// Matrix formed by coefficients of unknown
b = [0;-nw_CO2] ;// Matrix formed by constants
x = a\b ;// matrix of solution , x(1) = F;x(2) = P
F = x(1) ;//Volume flow rate of entering gas-[cubic metre/min]
P = x(2) ;//Volume flow rate of product [cubic metre/min]
printf('Volume flow rate of entering gas is %.2f cubic metre/min',F);
```

Chapter 14

Real Gases Compressiblity

Scilab code Exa 14.1 `Use of Compressiblity Factor to Calculate a Specific Volume`

```
clear ;
clc;
// Example 14.1
printf('Example 14.1\n\n');
//Page No. 442
// Solution

T1f = 125 ;// Temperature of NH3 -[degree F]
T1 =  460 +T1f ;// Temperature NH3 -[degree Rankine]
Pg =  292 ;// Pressure of NH3 -[psig]
Pa =  Pg+14.7 ;//Pressure of NH3 -[psia]
R = 10.73 ;//Universal gas constant -[(psia*cubic
   feet)/(lb mol*R)]
mw_NH3 = 17 ;// Molecular wt. 1 lb mol NH3-[lb]
n = 1/17 ;//[mol]
V_tank = 120 ;// Volume of tank-[cubic feet]
// Ideal V
V_id = (n*R*T1)/Pa ;// Specific volume of NH3
   treating it ideal gas-[cubic feet/lb]

//From appendix D
```

```
Tc = 729.9 ;//[degree R]
Pc = 1636 ;//[psia]
Tr =  T1/Tc;
Pr =  Pa/Pc;
// Using Tr and Pr we get z = 0.855 from Nelson and Obert chart
z_real = 0.855;
z_ideal = 1;
V_real =  V_id*z_real/z_ideal;// Specific volume of NH3 treating it real gas-[cubic feet/lb]
NH3 = V_tank/V_real ;// Actual amt. of NH3 in tank-[lb]
printf('Actual amt. of NH3 in tank is %.0f lb. Therefore , boss is wrong.',NH3);
```

Scilab code Exa 14.2 Use of Compressiblity Factor to Calculate Pressure

```
clear ;
clc;
// Example 14.2
printf('Example 14.2\n\n');
//Page No. 444
// Solution

//From appendix D
Tc = 154.4 ;//[K]
Pc1 = 49.7 ;// [atm]
Pc = 101.3 * Pc1;//[kPa]

T_O21 = -25 ;// Temperature-[degree C]
T_O2 =  273+T_O21;//Temperature -[K]
R = 8.134 ;// gas constant-[(cubic metre * kPa)/(kg mol * K)]
V_tank = 0.0284 ;// Volume of tank-[cubic metre]
mol_O2 = 32 ;// Kmol. wt. of O2-[kg]
```

```
m_O2 =  3.5 ;// Mass of liquid O2-[kg]
V_sp = V_tank * mol_O2/m_O2 ;// Specific molar
   volume-[m^3/kg]
Vc = R * Tc/Pc ;// [cubic metre/kg mol]
Vr = V_sp/Vc;
Tr = T_O2/Tc;

// Now use Vr and Tr to get Pr from Nelson and Obert
    chart ,Pr = 1.43
Pr = 1.43 ;// [kPa]
P_O2 = Pr * Pc ;// The pressure in the tank -[kPa]
printf('The pressure in the tank is %.2f  kPa.',P_O2
   );
```

Scilab code Exa 14.3 `Calculation of Properties of Real Gas Mixture`

```
clear ;
clc;
// Example 14.3
printf('Example 14.3\n\n');
//Page No. 448
// Solution

// Given
Tc = 100 ;// Temperature  -[degree C]
T   =  273 +Tc ;// Temperature -[K]
P   =  90 ;// Pressure [atm]
R = 82.06 ;// gas constant -[(cubic centimetre * atm)
   /(g mol * K)]
Y_CH4  = 20/100 ;// [mole fraction]
Y_C2H4 = 30/100 ;// [mole fraction]
Y_N2   =  50/100 ;//[mole fraction]

//Additional information from appendix D
Tc_CH4 = 191 ;//[K]
```

```
Pc_CH4 = 45.8 ;// [atm]
Tc_C2H4 = 283 ;//[K]
Pc_C2H4 = 50.5 ;// [atm]
Tc_N2 = 126 ;//[K]
Pc_N2 = 33.5 ;// [atm]

//(a)-Ideal gas law
V_sp1 = R * T/P ;// Molar volume-[cubic centimetre/g mol]
printf('(a) The volume per mole of mixture by ideal gas law is %.1f cubic centimetre/g mol.\n',V_sp1);

//(b)
Pc_mix = Pc_CH4 * Y_CH4+Pc_C2H4 * Y_C2H4+Pc_N2 * Y_N2;// [atm]
Tc_mix = Tc_CH4 * Y_CH4+Tc_C2H4 * Y_C2H4+Tc_N2 * Y_N2 ;// [K]
Pr_mix = P/Pc_mix;
Tr_mix = T/Tc_mix;
// With 2 parameters(Pr_mix and Tr_mix) , you can find from figure 14.4b that z * Tr_mix = 1.91
z = 1.91/Tr_mix;
V_sp2 = z * R * T/P ;// Molar volume-[cubic centimetre/g mol]
printf('\n(b) The volume per mole of mixture by treating it to be real gas is %.1f cubic centimetre/g mol.',V_sp2);
```

Chapter 15

Real Gases Equations of State

Scilab code Exa 15.1 `Application of Van der Walls Equation to Calculate a Temperat`

```
clear ;
clc;
// Example 15.1
printf('Example 15.1\n\n');
//Page No. 464
// Solution

// Given
R = 82.06 ;// gas constant -[(cm^3 *atm)/(g mol *K)]
a = 9.24 *10^(6) ;//(atm) *(cm^3/g mol)^2
b = 90.7 ;// (cm^3)/(g mol)
m_C3H8 = 22.7 ;// Mass of propane -[kg]
mw_C3H8 =  44 ;// Mol. wt. of  1kmol propane -[kg]
V = 0.15 *10^(6) ;// Volume of cylinder -[cm^3]
pg = 4790 ;// Gauge pressure -[kPa]
P = (pg +101.3)/101.3 ;// Pressure absolute -[atm abs]
n = (m_C3H8/mw_C3H8) *10^3 ;// Moles of propane
// Get T using Van der Waal's eqn.
T = ((P +((n^(2) *a/(V^(2))))) *(V-n *b))/(R *n) ;// Temperature of propane -[K]
```

```
printf('\nTemperature of propane is %.0f K.',T);
```

Scilab code Exa 15.2 `Solution of van der Waals Equation for V`

```
clear ;
clc;
// Example 15.2
printf('Example 15.2\n\n');
//Page No. 465
// Solution

// Given
R = 10.73 ;// gas constant -[(cubic feet *psia)/(lb mol *R)]
a = 3.49 * 10^4 ;//(psia) *(cubic feet/lb mol)^2
b = 1.45 ;// (cubic feet)/(lb mol)
p = 679.7 ;// Pressure -[ psia]
n = 1.136 ;// Amount of mole -[lb mol]
T = 683 ;// Temperature - [degree R]

// Get V using Van der Waal's eqn.
deff('[y]=g(V)','y=(V^3) -(((p*n*b) + (n*R*T))/p)*V^2 + ((n^2)*a*V/p) - ((n^3)*a*b)/p');
V=fsolve(b,g) ;// Volume of final solution (volume of vessel) [cubic feet]

printf('\nVolume of final solution (volume of vessel) is %.0f cubic feet.',V);
```

Chapter 16

Single Component Two Phase Systems

Scilab code Exa 16.1 `Vaporization of Metals for Thin Film Deposition`

```
clear ;
clc;
// Example 16.1
printf('Example 16.1\n\n');
//Page no. 486
// Solution Fig E16.1

// Given
Tc =  972 ;//[degree C]
T = 273+Tc ;//[K]
A = 8.799;
B = 1.615 * 10^4;
C = 0;
mw = 26.98;
// Use Antoine eqn. to get vapour pressure at 972
   degree C
vP = exp(A-(B/(C+T))) ;// vapour pressure at 972
   degree C-[mm Hg]
P = vP * 101.325/760 ;//[kPa]
```

```
// Use rate of vapourization(m) by given formula
m = 0.437 * (P * (mw^.5)/(T^0.5)) ;// Vapourization rate at 972 degree C-[g/(square centimetre * s)]
printf('\n Vapourization rate at 972 degree C is %.1e g/(square centimetre)(s).',m);
```

Scilab code Exa 16.2 `Interpolating in Steam Tables`

```
clear ;
clc;
// Example 16.2
printf('Example 16.2\n\n');
//Page no. 491
// Solution


//(a)
// Given
// get essential data from steam table
Ta = [310,315] ;//Temperature data from steam table-[K]
pa = [6.230,8.143] ;// Pressure data from steam table - [kPa]
pfa = interpln([Ta;pa],312) ;// Pressure at 312 K - [kPa]
printf('(a) Saturation pressure of water at 312 K is %.1f kPa.\n',pfa );

//(b)

//For initial condition get specific volume
  // Double interpolation
     //first interpolation
      // at 600 degree F ,data from steam table
       pb_600 = [90,95] ;// Pressure data - [psia]
```

```
       vb_600= [6.916,6.547] ;// specifc volume data - [(cubic feet/lb)]
       v_600 = interpln([pb_600;vb_600],92);// specifc volume at 92 psia and 600 degree F - [(cubic feet/lb)]

      //at 700 degree F,data from steam table
       pb_700 = [90,95]  ;// Pressure data - [psia]
       vb_700 = [7.599,7.195];// specifc volume data - [(cubic feet/lb)]
       v_700 = interpln([pb_700;vb_700],92) ;// specifc volume at 92 psia and 700 degree F - [(cubic feet/lb)]

     // second interpolation 92 psia ,data from steam table
       Tb_92 = [600,700];// Temperature data from steam table - [degree F]
       vb_92 = [v_600,v_700];// specifc volume data - [(cubic feet/lb)]
       v_640_92 = interpln([ Tb_92;vb_92],640);// specifc volume at 92 psia and 640 degree F - [(cubic feet/lb)]



//For final condition get specific volume
  // Double interpolation
     //first interpolation
      // at 450 degree F,data from steam table
           pc_450 = [50,55]  ;// Pressure data - [psia]
           vc_450 = [10.69,9.703];// specifc volume data - [(cubic feet/lb)]
           v_450 = interpln([pc_450;vc_450],52);// specifc volume at 52 psia and 450 degree F  - [(cubic feet/lb)]

```

```
       //at 500 degree F,data from steam table
        pc_500 = [50,55] ; // Pressure data - [psia]
        vc_500 = [11.30,10.26];// specifc volume data - [(cubic feet/lb)]
        v_500 = interpln([pc_500;vc_500],52);// specifc volume at 52 psia and 500 degree F - [(cubic feet/lb)]

      // second interpolation 52 psia ,data from steam table
        Tc_52 = [450,500];// Temperature data from steam table - [degree F]
        vc_52 = [v_450,v_500];// specifc volume data - [(cubic feet/lb)]
        v_480_52 = interpln([ Tc_52;vc_52],480) ;// specifc volume at 52 psia and 480 degree F - [(cubic feet/lb)]

del_v = v_480_52 - v_640_92 ;// Change in specific volume - [(cubic feet/lb)]
printf(' (b) Change in specific volume between initial and final condition is %.2f (cubic feet/lb).\n',del_v );
```

Scilab code Exa 16.3 `Extrapolation of Vapour Pressure data`

```
clear ;
clc;
// Example 16.3
printf('Example 16.3\n\n');
//Page no. 494
// Solution

//Given
T1 = 110 ;// Temperature of chlorobenzene - [degree
```

```
   C]
T1F = (9*T1)/(5) + 32 ;// Temperature of
   chlorobenzene - [degree F]
P1 = 400 ;//Pressure of chlorobenzene - [mm of Hg]
P1_psia = P1*14.7/760 ;//Pressure of chlorobenzene -
    [psia]
T2 = 205 ;// Temperature of chlorobenzene - [degree
   C]
T2F = (9*T2)/(5) + 32 ;// Temperature of
   chlorobenzene - [degree F]
P2 = 5 ;//Pressure of chlorobenzene - [atm]
P2_psia = P2*14.7 ;//Pressure of chlorobenzene - [
   psia]

// Data from steam table
x1 = [.9487,3.72,11.525,29.8,67,247,680,1543,3094];
y1 = [100,150,200,250,300,400,500,600,700];

x2 = [P1_psia,P2_psia];
y2 = [T1F,T2F];

// Cox chart using given and steam table data
plot2d(”ln”,x1,y1,5);
//plot2d(”ln”,x1,y1)
xgrid(3);
//plot2d(”ln”,x2,y2)
plot2d(”ln”,x2,y2,2);
xgrid(3);
legend(”Water”,”Chlorobenzene”);
plot2d(”ln”,x1,y1,-9);
plot2d(”ln”,x2,y2,-9);
title(’Figure E16.3 Cox chart for the problem’);
xlabel(’Vapour Pressure , psia (log(10) scale)’);
ylabel(’Temperature, degree F(special scale)’);

// Estiimate vapour pressure of chlorobenzene from
   cox chart prepared, it is
vp1 = 150 ;// vapour pressure of chlorobenzene from
```

```
    cox chart prepared at 245 degree C
vp2 = 700 ;// vapour pressure of chlorobenzene from cox chart prepared at 359 degree C

printf('Temperature            Estimated vapour pressure of chlorobenzene from cox chart\n');
printf('\n 245 degree C            %i psia\n',vp1);
printf(' 359 degree C            %i psia',vp2);
```

Scilab code Exa 16.4 `Solvent selection based on OSHA PEL Limits and Potential Haza`

```
clear ;
clc;
// Example 16.4
printf('Example 16.4\n\n');
//Page no. 495
// Solution

// Given
OP_Et = 400 ;//OSHA PEL of ethyl acetate -[ppm by volume]
OP_Mek = 200 ;//OSHA PEL of  Methyl ethyl ketone [ppm by volume]
OP_Nba = 1.3 ;//OSHA PEL of n-butyl acetate [ppm by volume]

vp_Et =   96.9 ;// Vapour pressure of ethyl acetate obtained from CD-[mm of Hg]
vp_Mek =   94.8 ;// Vapour pressure of Methyl ethyl ketone obtained from CD-[mm of Hg]
vp_Nba =   20 ;// Vapour pressure of n-butyl acetate obtained from Perry -[mm of Hg]

// Combined hazard criterion
Chz_Et = vp_Et/OP_Et ;//  Combined hazard criterion
```

```
 of ethyl acetate
Chz_Mek = vp_Mek/OP_Mek ;//  Combined hazard criterion of Methyl ethyl ketone
Chz_Nba = vp_Nba/OP_Nba ;//  Combined hazard criterion of n-butyl acetate

printf('\nCombined hazard criterion of solvents in increasing order are :\n');
printf('\nEthyl acetate :            %.2f',Chz_Et);
printf('\nMethyl ethyl ketone :      %.2f',Chz_Mek);
printf('\nn-butyl acetate :          %.2f',Chz_Nba);
```

Chapter 17

Two Phase Gas Liquid Systems Saturation Condensation and Vaporization

Scilab code Exa 17.1 `Calculation of Dew Point of the Products of Combustion`

```
clear ;
clc;
// Example 17.1
printf('Example 17.1\n');
// Page no. 511
// Solution

// Basis : F =  1 mol
F = 1 ;//H2C2O4- [mol]
ex_O2 =  248 ;//Excess air- [%]
f_C = 65/100 ;// Fraction of Carbon which convert to
    CO2
P = 101.3 ;// Atmospheric pressure -[kPa]

// H2C2O4 + 0.5*O2-->2*CO2  +  H2O
// H2C2O4 -->2*CO  +  H2O + 0.5*O2
O2_req = F*0.5 ;// O2 required by the above reaction
```

```
-[mol]
O2_in = (1 + ex_O2*F/100)*0.5 ;// Mol. of O2 entering

// Use Elemental balance moles of species in output
n_CO2 = f_C*2 ;// [mol]
n_H2O = (2*F)/2 ;// From 2H balance -[mol]
n_N2 = ((O2_in*0.79)/(0.21)) ;// From 2N balance -[mol]
n_CO = 2-n_CO2 ;// From C balance -[mol]
n_O2 = ((4 + O2_in*2)-(n_H2O + n_CO + 2*n_CO2))/2 ; // From O2 balance -[mol]
total_mol = n_CO2 + n_H2O + n_N2 + n_CO + n_O2 ;// Total moles in output stream -[mol]
y_H2O = n_H2O/total_mol ;// Mole fraction of H2O
pp_H2O = y_H2O*P ;// Partial pressure of H2O-[kPa]

printf('\nPartial pressure of H2O %.2f kPa.',pp_H2O);
printf('\nUse partial pressure of H2O to get dew point temperature T from steam table: T = 316.5 K');
```

Scilab code Exa 17.2 Condensation of Benzene from a Vapour Recovery Unit

```
clear ;
clc;
// Example 17.2
printf('Example 17.2\n');
//Page no. 517
// Solution Fig E17.2b

gas = 1 ;// Entering gas -[g mol]
T = 26 ;// Temperature (for isothermal process) -[degree C]
```

```
// From fig. its clear that at 26 C saturation pressure is at point A
// Get vapour pressure of benzene from Perry handbook or CD,it is
vp =  99.7 ;// vapour pressure of benzene at 26 C-[mm of Hg]

// Analysis of entering gas
f_C6H6 = 0.018 ;// Mol fraction of benzene
f_air  =  0.982 ;// Mol fraction of air
mol_C6H6 = 0.018*gas ;// Moles  of benzene -[g mol]
mol_air  =  0.982*gas ;// Moles  of air -[g mol]

// Analysis of exit gas
C6H6_rec =  95/100 ;// Fraction of benzene recovered
C6H6_out  = 1-C6H6_rec ;//Fraction of benzene in exit stream
C6H6_out = mol_C6H6*C6H6_out ;//Moles of benzene in exit stream -[g mol]
air_out = mol_air ;//Moles of air in exit stream -[g mol]
total_mol = C6H6_out+air_out ;// Total moles in exit stream
y_C6H6_out = C6H6_out/total_mol ;// Mole fraction of benzene in exit
P = vp/y_C6H6_out ;// Pressure total of exit

printf('\n Pressure total at exit of compressor %.2e mm of Hg.',P);
```

Scilab code Exa 17.3 Smokestack Emissions and Pollution

```
clear ;
clc;
// Example 17.3
```

```
printf('Example 17.3\n');
// Page no. 519
// Solution Fig E17.3b

// Given
// coal analysis from handbook
ex_air = .4 ;// Fraction of excess air required
w_C = 12 ;// Mol. wt. of C-[g]
mol_C = 71/w_C ;//[kg mol]
w_H2 = 2.016 ;// Mol. wt. of H2 - [g]
mol_H2 = 5.6/w_H2;
air_O2 = 0.21;// Fraction of O2 in air
air_N2 = 0.79;// Fraction of N2 in air

// Natural Gas
// Basis = 1 kg mol C
// CH4 + 2O2 --> CO2 + 2H2O .... Eqn. (a)
CO2_1 = 1 ;//  By Eqn. (a) CO2 produced -[kg mol]
H2O_1 = 2 ;// By Eqn. (a) H2O produced -[kg mol]
Req_O2_1 = 2 ;// By Eqn. (a) -[kg mol]
ex_O2_1 = Req_O2_1*ex_air  ;// Excess O2 required -[kg mol]
O2_1 = Req_O2_1 + ex_O2_1 ;// Total O2 required - [kg mol]
N2_1 = O2_1*(air_N2/air_O2) ;//Total N2 required - [kg mol]
Total_1 = CO2_1 + H2O_1 + N2_1 + ex_O2_1 ;// Total gas produced- [kg mol]

// Coal
// C + O2 --> CO2  ..eqn (b)
// H2 + 1/2(O2) --> H2O.... eqn (c)
CO2_2 = 1 ;//  By Eqn. (a) CO2 produced -[kg mol]
H2O_2 = mol_H2/mol_C ;// By Eqn. (a) H2O produced -[kg mol]
Req_O2_2 = 1 + (mol_H2/mol_C)*(1/2) ;// By Eqn. (b) and (c) -[kg mol]
ex_O2_2 = Req_O2_2*ex_air  ;// Excess O2 required -[
```

```
kg mol]
O2_2 = Req_O2_2 + ex_O2_2; // Total O2 required - [kg mol]
N2_2 = O2_2*(air_N2/air_O2); //Total N2 required - [kg mol]
Total_2 = CO2_2 + H2O_2 + N2_2 + ex_O2_2 ;// Total gas produced- [kg mol]

// Let P (total pressure) = 100 kPa
P = 100 ;// Total pressure -[kPa]
p1 = P*(H2O_1/Total_1) ;// Partial pressure of water vapour in natural gas - [kPa]
Eq_T1 = 52.5  ;// Equivalent temperature -[degree C]
p2 = P*(H2O_2/Total_2) ;// Partial pressure of water vapour in coal - [kPa]
Eq_T2 = 35  ;// Equivalent temperature -[degree C]
printf('                                        Natural gas                      Coal\n')
printf('-------------------------------------------------\n')
printf('Partial pressure:                    %.1f kPa                      %.1f kPa\n',p1,p2 ) ;
printf('Equivalent temperature:              %.1f C                        %.1f C\n',Eq_T1,Eq_T2 );
```

Scilab code Exa 17.4 `Material Balance involving Condensation`

```
clear ;
clc;
// Example 17.4
printf('Example 17.4\n\n');
//Page no. 522
// Solution Fig E17.4
```

```
F = 30 ;// Volume of initial gas-[m^3]
P_F =  98.6 ;// Pressure of gas-[kPa]
T_F =  273+100 ;// Temperature of gas-[K]
P_p = 109 ;//[kPa]
T_p = 14+273 ;// Temperature of gas-[K]
R = 8.314 ;// [(kPa*m^3)/(k mol*K)]
// Additional condition
vpW_30 = 4.24 ;//Vapour pressure -[kPa]
vpW_14 = 1.60 ;//Vapour pressure -[kPa]
n_F = (P_F*F)/(R*T_F) ;// Number of moles in F

// Material balance to calculate  P & W
P = (n_F*((P_F-vpW_30)/P_F))/((P_p-vpW_14)/P_p) ;//
    P from mat. bal. of air -[kg mol]
W = (n_F*(vpW_30/P_F))- P*(vpW_14/P_p); // W from
    mat. bal. of water -[kg mol]
iW = n_F*(vpW_30/P_F) ;// Initial amount of water -[
    kg mol]
fr_con = W/iW ;//Fraction of water condenseed

printf('\n Fraction of water condenseed %.3f.',
    fr_con);
```

Scilab code Exa 17.5 `Vaporization to Saturate Dry air`

```
clear;
clc;
// Example 17.5
printf('Example 17.5\n');
//Page no. 527
// Solution Fig E17.5

P =  100 ;// Pressure of air -[kPa]
T =  20 + 273 ;// Temperature of air -[K]
```

```
R = 8.314 ;// [(kPa*m^3)/(k mol*K)]
EOH =  6 ;// Amount of ethyl alcohol to evaporate-[kg]
mw_EOH = 46.07 ;// Mol.wt. of 1 k mol ethyl alcohol -[kg]
// Additional data needed
vp_EOH = 5.93 ;// Partial pressure of alcohol at 20 C-[kPa]
vp_air = P-vp_EOH ;// Partial pressure of air at 20 C-[kPa]
n_EOH  = EOH/mw_EOH ;//Moles of ethyl alcohol -[kg mol]
n_air = (n_EOH*vp_air)/vp_EOH ;// Moles of air -[kg mol]
V_air = n_air*R*T/P ;// Volume of air required

printf('\n Volume of air required to evaporate 6 kg of ethyl alcohol is %.1f cubic metre . \n',V_air);
```

Scilab code Exa 17.6 `Vaporization of a Hazardous Component of an Oil Slick`

```
clear ;
clc;
// Example 17.6
printf('Example 17.6\n\n');
//Page no. 529
// Solution

P =  760 ;// Pressure -[ mm of Hg]
// Get vapour pressure of n-heptane from Perry, 40 mm of Hg
vp = 40 ;// vapour pressure of n-heptane-[mm of Hg]

// Use the 2nd relation given in problem to find K
```

```
K = 10^((log10(vp/P)-0.16)/1.25) ;

// Get t using the 1st relation in the question
// For t_half
x = 0.5 ;// mole fraction after  t_half
x0 = 1 ;// initial mole fraction
t_half = (log(x/x0))/(-K);// Time required to reduce the concentration to one-half -[min]
printf('Time required to reduce the concentration to one-half is %.1f min. \n',t_half);
```

Chapter 18

Two Phase Gas Liquid Systems Partial Saturation and Humidity

Scilab code Exa 18.1 `Application of Relative Humidity to Calculate the Dew Point`

```
clear ;
clc;
// Example 18.1
printf('Example 18.1\n\n');
//Page no.539
// Solution

V = 1 ;// Volume of water vapour -[cubic metre]
rel_h =  43 ;// relative humidity -[%]
vp_H2O = 1.61 ;// vapour pressure of water at 94 F-[
   in. of Hg]
P_H2O = vp_H2O*(rel_h/100) ;// Pressure of water
   vapour in air -[in. of Hg]
P =  29.92 ;// [in of Hg]
T = 94+460 ;// Temperature -[Rankine]
Ts =  492 ;//Temperature std. -[Rankine]
mw_H2O = 18 ;// molecular mass of water -[lb]
```

```
H2O = (5280^3*Ts*P_H2O*mw_H2O)/(T*P*359) ;//mass of H2O-[lb]
// The dew point is temperature at which water vapour in air first condense ,i.e at realative humidity 100 %, therefore
psat_H2O = P_H2O ;// Saturation pressure of H2O -[in. of Hg]

printf('\nSaturation pressure of H2O %.3f in. of Hg\n',psat_H2O);
printf('Use saturation  pressure of H2O to get dew point temperature T from steam table: T is about 68-69 F.');
```

Scilab code Exa 18.2 Calculation involving Various Partial Saturation Terms

```
clear ;
clc;
// Example 18.2
printf('Example 18.2\n\n');
//Page no. 541
// Solution

// Data from steam table
psat_H2O = 31.8 ;// Saturation pressure -[mm of Hg]

//(c)
H =  .0055 ;// Humidity
mw_H2O =  18 ;// Molecular wt.  of water-[lb]
mw_air =  29 ;// Molecular wt. of air -[lb]
P = 750 ;// Pressure total -[mm of Hg]
p_H2O = ((H*mw_air*P)/mw_H2O)/(1+(H*mw_air/mw_H2O)) ;// Partial pressure of water vapour in air -[mm of Hg]

```

```
//(a)
rel_H = (p_H2O/psat_H2O)*100  ;// relative humidity -[%]

//(b)
mol_H = (p_H2O)/(P-p_H2O) ;// Molal humidity

printf('\n(a)Relative humidity is %.0f%% .\n',rel_H);
printf('\n(b)Molal humidity is %.1e \n',mol_H);
printf('\n(c)Partial pressure of water vapour in air is %.1f  mm of Hg.\n',p_H2O);
```

Scilab code Exa 18.3 `Dehydration of Moist air`

```
clear ;
clc;
// Example 18.3
printf('Example 18.3\n\n');
//Page No. 544
// Solution fig.E18.3

V_BDA  = 1000 ;// Volume of bone dry air(BDA) at 20 C & 108.0 kPa
mol_V =  22.4 ;// Molar volume of gas at standard condition -[m^3]
T = 20+273 ;// Temperature of BDA-[K]
P = 108.0 ;//Pressure of BDA-[kPa]
Ts = 273 ;// Standard temperature -[K]
Ps = 101.3 ;//Standard pressure -[kPa]
W = 0.93 ;// [kg]
mw_W =  18 ;// mol. wt. of 1kmol water  -[kg]
mol_W = W/mw_W ;// amount of water vapour(W) -[kg mol]
mol_BDA = (V_BDA*Ts*P)/(T*Ps*mol_V) ;//  amount of
```

```
BDA-[kg mol]
p_H2O =  (mol_W/(mol_W+mol_BDA))*P ;// Partial pressure of H2O-[kPa]

// Get vapour pressure for water at 15 C , namely 1.70 kPa
psat_H2O  =  1.70 ;//vapour pressure for water at 15 C-[kPa]
rel_H = (p_H2O/psat_H2O) ;//Fractional relative humidity -[]
printf('\n(a)Fractional relative humidity of original air was %.3f .\n',rel_H);
```

Scilab code Exa 18.4 `Humidification of Air`

```
clear ;
clc;
// Example 18.4
printf('Example 18.4\n\n');
//Page no.545
// Solution fig.E18.4

F  = 1000 ;// Volume of entering moist air at 22 C & 101.0 kPa
mol_V =  22.4 ;// Molar volume of gas at standard condition -[m^3]
T_in = 22+273 ;// Temperature of entering moist air -[K]
P_in = 101.0 ;//Pressure of entering moist air -[kPa]
dp_in = 11+273 ;// Dew point of entering air -[K]
Ts = 273 ;// Standard temperature -[K]
Ps = 101.3 ;//Standard pressure -[kPa]
T_out = 58+273 ;// Temperature of exiting moist air -[K]
```

```
P_out = 98 ;//Pressure of exiting moist air -[kPa]

// Additional vapour pressure data
psat_in = 1.31 ;//Vapour pressure of entering moist air -[kPa]
psat_out = 18.14 ;// Vapour pressure of exiting moist air -[kPa]
pBDA_in = P_in-psat_in ;// Pressure of entering dry air - [kPa]
pBDA_out  =  P_out - psat_out ;// Pressure of exiting dry air - [kPa]

mol_F = (F*P_in*Ts)/(Ps*T_in*mol_V) ;// Moles of moist air entering -[kg mol]

//Material Balances to get  W
mol_P = (mol_F*(pBDA_in/P_in))/(pBDA_out/P_out); // BDA balance- [kg mol]
mol_W = mol_P-mol_F ;// Total balance -[kg mol]

// To calculate kg of wet air entering
mw_BDA = 29 ;// Mol. wt. of dry air
mw_H2O = 18 ;// Mol. wt. of water vapour
m_BDA = (mol_F*pBDA_in/P_in)*mw_BDA ;// Mass of dry air entering -[kg]
m_H2O = (mol_F*psat_in/P_in)*mw_H2O ;// Mass of water vapour entering -[kg]
wa_in = m_BDA+m_H2O ;//Total wet air entering -[kg]
H2O_ad = mol_W*mw_H2O/wa_in ;//Water added to each kg of wet air entering the process -[kg]

printf('Water added to each kg of wet air entering the process is %.3f kg.\n',H2O_ad);
```

Scilab code Exa 18.5 `Condensation of Water from air`

```
clear ;
clc;
// Example 18.5
printf('Example 18.5\n');
//Page No.547
// Solution fig.E18.5

// Given data
//Basis: F = 29.76 lb mol
F  = 29.76 ;// amount of entering moist air -[lb mol]
F_rh = 90/100 ;// Relative humidity
T_in = 100 + 460 ;// Temperature of entering moist air -[Rankine]
P_in = 29.76 ;//Pressure of entering moist air -[in. of Hg]
psat_in =  1.93 ;// Saturation pressure from steam table -[in. of Hg]
T_out = 120 + 460 ;// Temperature of exiting dry air -[Rankine]
P_out = 131.7 ;//Pressure of exiting dry air -[in. of Hg]
psat_out =  3.45 ;// Saturation pressure from steam table -[in. of Hg]
mol_V =  22.4 ;// Molar volume of gas at standard condition -[m^3]
mw_H2O  =  18.02 ;// Mol. wt. of water -[lb]
mw_air  =  29 ;// Mol. wt. of air -[lb]
p_H2O_in = F_rh*psat_in ;// Partial pressure of water vapour at inlet --[in. of Hg]
p_air_in = P_in-p_H2O_in ;// Partial pressure of air at inlet --[in. of Hg]

// Assume condensation takes place , therefore output gas P is saturated ,
P_rh =  1;// Relative humidity of output gas
p_H2O_out = P_rh*psat_out ;// Partial pressure of water vapour at outlet --[in. of Hg]
```

```
p_air_out = P_out-p_H2O_out ;// Partial pressure of air at outlet --[in. of Hg]

// Get W and P from  balance of air and water
P = (p_air_in*F/P_in)/(p_air_out/P_out) ;// From air balance -[ lb mol]
W = (p_H2O_in*F/P_in)-(P*p_H2O_out/P_out);// From water balance -[lb mol]
W_ton = (W*mw_H2O*2000)/(p_air_in*mw_air) ;// Moles of water condenses per ton dry air -[lb mol]
W_m = mw_H2O*W_ton ;// Mass of water condenses per ton dry air -[lb]
//  Since W is positive our assumption(condensation takes place ) is right .
printf('\n(a) Yes water condense out during compression ,since W(%.3f  lb mol) is positive our assumption(condensation takes place ) is right .\n',W);
printf('(b) Amount of water condenses per ton dry air is %.1f lb mol i.e %.0f lb water.\n',W_ton,W_m);
```

Chapter 19

The Phase Rule and Vapor Liquid Equilibria

Scilab code Exa 19.1 `Applications of Phase Rule to Systems without Reaction`

```
clear ;
clc;
// Example 19.1
printf('Example 19.1\n\n');
//Page No. 563
// Solution

// Use phase rule to get degree of freedom(F) =  2-P+C
// (a)
N1 = 1;
P1 = 1 ;// Number of phases present
C1 = 1 ;//Number of components present
F1 = 2-P1+C1 ;//Number of degree of freedom
printf('\n (a) Number of degree of freedom of pure benzene is %i. Therefore %i additional intensive variables must be specified to fix the system.\n',F1,F1);

```

```
// (b)
N2 = 1;
P2 = 2 ;// Number of phases present
C2 = 1 ;//Number of components present
F2 = 2-P2+C2 ;//Number of degree of freedom
printf('(b) Number of degree of freedom of a mixture of ice and water only is %i. Therefore %i additional intensive variables must be specified to fix the system.\n ',F2,F2);

// (c)
N3 = 2;
P3 = 2 ;// Number of phases present
C3 = 2 ;//Number of components present
F3 = 2-P3+C3 ;//Number of degree of freedom
printf('(c) Number of degree of freedom of a mixture of liquid benzene,benzene vapour and helium gas is %i. Therefore %i additional intensive variables must be specified to fix the system.\n ',F3,F3);

// (d)
N4 = 2;
P4 = 2 ;// Number of phases present
C4 = 2 ;//Number of components present
F4 = 2-P4+C4 ;//Number of degree of freedom
printf('(d) Number of degree of freedom of a mixture of salt and water designed to achieve a specific vapour pressure is %i. Therefore %i additional intensive variables must be specified to fix the system.\n ',F4,F4);
```

Scilab code Exa 19.2 `Application of the Phase Rule to Systems in which Reactions c`

```
clear ;
```

```
clc;
// Example 19.2
printf('Example 19.2\n\n');
//Page No.564
// Solution

// Use phase rule to get degree of freedom(F) = 2-P+C
// (a)
N1 = 5;
P1 = 1; // Number of phases present,here 1 gas
C1 = 3 ;//Number of  independent components present, here 3 because 3 elements(C,O and H)
F1 = 2-P1+C1 ;//Number of degree of freedom
printf('\n (a) Number of degree of gas composed of CO,CO2,H2,H2O and CH4 is %i. Therefore %i additional intensive variables must be specified to fix the system.\n ',F1,F1);

// (b)
N2 = 4;
P2 = 4 ;// Number of phases present,here 3 different solid phases and 1 gas phase
C2 = 3 ;//Number of components present, here 3 because 3 elements(Zn,O and C) ,you can also use method explained in Appendix L1
F2 = 2-P2+C2 ;//Number of degree of freedom
printf('(b) Number of degree of freedom of a mixture of ZnO(s), C(s) ,CO(g) and Zn(s)  is %i. Therefore %i additional intensive variables must be specified to fix the system.\n ',F2,F2);
```

Scilab code Exa 19.3 Bubble Point Calculation

```
clear;
```

```
clc;
// Example 19.3
printf('Example 19.3\n\n');
//Page No.576
// Solution

P_atm =  1 ;//[atm]
P =  760 ;//[mm of Hg]
x_1 = 4/100 ;// Mole fraction of hexane in liquid phase
// Constant A,B and C for Antoine eqn. of n_hexane
A1 = 15.8366;
B1 = 2697.55 ;
C1 = -48.784;
// Constant A,B and C for Antoine eqn. of n_octane
A2 = 15.9798;
B2 = 3127.60 ;
C2 = -63.633;

// Solve for bubble point temperature by eqn. obtained by using Antoine equation
deff('[y] = f(T)','y = exp(A1-(B1/(C1+T)))*x_1 +exp(A2-(B2/(C2+T)))*(1-x_1)  - P');
T = fsolve(390,f) ;// Bubble point temperature
funcprot(0);
printf('Bubble point temperature is %.1f K\n',T);

// Composition of first vapour
// Get vapour pressure of hexane and octane from Perry, it is
vp_1 =  3114 ;//  vapour pressure of hexane-[mm of Hg]
vp_2 = 661 ;//  vapour pressure of octane-[mm of Hg]
y_1 = vp_1*x_1/P ;// Mole fraction of hexane in vapour phase
y_2 =  1- y_1 ;//Mole fraction of octane in vapour phase
printf('\n Composition of first vapour.\n ');
```

```
printf('Component                  Mole fraction.\n ');
printf('n_hexane                  %.3f\n',y_1);
printf(' n_octane                  %.3f\n',y_2);
```

Scilab code Exa 19.4 Flash Calculation for a Binary Liquid Mixture

```
clear ;
clc;
// Example 19.4
printf('Example 19.4\n\n');
//Page no. 577
// Solution

// Basis : 100 g solution
F = 100 ;// Amount of solution -[g]
P_atm = 1 ;//[atm]
P = 760 ;// Total pressure -[mm of Hg]
wf_hex = 68.6/100 ;//Weight fraction of hexane in mixture
wf_tol = 31.4/100 ;//Weight fraction of toluene in mixture
mw_hex = 86.17 ;// Mol.wt. of hexane-[g]
mw_tol = 92.13 ;// Mol.wt. of toluene -[g]
mol_hex = wf_hex *F/mw_hex ;// moles of hexane-[g mol]
mol_tol = wf_tol*F/mw_tol ;// moles of toluene-[g mol]
mol_total = mol_hex + mol_tol ;// Total moles in mixture-[g mol]
molf_hex = mol_hex/mol_total ;// Mole fraction of hexane
molf_tol = mol_tol/mol_total ;// Mole fraction of toluene

// Get vapour pressure of hexane and toluene at 80
```

```
deg. C from Perry, it is
vp_hex = 1020 ;// vapour pressure of hexane-[mm of Hg]
vp_tol = 290  ;// vapour pressure of toluene -[mm of Hg]
K_hex = vp_hex/P ;// K-value of hexane
K_tol = vp_tol/P  ;// K-value of toluene
rec_K_hex = 1/K_hex ;// Reciprocal of K-value of hexane
rec_K_tol = 1/K_tol ;// Reciprocal of K-value of toluene

// Let L/F = x, then use eqn. 19.11 to find x(L/F)
deff('[y] = g(x)','y = (molf_hex)/(1-x*(1-rec_K_hex)) + (molf_tol)/(1-x*(1-rec_K_tol))-1');
x = fsolve(1,g) ;// L/F value

printf('\n Fraction of liquid(L/F) that will remain at equilibrium after vaporization is %.3f.\n ',x);
```

Scilab code Exa 19.5 Seperation of Virus from a Culture

```
clear ;
clc;
// Example 19.5
printf('Example 19.5\n\n');
//Page no. 578
// Solution

Vo = 3.0 ;// Initial volume of the solution containing the culture and virus -[L]
Vp = 0.1 ;// Volume of the polymer solution added to the vessel  -[L]
Kpc = 100 ;// Partition coefficient for virus(cp/cc)
```

```
 between two phases

//Use virus particle balance to get cp/co, where co is initial concentration of  virus in solution of culture and virus
Vc = Vo ;// At equilibrium -[L]
cp_by_co = Vo/(Vp+(Vo/Kpc)) ;// Ratio of concentration of virus in polymer phase at equilibrium to initial concentration of virus in culture phase
Fr_rec = cp_by_co*(Vp/Vo) ;// Fraction of the initial virus in the culture phase that is recovered in the polymer phase

printf('\n Fraction of the initial virus in the culture phase that is recovered in the polymer phase  is %.2f .\n ',Fr_rec);
```

Chapter 20

Liquid and Gases in Equilibrium with Solids

Scilab code Exa 20.1 `Fitting Adsorption Isotherms to Experimental Data`

```
clear ;
clc;
// Example 20.1
printf('Example 20.1\n\n');
// Page no. 594
// Solution Fig E20.1

// Given
p_CO2 = [0,25,50,100,200,400,760] ;// Values of partial pressure of CO2 − [mm Hg]
y = [0,6.69*10^-2,9.24*10^-2,0.108,0.114,0.127,0.137] ;// adsorption of CO2 −[g adorbed / g seives]

// R square is a perfect fit
plot(p_CO2,y);
title('Figure E20.1 The Freundlich and Langmuir iotherms coincide for the adsorption of CO2 on 5A molecular seives');
```

```
xlabel('P- partial pressure of CO2');
ylabel('y');
xgrid(1);
```

Scilab code Exa 20.2 Seperation of Biochemicals by Solvent Extraction

```
clear;
clc;
// Example 20.2
printf('Example 20.2\n\n');
//page no. 596
// Solution

//Given
G = 1000 ;// Volume of solution - [L]
S_ad = 1.56 ;// amount of Steptomycin adsorbed per gram resin -[g strep./g resin]
cn_S = 6 ;// Concentration of streptomycin solution -[g/L]
// Assume equilibrium occurs so that total(max) amount of streptomycin is adsorbed
max_S = cn_S*G ;// Maximum streptomycin adsorbed -[g]
//Use streptomycin balance to get amount of resin required
R = max_S/S_ad ;//Amount of resin required to adsorb required amount of streptomycin

printf('Amount of resin required to adsorb required amount of streptomycin is %.0f g .\n ',R);
```

Scilab code Exa 20.3 Combination of an Adsorption Isotherm with a Material Balance

```
clear ;
```

```
clc;
// Example 20.3
printf('Example 20.3\n\n');
//page no. 596
// Solution

//Given
G = 1000 ;// Volume of solution - [L]
x = [19.2,17.2,12.6,8.6,3.4,1.4]  ;// concentration of solute- [g/L]
ac = [0,0.01,0.04,0.08,0.20,0.40]  ;// Activated charcoal added-[g/1000g sol]
// Assume all concentration can be treated as g solute/1000 g sol.

y2 = (x(1)-x(2))/ac(2)  ;// -[ g solute/g carbon]
y3 = (x(1)-x(3))/ac(3)  ;// -[ g solute/g carbon]
y4 = (x(1)-x(4))/ac(4)  ;// -[ g solute/g carbon]
y5 = (x(1)-x(5))/ac(5)  ;// -[ g solute/g carbon]
y6 = (x(1)-x(6))/ac(6)  ;// -[ g solute/g carbon]

// Use polymath to get Freundlich isotherm to bo y=37.919*x^(0.583)
y = 37.919*x(6)^(0.583) ;//From Freundlich isotherm
A_by_G = (x(1)-x(6))/y ;//Minimum mss of activated carbon required- [g carbon/1000 g sol.]

printf('Minimum mass of activated carbon required is %.2f g carbon/1000 g sol. \n ',A_by_G);
```

Chapter 21

Energy Terminology Concepts and Units

Scilab code Exa 21.1 `Calculation of Mechanical Work by a Gas on a Piston Showing H`

```
clear ;
clc;
// Example 21.1
printf('Example 21.1\n\n');
//page no. 616
// Solution Fig. E21.1a and E21.1b

//Given
V1 = 0.1 ;// Volume of gas initially -[cubic metres]
V2 = 0.2 ;// Volume of gas finally -[cubic metres]
T1 = 300 ;// Temperature of gas initially -[K]
P1 = 200 ;// Pressure of gas finally -[kPa]
R = 8.314 ;// Universal gas constant
n = (P1*V1)/(T1*R) ;// Moles of gas taken -[kg mol]
//You are asked to calculate work by eqn. 21.1 , but you do not know the F(force) exerted by gas , so write F = P.A, multiply divide A and eqn 21.1 reduces to W= integate(P.dv)

```

```
//(a)
// Isobaric process see fig E21.1b to see the path followed
W= integrate('-(P1)','V',V1,V2) ;// Work done by gas on piston -[kJ]
printf('\n (a)Work done by gas on piston for isobaric process is %.0f kJ .\n ',W);

//(b)
// Isobaric process see fig E21.1b to see the path followed
W= integrate('-(T1*R*n/V)','V',V1,V2) ;// Work done by gas on piston -[kJ]
printf('(b)Work done by gas on piston for isothermal process is %.2f kJ .\n ',W);
```

Scilab code Exa 21.2 Calculation of the Specific Kinetic Energy for a Flowing Flui

```
clear ;
clc;
// Example 21.2
printf('Example 21.2\n\n');
//page no. 624
// Solution

//Given
id = 3 ;// Internal diameter of tube-[cm]
Vf = 0.001 ;// Volume flow rate of water in tube-[cubic meter/s]
rho = 1000 ;// Assumed density of water-[kg/cubic meter]

rad = id/2 ;// Radius of tube -[ cm]
a = 3.14*rad^2 ;// Area of flow of tube -[squqre centimeter]
```

```
v = Vf*(100)^2/a ;// Velocity of water in tube - [m/s]
KE = v^2/2 ;// Specific(mass=1kg) kinetic energy of water in tube -[J/kg]

printf('Specific kinetic energy of water in tube is %.2f J/kg .\n ',KE);
```

Scilab code Exa 21.3 Calculation of Potential Energy Change in Water

```
clear ;
clc;
// Example 21.3
printf('Example 21.3\n\n');
//page no. 626
// Solution

//Given
// Let water level in first reservoir be the reference plane
h = 40 ;// Difference of water-[ft]
g = 32.2 ;// acceleration due to gravity-[ft/square second]
PE=g*h/(32.2*778.2) ;//// Specific(mass=1kg) potential energy of water -[Btu/lbm]

printf('Specific potential energy of water is %.4f Btu/lbm .\n ',PE);
```

Scilab code Exa 21.4 Calculation of Internal Energy Change using the Heat Capacity

```
clear ;
clc;
```

```
// Example 21.4
printf('Example 21.4\n\n');
//page no. 629
// Solution

//Given
//Constant volume process
mol_air = 10 ;// Moles of air -[kg mol]
T1 = 60+273 ;// Initial temperature of air -[K]
T2 = 30+273 ;// final temperature of air -[K]
// Additional data needed
Cv = 2.1*10^4 ; // Specific heat capacity of air at constant volume-[J/(kg mol*C)]

// Use eqn. 21.6 for del_U
del_U = integrate('mol_air*Cv','T',T1,T2) ;//Change in internal energy -[J]
printf('\nChange in internal energy is %.1e J .\n ',del_U);
```

Scilab code Exa 21.5 Calculation of Internal Energy Change using Different Paths

```
clear ;
clc;
// Example 21.5
printf('Example 21.5\n\n');
//page no. 629
// Solution

printf('\n As we know that internal energy(U) is state variable , therefore change in internal energy(del_U) depends only on initial and final state , independent of the path taken for process .\n Hence, change in internal energy for both paths A and B are same.  ');
```

Scilab code Exa 21.6 `Calculation of the Change in Enthalpy by two Different Paths`

```
clear ;
clc;
// Example 21.6
printf('Example 21.6\n\n');
//page no. 632
// Solution

printf('\n As we know that enthalpy(H) is state
   variable , therefore change in enthalpy(del_H)
   depends only on initial and final state ,
   independent of the path taken for process.\n
   Hence, change in enthalpy for both paths A-B-D
   and A-C-D are same.  ');
```

Scilab code Exa 21.7 `Calculation of an Enthalpy Change`

```
clear;
clc;
// Example 21.7
printf('Example 21.7\n\n');
//page no. 633
// Solution

//Given
//Constant pressure process
mol_air = 10 ;// Moles of air -[kg mol]
T1 = 60+273 ;// Initial temperature of air -[K]
T2 = 30+273 ;// final temperature of air -[K]
// Additional data needed
```

```
Cp = 2.9*10^4  ;// Specific heat capacity of air at constant pressure -[J/(kg mol*C)]

// Use eqn. 21.11 for del_H
del_H = integrate('mol_air*Cp','T',T1,T2) ;//Change in enthalpy -[J]
printf('\nChange in enthalpy is %.1e J .\n ',del_H);
```

Chapter 22

Introduction to Energy Balances for Process without Reaction

Scilab code Exa 22.1 `Application of the Energy Balance to a Closed System`

```
clear;
clc;
// Example 22.1
printf('Example 22.1\n\n');
//page no. 651
// Solution

//Assume that properties of water can be used to
   substitute properties of solution
// Given
V = 1.673 ;// Volume of closed vessel -[cubic metre]
m = 1 ;// mass of saturated liquid vaporized -[kg]
Pi = 1 ;// Initial pressure -[atm]
Ti = 10 ;// Initial temperature -[degree C]
Pf = 1 ;// final pressure -[atm]
Tf = 100 ;// final temperature -[degree C]

```

```
// Use steam table to obtain additional information at given condition
Ui = 35 ;// Initial enthalpy -[kJ/kg]
Uf = 2506.0 ;// Final enthalpy -[kJ/kg]

// Use eqn. 22.2 after modifiying it using given conditions(W = 0,del_KE = 0 and del_PE = 0 )
Q = m*(Uf - Ui) ;// Heat transferred to  the vessel - [kJ]

printf('\nHeat transferred to  the vessel is %.1f kJ.\n ',Q);
```

Scilab code Exa 22.2 Calculation of delta U using American Engineering Units

```
clear ;
clc;
// Example 22.2
printf('Example 22.2\n\n');
//page no. 652
// Solution

// Given
T1 = 80 ;// Initial temperature -[degree F]
T1 = 40 ;// final temperature -[degree F]

// Additional data obtained from steam table at given temperatures and corresponding vapour pressures
p1 = 0.5067 ;// Initial saturation pressure -[psia]
p2 = 0.1217 ;// Final saturation pressure -[psia]
V1 = 0.01607 ;// Initial specific volume - [cubic feet/lb]
V2 = 0.01602 ;// Final specific volume - [cubic feet/lb]
```

```
H1 = 48.02 ;//Initial specific enthalpy -[Btu/lb]
H2 = 8.05 ;// Final specific  enthalpy -[Btu/lb]

del_P = p2 - p1 ;// Change in pressure -[psia]
del_V = V2 - V1 ;// Change in specific volume -[cubic feet/lb]
del_H = H2 - H1 ;// Change in specific enthalpy -[Btu/lb]
del_pV = p2*144*V2/778 - p1*144*V1/778 ;// Change in pv-[Btu]
del_U = del_H - del_pV ;// Change in specific internal energy - [Btu/lb]
del_E = del_U ;// Change in specific total energy(since KE=0,PE=0 and W=0) -[Btu/lb]

printf('\nChange in pressure is %.3f psia .\n ',del_P);
printf('\nChange in specific volume is %.5f cubic feet/lb (negligible value) .\n ',del_V);
printf('\nChange in specific enthalpy is %.2f Btu/lb .\n ',del_H);
printf('\nChange in specific internal energy is %.2f Btu/lb .\n ',del_U);
printf('\nChange in specific total energy is %.2f Btu/lb .\n ',del_E);
```

Scilab code Exa 22.3 Energy Balance to analyze an Open Unsteady State System

```
clear ;
clc;
// Example 22.3
printf('Example 22.3\n\n');
//page no. 662
// Solution fig.E22.3a

```

```
//Lets take tank to be system
// Given
T = 600 ; // Temperature of steam  -[K]
P = 1000 ;// Pressure of steam -[kPa]

// Additional data for steam obtained from CD database at T and P
U = 2837.73 ;// Specific internal energy-[kJ/kg]
H = 3109.44 ;// Specific enthalpy -[kJ/kg]
V = 0.271 ;// Specific volume -[cubic metre/kg]

// Use eqn. 22.6 to get change in specific internal energy,by simplifing it with following assumption:
//1. Change in KE and PE of system = 0, therefore change in total energy = change in internal energy
//2. W = 0,work done by or on the system
//3. Q = 0 , system is well insulated
//4. Change in KE and PE of entering steam = 0
//5. H_out = 0, no stream exits the system
//6. Ut1 = 0, initially no mass exists in the system

// By the reduced equation
Ut2 = H ;// Internal energy at final temperature-[kJ/kg]

printf('\nThe specific internal energy at final temperature is %.2f kJ/kg. \nNow use two properties of the steam (P = %i kPa and Ut2 = %.2f kJ/kg) to find final temperature (T) from steam table. \nFrom steam table we get T = 764 K.',Ut2,P,Ut2);
```

Scilab code Exa 22.4 Application of Energy Balance to an Open Steady State System

```
clear;
clc;
// Example 22.4
printf('Example 22.4\n\n');
//page no. 669
// Solution

// Take milk plus water in tank to be system
// Given
T1_water = 70  ;// Temperature of entering water  -[degree C]
T2_water = 35 ;// Temperature of exiting water  -[degree C]
T1_milk = 15 ;//Temperature of entering milk  -[degree C]
T2_milk = 25 ;//Temperature of exiting milk  -[degree C]

// Get additional data from steam table for water and milk,assuming milk to have same properties as that of water.
H_15 = 62.01 ;//Change in specific internal energy-[kJ/kg]
H_25 = 103.86  ;//Change in specific internal energy-[kJ/kg]
H_35 = 146.69  ;//Change in specific internal energy-[kJ/kg]
H_70 = 293.10  ;//Change in specific internal energy-[kJ/kg]

// Assumptions to simplify Equation 22.8 are:
printf('\nAssumptions to simplify Equation 22.8 are:\n');
printf('1. Change in KE and PE of system = 0.\n');
printf('2. Q = 0 ,because of way we picked the system,it is is well insulated.\n');
printf('3. W = 0,work done by or on the system.\n');

```

```
//Basis m_milk = 1 kg/min , to directly get the answer .
m_milk = 1 ;// Mass flow rate of milk-[kg/min]
// By applying above assumtions eqn. 22.8 reduces to del_H = 0 .Using it get m_water-
m_water = (m_milk*(H_15 - H_25))/(H_35 - H_70) ; // Mass flow rate of water-[kg/min]
m_ratio = m_water/m_milk ;// Mass flow rate of water per kg/min of milk-[kg/min]
printf('\nMass flow rate of water per kg/min of milk is %.2f (kg water/min )/(kg milk/min).',m_ratio);
```

Scilab code Exa 22.5 Calculation of Power needed to Pump Water

```
clear ;
clc;
// Example 22.5
printf('Example 22.5\n\n');
//page no. 670
// Solution

// Take pipe between initial and final level of water
// Given
h_in = -20 ;// Depth of water below ground-[ft]
h_out = 5 ;// Height of water level above ground-[ft]
h = h_out - h_in ;// Total height to which water is pumped-[ft]
V = 0.50 ;// Volume flow rate of water - [cubic feet/s]
ef = 100; // Efficiency of pump - [%]
g = 32.2; // Acceleration due to gravity -[ft/square second]
```

```
gc = 32.2 ;//[(ft*lbm)/(second square*lbf)]

M = V * 62.4 ;// mass flow rate - [lbm/s]
PE_in = 0 ;// Treating initial water level to be reference level
PE_out = (M*g*h*1.055)/(gc*778.2) ;// PE of discharged water -[lbm*(square feet/square second)]

// Assumptions to simplify Equation 22.8 are:
//1. Change in KE = 0.
//2. Q = 0 -By given assumption
//3. Let us assume that temperature of water is same in well and when it is discharged , therefore del_H = 0
// Reduced equation is W = del_PE , using this:
W = PE_out - PE_in ;//Work done on system = power delivered by pump, (since we are using mass flow rate and pump efficiency is 100 % , so W = Power) -[kW]

printf('The electric power required by the pump is %.2f kW. \n', W);
```

Chapter 23

Calculation of Enthalpy Changes

Scilab code Exa 23.1 `Graph Showing the Heat of Vaporization of Water`

```
clear ;
clc;
// Example 23.1
printf('Example 23.1\n\n');
// Page no. 686
// Solution Fig E23.1

// Given
x_Tl = [90,92,97,100] ;// Temperature of saturated
   liquid- [degree C]
x_Tg = [100,102,107,110] ;// Temperature of
   saturated vapour- [degree C]
y_Hl = [376.9,385.3,406.3,418.6] ;// Enthalpy change
    of saturated liquid -[kJ/kg]
y_Hg = [2256.44,2251.2,2237.9,2229.86] ;// Enthalpy
   change of saturated vapour -[kJ/kg]
plot(x_Tl,y_Hl,x_Tg,y_Hg);
title('Figure E23.1 Change in enthalpy Vs
   Temperature ');
```

```
xlabel('T,degree C');
ylabel('H, kJ/kg');
xgrid(1);
```

Scilab code Exa 23.2 Comparision of Various Sources to Estimate the Heat of Vapour

```
clear ;
clc;
// Example 23.2
printf('Example 23.2\n\n');
//page no. 687
// Solution

// Basis : 1 g mol
R = 8.314 * 10^-3 ;// Ideal gas constant −[kJ/(g mol ∗ K)]
Hv = 30.20 ;// Experimental value of heat of vaporization of acetone −[kJ/g]

// additional needed data for acetone from Appendix D
T = 329.2 ;// Normal boiling point of acetone − [K]
Tc = 508.0 ;// Critical temperature  of acetone − [K]
Pc = 47.0 ;// Critical presure of acetone −[atm]

Tbc = T/Tc ;// variable required in etimation equations
lnPc = log(Pc)  ;//  variable required in etimation equations

//(a)
//Using the Clayperon and Antoine's equation(from eqn. 23.2), we get
// del_Hv=(RBT^2)/(C+T)^2 -- eqn. c:
```

```
//From Appendix G
B = 2940.46 ;
C = -35.93 ;
// using eqn. c
del_Hv1 = (R*B*T^2)/((C+T)^2) ;//Heat of vapourization -[kJ/g]
d1 = (abs(Hv - del_Hv1)*100)/Hv ;// differece of experimental and calculated value -[%]
printf('(a) Heat of vapourization of acetone is %.2f kJ/g mol. And differece of experimental and calculated value is %.1f %% . \n', del_Hv1,d1);

//(b)
// Using Chen's equation (from eqn. 23.5)
del_Hv2 = R*T*((3.978*Tbc - 3.938 +1.555*lnPc)/(1.07 - Tbc)) ;//Heat of vapourization -[kJ/g]
d2 = (abs(Hv - del_Hv2)*100)/Hv ;// differece of experimental and calculated value -[%]
printf(' (b) Heat of vapourization of acetone is %.2f kJ/g mol. And differece of experimental and calculated value is %.1f %% . \n', del_Hv2,d2);

//(c)
// Using Riedel's Equation , from equation 23.6
del_Hv3 = 1.093*R*Tc*((Tbc*(lnPc-1))/(0.93-Tbc)) ;// Heat of vapourization -[kJ/g]
d3 = (abs(Hv - del_Hv3)*100)/Hv ;// differece of experimental and calculated value -[%]
printf(' (c) Heat of vapourization of acetone is %.2f kJ/g mol. And differece of experimental and calculated value is %.1f %% . \n', del_Hv3,d3);
```

Scilab code Exa 23.3 Conversion of Units in a Heat Capacity Equation

```
clear ;
```

```
clc;
// Example 23.3
printf('Example 23.3\n\n');
// Page no. 693
// Solution

// Given
// Heat capacity = 2.675*10^4 + (42.27)Tk - 1.425*10^-2Tk^2 J/(kg mol K)
// First convert heat capacity to Btu/(lb mol*F)  to get c + dT - eT^2, where
c = (2.675*10^4*.4536)/(1055*1.8) ;
d = (42.27*.4536)/(1055*1.8) ;
e = (1.425*10^-2*.4536)/(1055*1.8) ;

//Now convert Tk (Temperature in K) to TF (temperature in F) to get answer of form x + yT - zT^2,where
x = c + d*460/1.8 - e*((460/1.8)^2) ;
y = d/1.8;
z = e/(1.8*1.8) ;

printf('The required answer is %.2e + (%.2e)T - (%.3e) T^2 Btu/(lb mol*F) , where T is in degree F .\n',x,y,z);

// Note answer in textbook seems wrong by order of 10^-3
```

Scilab code Exa 23.4 `Fitting Heat Capacity Equation to Heat Capacity Data`

```
clear ;
clc;
// Example 23.4
printf('Example 23.4\n\n');
```

```
//page no. 694
// Solution



//Given
// Cp = a + bT +cT^2
// we will use the least square procedure defined in
     Appendix M
// step 1 : find expression for sum of square of
   residuals: Sr = sum(Cpi - a - bTi - cTi^2)^2

// step 2 : Now differentiate Sr wrt to each
   coefficient to get 3 equation in 3 unknown
   coefficient , the equations are:

     //n*a + sum(Ti)*b +sum(Ti^2)*c = sum(Cpi)
                       ...Eqn.(a)
     //sum(Ti)*a + sum(Ti^2)*b +sum(Ti^3)*c = sum(
        Cpi*Ti)      ...Eqn.(b)
     //sum(Ti^2)*a + sum(Ti^3)*b +sum(Ti^4)*c = sum(
        Cpi*Ti^2)  ...Eqn.(c)

// Take all 18 experimenta data in an array Cp
Cpi =
   [39.87,39.85,39.90;45.16,45.23,45.17;50.72,51.03,50.90;56.85,56.80
    ;// Array of Cpi(Heat capacity) values
// Take corresponding temperatures in array T
Ti =
   [300,300,300;400,400,400;500,500,500;600,600,600;700,700,700;800,8
    ;// array of Ti
Ti_sqr =
   [300^2,300^2,300^2;400^2,400^2,400^2;500^2,500^2,500^2;600^2,600^
    ;// array of Ti^2
Ti_cub =
   [300^3,300^3,300^3;400^3,400^3,400^3;500^3,500^3,500^3;600^3,600^
   // array of Ti^3
Ti_qd =
   [300^4,300^4,300^4;400^4,400^4,400^4;500^4,500^4,500^4;600^4,600^
```

```
    // array of Ti^4
Cpi_Ti =
    [39.87*300,39.85*300,39.90*300;45.16*400,45.23*400,45.17*400;50.7
    ;// Array of Cpi(Heat capacity)*Ti  values
Cpi_Ti_sqr =
    [39.87*300^2,39.85*300^2,39.90*300^2;45.16*400^2,45.23*400^2,45.1
    ;// Array of Cpi(Heat capacity)*Ti^2  values

n = 18 ;// Number of data

// Solve equations (a),(b) & (c) simultaneously
    using matrix
a = [n sum(Ti) sum(Ti_sqr);sum(Ti) sum(Ti_sqr) sum(
    Ti_cub);sum(Ti_sqr) sum(Ti_cub) sum(Ti_qd)] ;//
    Matrix of coefficients of unknown
b = [sum(Cpi);sum(Cpi_Ti);sum(Cpi_Ti_sqr)] ;//
    Matrix of constants
x = (a)^-1 * b ;// Matrix of solutions a = x(1), b =
     x(2) , c = x(3)

printf('The solution is Cp = %.2f + %.3e T + %.2e T
    ^2 .\nTherefore coefficients are as follows :',x
    (1),x(2),x(3));
printf('\n a = %.2f.\n b = %.3e .\n c = %.2e .',x(1)
    ,x(2),x(3));
```

Scilab code Exa 23.5 Calculation of Change in Specific Enthalpy for a Gas Mixture

```
clear ;
clc;
// Example 23.5
printf('Example 23.5\n\n');
//page no. 695
// Solution

```

```
// Basis : 1 g mol of gas
//Given
T1 = 550 ;// Initial temperature - [degree F]
T2 = 200 ;// Final temperature - [degree F]
CO2 = 9.2/100 ;// Mole fraction
CO = 1.5/100 ;// Mole fraction
O2 = 7.3/100 ;// Mole fraction
N2 = 82.0/100 ;//Mole fraction

// Additional data needed :
// Coefficients in the heat capacity equations
a_N2 = 6.895;// constant
b_N2 = 0.7624*10^-3;// coefficient of T
c_N2 = -0.7009*10^-7;// coefficient of square T
a_O2 = 7.104 ;// constant
b_O2 = (0.7851*10^-3);// coefficient of T
c_O2 = (-0.5528*10^-7); // coefficient of square T
a_CO2 = 8.448;// constant
b_CO2 = 5.757*10^-3;// coefficient of T
c_CO2 = -21.59*10^-7;// coefficient of square T
d_CO2 = 3.059*10^-10;// coefficient of cubic T
a_CO = 6.865 ;// constant
b_CO = 0.8024*10^-3;// coefficient of T
c_CO = -0.7367*10^-7; // coefficient of square T

// New coefficients after multiplying mole fraction of each component
a1_N2 = 6.895*N2 ;// constant
b1_N2 = N2*0.7624*10^-3; // coefficient of T
c1_N2 = (-0.7009*10^-7)*N2; // coefficient of square T
a1_O2 = 7.104*O2 ;// constant
b1_O2 = (0.7851*10^-3)*O2;// coefficient of T
c1_O2 = (-0.5528*10^-7)*O2; // coefficient of square T
a1_CO2 = 8.448*CO2;// constant
b1_CO2 = (5.757*10^-3)*CO2;// coefficient of T
c1_CO2 = (-21.59*10^-7)*CO2; // coefficient of
```

```
square T
d1_CO2 = (3.059*10^-10)*CO2; // coefficient of cubic T
a1_CO = 6.865*CO;// constant
b1_CO = (0.8024*10^-3)*CO;// coefficient of T
c1_CO = (-0.7367*10^-7)*CO; // coefficient of square T

// Get net coefficients of T , square T and cubic T by adding them
a_net = a1_N2+a1_CO2+a1_CO+a1_O2; //Net constant
b_net = b1_N2+b1_CO2+b1_CO+b1_O2; //Net coefficient of T
c_net = c1_N2+c1_CO2+c1_CO+c1_O2 ;//Net coefficient of square T
d_net = d1_CO2;//Net coefficient of cubic T

//Cp_net = a_net + b_net*T + c_net*T^2 + d_net*T^3
// Integrate Cp_net*dT over given temperature range to get change in enthalpy
del_H = integrate('(a_net )+( b_net*T) + (c_net*(T^2)) + (d_net*(T^3))','T',T1,T2); // Change in enthalpy of gas over given range -[Btu/lb mol gas]

printf('\n Change in enthalpy of gas over given range is %.0f Btu/lb mol gas .\n ',del_H);
```

Scilab code Exa 23.6 Calculation of the Change in Enthalpy for a Gas using Tabulat

```
clear;
clc;
// Example 23.6
printf('Example 23.6\n\n');
//page no. 700
// Solution
```

```

//Given
N2 = 1 ;// Moles of N2 - [kg mol]
P = 100 ;// Pressure of gas - [kPa]
T1 = 18 ;// Initial temperature - [degree C]
T2 = 1100  ;// Final temperature - [degree C]

// In the book it is mentioned to use tables in
   Appendix D6 to calculate enthalpy change, we get
H_T1 = 0.524;// Initial enthalpy -[kJ/kg mol]
H_T2 = 34.715   ;// Final enthalpy - [kJ/kg mol]
del_H =  H_T2 - H_T1 ;// Change in enthalpy - [kJ/kg
   ]

printf('\n Change in enthalpy of N2 over given range
    is %.3f kJ/kg mol N2 .\n ',del_H);
```

Scilab code Exa 23.7 Use of Steam Tables to Calculate Change in Enthalpy

```
clear;
clc;
// Example 23.7
printf('Example 23.7\n\n');
//page no. 701
// Solution Fig.E23.7

//Given
T1 = 640 ;// Initial temperature -[degree F]
T2 = 480 ;// Final temperature -[degree F]
P1 = 92 ;// Initial pressure -[psia]
P2 = 52 ;// Final pressure - [psia]

// We need to use steam table to get H value at
   initial and final condition by interpolation
//From steam table
```

```
//At 90 psia
H1_600 = 1328.7 ;//H at 90 psia and 600 F-[Btu/lb]
H1_700 = 1378.1 ;//H at 90 psia and 700 F-[Btu/lb]
//At 95 psia
H2_600 = 1328.4 ;//H at 95 psia and 600 F-[Btu/lb]
H2_700 = 1377.8 ;//H at 95 psia and 700 F-[Btu/lb]
//H at 92 psia and 600 F
H3_600 = H1_600+ ((H2_600-H1_600)/(95-90))*(92-90);
    //H  at 92 psia and 600 F-[Btu/lb]
//H  at 92 psia and 700 F
H3_700 = H1_700+ ((H2_700-H1_700)/(95-90))*(92-90);
    //H at 92 psia and 700 F-[Btu/lb]
// Now we need to get V at 92 psia and 640 F
H3_640 = H3_600+((H3_700-H3_600)/(700-600))
    *(640-600);//H at 92 psia and 640 F-[Btu/lb]

//At 50 psia
H1_450 = 1258.7 ;//H at 50 psia and 450 F-[Btu/lb]
H1_500 = 1282.6 ;//H at 50 psia and 500 F-[Btu/lb]
//At 55 psia
H2_450 = 1258.2 ;//H at 55 psia and 450 F-[Btu/lb]
H2_500 = 1282.2 ;//H at 55 psia and 500 F-[Btu/lb]
//V.P at 52 psia and 450 F
H3_450 = H1_450+ ((H2_450-H1_450)/(55-50))*(52-50) ;
    //H at 52 psia and 450 F-[Btu/lb]
//V at 52 psia and 500 F
H3_500 = H1_500+ ((H2_500-H1_500)/(55-50))*(52-50);
    //H at 52 psia and 500 F-[Btu/lb]
// Now we need to get H at 52 psia and 480 F
H3_480 = H3_450+((H3_500-H3_450)/(500-450))
    *(480-450);// H at 52 psia and 480 F-[Btu/lb]

del_H =   H3_480 - H3_640;// Change in enthalpy - [
    Btu/lb]
printf('Change in enthalpy is %.1f Btu/lb .\n',del_H
    );
```

Scilab code Exa 23.8 Use of Steam Table when a Phase Change is involved to Calcula

```
clear ;
clc;
// Example 23.8
printf('Example 23.8\n\n');
//page no. 702
// Solution

//Given
W = 4 ;// Mass of water -[kg]
Ti= 27+273 ;// Initial temperature -[K]
Pi = 200 ;// Initial pressure -[kPa]
// Neglect the effect of pressure on vloume of liquid , therefore
Pf = Pi ;// Final pressure -[kPa]

// From steam table
V1 = 0.001004 ;// Specific volume at Ti -[cubic metre/kg]
V2 = 1000 * V1 ;//Specific volume at final temperature(Tf) from given condition in problem -[cubic metre/kg]

// We need to do interpolation , look in the steam table to get V so as to bracket 1.004 cubic metre / kg at 200 kPa
va = 0.9024  ;// Specific volume -[cubic metre/kg]
Ta = 400 ;// [K]
vb = 1.025 ;// Specific volume -[cubic metre/kg]
Tb = 450 ;//[K]
vf = V2 ;// Final specific volume -[cubic metre/kg]

// Linear interpolation
```

```
m=(Tb - Ta)/(vb - va);// slope
Tf=Ta + m*(vf - va) ;// Final temperature - [K]

printf('\n Final temperature is %.0f K.\n',Tf);
```

Scilab code Exa 23.9 Calculate Enthalpy Difference betwee two States by Pressure E

```
clear ;
clc;
// Example 23.9
printf('Example 23.9\n\n');
//page no. 704
// Solution

//Given
mv = 1 ;// Mass of saturated vapour - [lb]
P1 = 2 ;// Initial pressure -[atm]
P2 = 20 ;// Final pressure -[atm]

// Additional data is obtained from figure 23.6 of the book on page no. 703
H_2 = 179 ;// Specific enthalpy at 2 atm - [Btu/lb]
H_20 = 233 ;//  Specific enthalpy at 20 atm - [Btu/lb]
V_2 = 3.00 ;// Specific volume at 2 atm - [cubic feet/lb]
V_20 = 0.30 ;//  Specific volume at 20 atm - [cubic feet/lb]
T_2 = 72 ;// Temperature at 2 atm -[degree F]
T_20 = 239 ;// Temperature at 20 atm -[degree F]
del_H = H_20 - H_2 ;// Change in specific enthalpy -[Btu/lb]
del_V = V_20 - V_2 ;// Change in specific volume -[cubic feet/lb]
del_T = T_20 - T_2 ;// Change in temperature -[
```

```
degree F]
printf('(a) Change in specific enthalpy is %.0f Btu/lb.\n',del_H);
printf(' (b) Change in specific volume is %.2f cubic feet/lb.\n',del_V);
printf(' (c) Change in temperature is %.1f degree F.\n',del_T);
```

Chapter 24

Applications of Energy Balances in the Absence of Chemical Reactions

Scilab code Exa 24.1 `Simplification of General Energy Balance`

```
clear ;
clc;
// Example 24.1
printf('Example 24.1\n\n');
//page no. 720
// Solution Fig. E24.1

// Assumptions to be made in eqn. 24.1 in following
   segment
printf('Assumptions to be made in eqn. 24.1 in
   following segments are:\n');
//(a)- 1 to 5
printf('\n(a)- 1 to 5.\n');
printf('  1. Change in potential energy(del_PE) = 0(
   no change in level) .\n');
printf('  2. Probably change in kinetic energy(
   del_KE)=0 .\n');
```

```
printf('  3. Change in energy = 0 (process appears to be steady).\n');
printf('  Result : Q + W = del_H.\n');

//(b) 4 to 5
printf('\n\n(b) 4 to 5.\n');
printf('  1. Q = W = 0 \n');
printf('  2. Probably change in kinetic energy(del_KE)=0.\n');
printf('  3. Change in energy = 0 (process appears to be steady).\n');
printf('  Result : del_H = -del_PE . \n');

//(c) 3 to 4
printf('\n\n(c) 3 to 4.\n');
printf('  1. Q = W = 0 \n');
printf('  2. Probably change in kinetic energy(del_KE)=0.\n');
printf('  3. Change in energy = 0 (process appears to be steady).\n');
printf('  Result : del_H = -del_PE . \n');

//(d) 3 to 5
printf('\n\n(d) 3 to 5.\n');
printf('  1. Q = W = 0 \n');
printf('  2. Probably change in kinetic energy(del_KE)=0.\n');
printf('  3. Change in energy = 0 (process appears to be steady).\n');
printf('  4. Change in potential energy(del_PE) = 0(no change in level) .\n');
printf('  Result : del_H = 0 . \n');

//(e)- 1 to 3
printf('\n(e) 1 to 3.\n');
printf('  1. Change in potential energy(del_PE) = 0(no change in level) .\n');
printf('  2. Probably change in kinetic energy(
```

```
   del_KE)=0 .\n');
printf('  3. Change in energy = 0 (process appears
   to be steady).\n');
printf('  Result : Q + W = del_H.\n');
```

Scilab code Exa 24.2 `Degree of Freedom Analysis Including an Energy Balance`

```
clear ;
clc;
// Example 24.2
printf('Example 24.2\n\n');
//page no. 725
// Solution

printf('Table to carry out degree of freedom
   analysis:\n');
// Number of variables involved
printf('\nI. Number of variables involved.\n');
printf('\n  For materials:\n');
printf('        Hot gas : 4 component flows, T, and
   p                        6 \n');
printf('        Cool gas : 4 component flows, T, and
    p                       6\n');
printf('        Water in : 1 component flow, T, and
   p                        3\n');
printf('        Water out : 1 component flow, T, and
    p                       3\n');
printf('\n  Energy:\n');
printf('        Q and W

   2 \n');
printf('        H,KE and PE associated with each
   stream flow                 12 \n');
printf('\
    ----------------------------------------------------------------
```

```
    \n');
printf('  Total

    32\n');
printf('\n\nII. Number of equations and
    specifications.\n');
printf('\n  Specified values:\n');
printf('        Hot gas : 4 component flows, T, and
    p                    6 \n');
printf('        Cool gas : T, and p
                                        2\n');
printf('        Water in : T, and p
                                        2\n');
printf('        Water out : T, and p
                                       2\n');
printf('\n  Specified in the energy balance:\n');
printf('        Q and W

    2 \n');
printf('        KE and PE associated with each of 4
    stream flow          8 \n');
printf('\n  Material balance:              \n');
printf('        4 species balances plus water
                              5 \n');
printf('\n  Energy balance:

    1\n');
printf('\n  H in each stream is a function of
    specified T and p             4\n');
printf('\
    ---------------------------------------------------------------------------------
    \n');
printf('  Total

    32\n');
printf('\n Therefore, by analysing the above table
    it is clear that degrees of freedom of system is
    (32 - 32) = 0 \n');
```

Scilab code Exa 24.3 `Application of Energy Balance`

```
clear ;
clc;
// Example 24.3
printf('Example 24.3\n\n');
//page no. 728
// Solution Fig. E24.3

// Given
 m_CO2 = 10 ;// mass of CO2 - [lb]
 Ti_CO2 = 80 ;// Initial temperature of CO2 - [
    degree F]
 Vi = 4.0 ;// Initial volume of CO2-[cubic feet]
 f_CO2 = 40/100 ;// Fraction of CO2 that convert to
    liquid finally
 s_Vi = Vi /m_CO2 ;// Initial specific volume of CO2
     - [cubic feet/lb]
 s_Vf = s_Vi ;// Constant volume -[cubic feet/lb]
// Use the CO2 chart in Appendix J  to necessary
   data, according to book it is
// CO2 is gas at start of process and reference
   state for the CO2 chart is -40 degree F ,
   saturated liquid
// From chart
Pi = 300 ;// Intial pressure - [psia]
del_Hi = 160 ;// Intial change in specific enthalpy
   - [Btu/lb]
// Now again use chart to get fnal condition fixed
   by constant volume line and quality 0.6 ,
   according to book it is
del_Hf = 81 ;// Final change in specific enthalpy -
   [Btu/lb]
Pf = 140 ;//Final pressure - [psia]
```

```
// Use conditions given in problem ( W= 0 ,since volume is constant ,therefore del_PE and del_KE =0 ),simplifing the energy balance equation we get Q = del_H - del_(PV)
// Analysing the given conditions dof of system = 0 , with 1 eqn. and 1 unknown Q
Q = ((del_Hf - del_Hi) - (Pf * s_Vf * 144/778.2 - Pi * s_Vi * 144/778.2))*m_CO2 ;// Heat removed from the extinguisher -[Btu]

 printf(' Heat removed from the extinguisher is %i Btu .\n',Q);
```

Scilab code Exa 24.4 Applications of Energy Balance to Plasma Etching

```
clear ;
clc;
// Example 24.4
printf('Example 24.4\n\n');
//page no. 730
// Solution

// Pick the system as gas plus heater
// Given
Pi = 1.5 ;// Intial pressure - [Pa]
Vi = 2*10^-3 ;// Initial volume of gas - [cubic metre]
Ti = 300 ;// Initial temperature - [K]
W = 480 ;// Work done by heater on system
t = 5 ;// Time for which current is supplied -[ min]
m_ht = 12 ;// Mass of the heater - [g]
C_ht = 0.35 ;// Heat capacity of heater - [ J/gK]
R = 8.314 ;// Ideal gas constant - [(Pa*cubic metre)/(g mol* K)]

```

```
// It is assumed that heat transfer across system boundary for this short time is negligible , therefore Q = 0
// Using the above assumption the equation reduces to del_U = W, therefore
del_U = W ;// Change in nternal energy - [J]

// Gas is assumed to be ideal , therefore get n by using pv = nRT
n = (Pi*Vi)/(R*Ti) ;// Number of moles of argon gas -[g mol]
Cp = (5/2)* R ;// Specific heat capacity of argon gas at constant pressure - [ J/gK]
Cv = Cp - R ;//  Specific heat capacity of argon gas at constant volume - [ J/gK]
// del_Ug = n*Cv*(Tf - Ti) - change in internal energy of gas
// del_Uh = m_ht*C_ht*(Tf - Ti) - change in internal energy of gas
// get total change in internal energy =  del_Ug + del_Uh , and solve it for Tf ( final temperature )
deff('[y]=f(Tf)','y=m_ht*C_ht*(Tf - Ti) + n*Cv*(Tf - Ti) - del_U');
Tf=fsolve(400,f) ;// Final temperature -[K]
funcprot(0);
 printf(' Final temperature of gas is %.0f K .\n',Tf);

 Pf = (Tf/Ti)*Pi ;// Final pressure - [Pa]
  printf(' Final pressure in chamber is %.2f Pa .\n',Pf);
```

Scilab code Exa 24.5 Energy Balance applied to a Batch Process

```
clear ;
clc;
// Example 24.5
printf('Example 24.5\n\n');
//page no. 732
// Solution Fig. E24.5

// Pick the system as shown in above figure of book
// Given
m_water = 10 ;// Mass of water  - [lb]
T_water = 35 ;// Temperature of water - [degree F]
m_ice = 4 ;// Mass of ice - [lb]
T_ice = 32 ;// Temperature of ice - [degree F]
m_stm = 6 ;// Initial mass of steam -[lb]
T_stm = 250 ;// Temperature of stm - [degree F]
p = 20 ;// Pressure of system -[psia]

m_total = m_water + m_ice + m_stm ;// Mass of H2O in three phases initially -[lb]
// By following conditions of book, Q = 0, W = 0 , change in PE and change in KE = 0, the energy balance reduces to del_U = 0

// According to book additional information is obtained from the steam table and CD at given conditions ,it is as follows
U_ice = -143.6 ;// Specific internal energy of ice -[Btu/lb]
U_water = 3.025 ;// Specific internal energy of water -[Btu/lb]
U_stm = 1092.25 ;// Specific internal energy of steam -[Btu/lb]
V_water = 0.0162 ;// Specific volume of water -[cubic feet/lb]
V_stm = 20.80 ;// Specific volume of steam -[cubic feet/lb]
V_total = m_stm*V_stm ;//Total volume of container ignoring volume of water and ice as they are
```

```
neglgible

V_sys = V_total/m_total  ;// Specific volume of system -[cubic feet/lb]
U_sys =(m_water*U_water + m_ice*U_ice + m_stm*U_stm)/m_total ;// Final specific internal energy of system -[Btu/lb]

// Trial and error method
// Assume two temperatures and find volume total so as to bracket value of U_sys, Here e take T1 = 190 and T2 = 200 degree F
// Obtain necessary data from steam table at corresponding temperatures

T1 = 190 ;// assumed temperature
U1 = [157.17 1071.83] ;//specific internal energy of liquid and vapour respetively -[Btu/lb]
V1 = [0.0165 41.01] ;// Specific volume of liquid and vapour respetively  -[cubic feet/lb]
x1 = V_sys/V1(2) ;// Quality of vapour
U1_sys = (1-x1)*U1(1) + x1*U1(2); // Specific internal energy of system at T1-[Btu/lb]

T2 = 200 ;// assumed temperature
U2 = [168.11 1073.96];//  specific internal energy of liquid and vapour respetively -[Btu/lb]
V2 = [0.017 33.601] ;// Specific volume of liquid and vapour respetively  -[cubic feet/lb]
x2 = V_sys/V2(2) ; // Quality of vapour
U2_sys = (1-x2)*U2(1) + x2*U2(2) ;// Specific internal energy of system at T2-[Btu/lb]

// Check whether assumption is right
if (U_sys > U1_sys  )
   if ( U_sys <U2_sys)
        printf('Assumption is right, now find exact temperature by interpolation between 2
```

```
assumed temperatures.\n');
    else
        printf('Assumption is wrong, assume different T2.\n');
    end
  else
 printf('Assumption is wrong,assume different T1.\n');
 end

// Interpolation , to get final temperature corresponding to U_sys
T_sys = T1 + ((T2 - T1)*(U_sys - U1_sys))/(U2_sys - U1_sys);

 printf(' The final temperature obtained by interpolation between 2 assumed temperatures is %.2f degree F.\n',T_sys);

// Now obtain specific volume of vapour data at final temperature from steam table and use it to calculate x(quality) , according to book it is
V_vap = 39.35 ;//specific volume of vapour data at final temperature -[cubic feet/lb]
x = V_sys /V_vap ;// Quality of gas at final temperature

 //Final state
Vap = m_total*x ;// Mass of vapour at final state -[lb]
stm_con =  m_stm - Vap ;// Mass of steam condenses -[lb]

printf('\nTherefore, mass of steam condenses is %.2f lb.\n',stm_con);
```

Scilab code Exa 24.6 Applications of Energy Balance to Pumping Water

```
clear ;
clc;
// Example 24.6
printf('Example 24.6\n\n');
//page no. 736
// Solution Fig. E24.6

// Pick the system as shown in above figure of book
// Given
h1 = -15 ;// Initial level of water from ground level -[ft]
h2 = 165 ;//Final level of water from ground level -[ft]
V_rate = 200 ;// Volume flow rate of water - [gal/hr]
Q1 = 30000 ;// Heat input by heater - [Btu/hr]
Q2 = 25000 ;// Heat lost by system -[Btu/hr]
T1 = 35 ;// Initial temperature of water - [degree F]
g = 32.2 ;// Acceleration due to gravity - [ft/square second]
p_pump = 2 ;// Power of pump -[hp]
f_w = 55/100 ;// Fraction of rated horsepower that i used in pumping water
Cp = 1 ;// Specific heat capacity of water - [Btu/lb*F]

// Use following conditions to simplify the energy balance
// 1. Proces is in steady state , so change in energy = 0
// 2. m1 = m2 = m
```

```
// 3. change in KE = 0 , because we will assume that v1 = v2 = 0
// The energy balance reduce to Q + W = del_(H*m + PE*m)

m = V_rate * 8.33 ;// Total mass of water pumped -[lb]
del_PE = (m* g *(h2 - h1))/(32.2*778) ;// Change in PE - [Btu/hr]
Q = Q1 - Q2 ;// Net heat exchange -[Btu/hr]
W = 2* f_w * 60 * 33000/778 ;// Work on system - [Btu/hr]
del_H = Q + W - del_PE ;// By using reduced energy balance - [Btu/hr]
// Also del_H = m* Cp * (T2 - T1), all is known except T2 , solve for T2
deff('[y] = f(T2)','y = m*Cp*(T2-T1) - del_H');
T2 = fsolve(40,f) ;// Boiling point temperature
funcprot(0);

  printf(' Final temperature of water that enters storage tank is %.1f degree F .\n',T2);
```

Scilab code Exa 24.7 Applications of Energy Balance to Heating a Biomass

```
clear ;
clc;
// Example 24.7
printf('Example 24.7\n\n');
//page no. 738
// Solution Fig. E24.7

// Pick the system as shown in above figure of book
// Given
T_stm = 250 + 273 ;// Temperature of entering steam
```

```
- [K ]
Q_loss = -1.5 ;// Rate of heat loss from system - [ kJ/s ]
T_mi = 20 + 273 ;//Temperature of entering material -[K ]
T_mf = 100 + 273 ;// Temperature of material after heating - [K]
m_m = 150 ;// Mass of charged material - [kg]
Cp_m = 3.26 ;// Average heat capacity of material - [ J/(g*K)]

// Use following conditions to simplify the energy balance
// 1. Proces is not in steady state , so change in energy not equals  0.
// 2. Assume del_KE and del_PE = 0.
// 3. Assume del_KE and del_PE = 0, for entering and  exiting material .
// 4. W = 0.
// 5. Assume m1 = m2 = m_stm
// The energy balance reduce to    del_E = del_U = Q - del_(H*m) .... eqn. (b)

del_U = m_m*Cp_m*(T_mf - T_mi) ;// Change in enthalpy of system , because del_(pV) = 0 for liquid and solid charge -[kJ]
Q_loss_total = Q_loss * 3600; // Total heat loss by system n 1 hour - [kJ]
// We need the value of specific change in enthalpy value of saturated steam(del_H_steam), according to book we can obtain this value from steam table , it's value is -1701 kJ/kg
del_H_steam = -1701 ;// Specific change in enthalpy value of saturated steam -[kJ/kg]
// Get mass of steam per kg charge from reduced energy balance(eqn. (b))
m_stm_total = (del_U - Q_loss_total)/(-del_H_steam) ;// Total mass of stea used - [kg]
```

```
m_stm = m_stm_total/m_m ;// Mass of steam used per kg of charge - [kg]

 printf(' Mass of steam used per kg of charge is %.3f kg .\n',m_stm);
```

Scilab code Exa 24.8 `Sterilization of a Fermentation Medium`

```
clear ;
clc;
// Example 24.8
printf('Example 24.8\n\n');
//page no. 741
// Solution Fig. E24.8

// Pick the system of whole process as shown in above figure of book
// Given
Q = 1.63 ;// Heat loss from the process - [ kW ]
m_bm = 150 ;// Mass flow rate of biological media into the sterlizer -[kg/min]
T_bm = 50 +273 ;// Temperature of biological media into the sterlizer - [K]
T_sm = 75 + 273 ;// Temperature of sterlize media out of the sterlizer - [K]
P_ss = 300 ;// Pressure of satureted steam entering the steam heater - [kPa]
P_sc = 300 ;// Pressure of satureted condensate exiting the steam heater - [kPa]

// Additional data of change in enthalpy is obtained from the steam table , according to book the data are
H_w1 = 207.5 ;// Change in specific enthalpy of water at 50 degree C - [kJ/kg]
```

```
H_w2 = 310.3 ;// Change in specific enthalpy of water at 75 degree C - [kJ/kg]
H_ss = 2724.9 ;//Change in specific enthalpy of satureted steam entering the steam heater at 300 kPa - [kJ/kg]
H_sc = 561.2 ;//Change in specific enthalpy of satureted condensate exiting the steam heater at 300 kPa - [kJ/kg]

// Use following conditions to simplify the energy balance
// 1. Proces is in steady state , so change in energy = 0.
// 2. Assume del_KE and del_PE = 0.
// 3. W = 0.
// 4. Assume m1 = m2 = m_stm
// The energy balance reduce to   Q = H_out - H_in , using it
m_sm = m_bm ;// By material balance -[kg/min]
m_stm = (Q*60 - m_sm*H_w2 + m_bm * H_w1  )/(H_sc - H_ss ) ;// Mass flow rate of steam entering the steam heater - [kg/min]

 printf(' Mass flow rate of steam entering the steam heater is %.2f kg steam/min .\n',m_stm);
```

Scilab code Exa 24.9 Use of Combined Material and Energy Balances to Solve a Disti

```
clear ;
clc;
// Example 24.9
printf('Example 24.9\n\n');
//page no. 742
// Solution Fig. E24.9a and Fig. E24.9b

```

```
// Given

// For material balance
F = 20000 ;// Feed rate of saturated liquid - [kg/h]
F_Bz = 0.5 ;// Fraction of benzene in feed
F_Tol = 0.5 ;// Fraction of toluene in feed
D_Bz = 0.98 ;// Fraction of benzene in distillate
D_Tol = 0.02 ;// Fraction of toluene in distillate
B_Bz = 0.04 ;// Fraction of benzene in bottoms
B_Tol = 0.96 ;// Fraction of toluene in bottoms
R_by_D = 4.0 ;// Recycle ratio
// Analysing the condition for material balance ,
   degree of freedom  is 0.
// Solve equations obtained by material balances ,
   simultaneously to get B and D
a = [1 1;B_Bz D_Bz] ;// Matrix formed by
   coefficients of unknown
b = [ F ; F_Bz*F ] ;// Matrix formed by contants
x = a\b ;// Matrix of solutions
B = x(1) ;// Bottoms - [kg/h]
D = x(2) ;//Distillate - [kg/h]
R = D * R_by_D ;// Recycle - [kg/h]
V = R + D ;// Overhead vapour - [kg/h]

// For energy balance
// According to book additional data obtained from
   the fig.E24.9b are
H_F = 165 ;// Change in enthalpy of F - [kJ/kg]
H_B = 205 ;// Change in enthalpy of B - [kJ/kg]
H_D = 100 ;// Change in enthalpy of D - [kJ/kg]
H_R = 100 ;// Change in enthalpy of R - [kJ/kg]
H_V = 540 ;// Change in enthalpy of V - [kJ/kg]

Qc = R*H_R + D*H_D - V*H_V ;// The heat duty in the
   condenser - [kJ]
Qr = D*H_D + B*H_B - F*H_F - Qc;// The heat duty to
   the reboiler - [kJ]

```

```
printf('  Ditillate (D)                                  %.2e kg/h.\n',D);
printf('  Bottoms (B)                                     %.2e kg/h.\n',B);
printf('  The heat duty in the condenser (Qc)      %.2e kJ/h.\n',Qc);
printf('  The heat duty to the reboiler (Qr)         %.2e kJ/h.\n',Qr);
```

Chapter 25

Energy Balances How to Account for Chemical Reaction

Scilab code Exa 25.1 `Determination of a Heat of Formation from Heat Transfer Measu`

```
clear ;
clc;
// Example 25.1
printf('Example 25.1\n\n');
//page no. 766
// Solution Fig. E25.1

// Given
// C(s) + O2(g) --> CO2(g)                    (A)
// CO(g) + (1/2)(O2)(g) --> CO2 (g)       (B)
Qa = -393.51 ;// Heat of reaction of reaction (a) - [kJ/g mol C]
Qb = -282.99 ;// Heat of reaction of reaction (b) - [kJ/g mol CO]
del_Ha = Qa ;// Change in enthalpy of reaction A - [kJ/g mol C]
del_Hb = Qb ;// Change in enthalpy of reaction B - [
```

```
kJ/g mol CO]

// According to Hess's Law , subtract reaction (B) from reaction (A) , subtract corresponding del_H's to get enthalpy of formation of reaction (C)- C(s) + (1/2)*O2 --> CO(g) , therefore
del_Hfc = del_Ha - del_Hb ;// Standard heat of formation of CO - [kJ/g mol C]
printf('Standard heat of formation of CO is %.2f kJ/g mol C.',del_Hfc) ;
```

Scilab code Exa 25.2 Retrieval of Heats of Formation from Reference Data

```
clear ;
clc;
// Example 25.2
printf('Example 25.2\n\n');
//page no. 767
// Solution

// Given
// The main reaction is (1/2)*H2(g) +(1/2)* Cl2(g) --> HCl(g)                    (A)
// Look in Appendix F for heat of formation of H2 , Cl2 and HCl
H_H2 = 0 ;// Standard heat of formation of H2 -[kJ/g mol H2]
H_Cl2 = 0 ;// Standard heat of formation of Cl2 -[kJ/ g mol Cl2]
H_HCl = -92.311 ;//  Standard heat of formation of HCl -[kJ/ g mol HCl]

H_f = 1*H_HCl - (1/2)*(H_H2 + H_Cl2) ; // Standard heat of formation of HCl by reaction - [kJ/ g mol HCl]
```

```
printf('Standard heat of formation of HCl(g) is %.3f kJ/g mol HCl.',H_f) ;
```

Scilab code Exa 25.3 Calculation of the Standard Heat of Reaction from the Standar

```
clear ;
clc;
// Example 25.3
printf('Example 25.3\n\n');
//page no. 771
// Solution

// Given
// The main reaction is 4*NH3(g) + 5*O2(g) --> 4*NO(g) + 6*H2O                    (A)
H_fNH3 = -46.191 ;// Standard heat of formation of NH3 -[kJ/ g mol]
H_fO2 = 0 ;//Standard heat of formation of O2 -[kJ/ g mol]
H_fNO = 90.374 ;// Standard heat of formation of NO -[kJ/ g mol]
H_fH2O = -241.826 ;// Standard heat of formation of H2O -[kJ/ g mol]

// Heat of above reaction is calculated by eqn. 25.1
H_rxn = ((4*H_fNO + 6*H_fH2O) - (4*H_fNH3 + 5*H_fO2))/4 ;// Heat of above reaction -[kJ/ g mol NH3]

printf('Heat of above reaction is %.3f kJ/g mol NH3.',H_rxn) ;
```

Scilab code Exa 25.4 Calculation of the Heat of Reaction at a Temperature differen

```
clear ;
clc;
// Example 25.4
printf('Example 25.4\n\n');
//page no. 773
// Solution

// Given
P1 = 1 ;// Initial pressure - [atm]
P2 = 1 ;// Final pressure - [atm]
T1 = 500 ;// Initial temperature -[degree C]
T2 = 500 ;// Final temperature -[degree C]

// The main reaction is CO2(g) + 4H2(g) + ---> 2H2O(g) + CH4(g)                                  (A)
// Data obtained from above reaction
m_CO2 = 1 ;// Moles of CO2 - [ g mol]
m_H2 = 4 ;// Moles of H2 - [ g mol]
m_H2O = 2 ;// Moles of H2O - [ g mol]
m_CH4 = 1 ;// Moles of CH4 - [ g mol]

// Additional required data are obtained from CD, according to book it is a follows-
H_fCO2 = -393.250 ;// Heat of formation of CO2 - [kJ/g mol]
H_fH2 = 0 ;// Heat of formation of H2 - [kJ/g mol]
H_fH2O = -241.835 ;// Heat of formation of H2O - [kJ/g mol]
H_fCH4 = -74.848 ;// Heat of formation of CH4 - [kJ/g mol]

H_CO2 = 21.425 ;// Change in enthalpy during temperature change from 25 to 500 degree C  of CO2 - [kJ/g mol]
H_H2 = 13.834 ;// Change in enthalpy during temperature change from 25 to 500 degree C of H2 - [kJ/g mol]
H_H2O = 17.010 ;// Change in enthalpy during
```

```
temperature change from 25 to 500 degree C of H2O - [kJ/g mol]
H_CH4 = 23.126 ;// Change in enthalpy during temperature change from 25 to 500 degree C of CH4 - [kJ/g mol]

H_rxn_25 = (m_CH4*H_fCH4 + m_H2O*H_fH2O) - (m_CO2*H_fCO2 + m_H2*H_fH2) ;// Heat of reaction at 25 C
sum_H_rct = m_CO2*H_CO2 + m_H2*H_H2 ;// sum of heat of formation of reactant - [kJ]
sum_H_pdt = m_CH4*H_CH4 + m_H2O*H_H2O ;//sum of heat of formation of product - [kJ]
// Heat of above reaction is calculated by eqn. 25.4
H_rxn_500 = sum_H_pdt - sum_H_rct + H_rxn_25  ;// Heat of reaction at 500 C

printf('Heat of above reaction at 500 degree C and 1 atm is %.1f kJ.',H_rxn_500) ;
```

Scilab code Exa 25.5 Calculation of the Heat Transfer using Heat of Reaction in a

```
clear ;
clc;
// Example 25.5
printf('Example 25.5\n\n');
//page no. 775
// Solution

// Given
// The main reaction is CO2(g) + 4H2(g) + --> 2H2O(g) + CH4(g)                    (A)
// Data obtained from above reaction
m_CO2 = 1 ;// Moles of CO2 - [ g mol]
m_H2 = 4 ;// Moles of H2 - [ g mol]
m_H2O = 2 ;// Moles of H2O - [ g mol]
```

```
m_CH4 = 1 ;// Moles of CH4 - [ g mol]
P1 = 1 ;// Initial pressure  - [atm]
P2 = 1 ;// Final pressure - [atm]

T1_CO2 = 800 ;// Initial temperature of entering CO2 -[K]
T1_H2 = 298 ;// Initial temperature of entering H2 -[K]
T2 = 1000 ;// Temperature of exiting product - [K]

// The main reaction is CO2(g) + 4H2(g) + --> 2H2O(g) + CH4(g)                              (A)
// Data obtained from above reaction
m1_CO2 = 1 ;// Moles of entering CO2 - [ g mol]
m1_H2 = 4 ;// Moles of entering H2 - [ g mol]
f_con = 70/100 ;// Fractional conversion of CO2
m2_H2O = 2*f_con ;// Moles of H2O in product - [ g mol]
m2_CH4 = 1*f_con ;// Moles of CH4 in product - [ g mol]
m2_CO2 = m1_CO2*(1-f_con) ;//  Moles of CO2 in product - [ g mol]
m2_H2 = m1_H2*(1-f_con) ;//  Moles of CO2 in product - [ g mol]

// Additional required data are obtained from CD, according to book it is a follows-
H_fCO2 = -393.250 ;// Heat of formation of CO2 - [kJ/g mol]
H_fH2 = 0 ;// Heat of formation of H2 - [kJ/g mol]
H_fH2O = -241.835 ;// Heat of formation of H2O - [kJ/g mol]
H_fCH4 = -74.848 ;// Heat of formation of CH4 - [kJ/g mol]


H1_CO2 = 22.798 ;// Change in enthalpy during temperature change from 298K to 800 K  of CO2 - [
```

```
kJ/g mol]
H1_H2 = 0 ;// Change in enthalpy during temperature change from 298K to 298 K  of H2 - [kJ/g mol]
H2_H2O = 25.986 ;// Change in enthalpy during temperature change from 298K to 1000 K  of H2O - [kJ/g mol]
H2_CH4 = 38.325 ;// Change in enthalpy during temperature change from 298K to 1000 K  of CH4 - [kJ/g mol]
H2_CO2 = 33.396; // Change in enthalpy during temperature change from 298K to 1000 K  of CO2 - [kJ/g mol]
H2_H2 = 20.620; // Change in enthalpy during temperature change from 298K to 1000 K  of H2 - [kJ/g mol]

H_rxn_25 = (m_CH4*H_fCH4 + m_H2O*H_fH2O) - (m_CO2*H_fCO2 + m_H2*H_fH2) ;// Standard heat of reaction at 25 C-[kJ]
H_rxn_ac = f_con*H_rxn_25 ;// Heat of reaction actual - [kJ]
sum_H_rct = m1_CO2*H1_CO2 + m1_H2*H1_H2 ;// sum of heat of formation of reactant - [kJ]
sum_H_pdt = m2_CH4*H2_CH4 + m2_H2O*H2_H2O + m2_CO2*H2_CO2 + m2_H2*H2_H2 ;//sum of heat of formation of product - [kJ]
// Heat of above reaction is calculated by eqn. 25.4
H_rxn = sum_H_pdt - sum_H_rct + H_rxn_ac  ;// Heat of reaction -[kJ/ g mol CO2]

// By above conditions the energy balance reduces to  Q = del_H , therefore
Q = H_rxn ;// Heat transfer to/from the reactor - [kJ]
printf('Heat transfer to/from the reactor  is %.3f kJ.\nSince Q is negative , the reactor losses heat.',Q) ;
```

Scilab code Exa 25.6 Calculation of the Enthalpy Change in an Anerobic Culture

```
clear ;
clc;
// Example 25.6
printf('Example 25.6\n\n');
//page no. 776
// Solution

// Given
H_EtOH =-1330.51  ;// Change in enthalpy of ethanol -[kJ/g mol]
H_Ac = -887.01 ;// Change in enthalpy of acetate -[kJ/g mol]
H_Fr = -221.75 ;// Change in enthalpy of formate -[kJ/g mol]
H_Lc = -1330.51 ;// Change in enthalpy of lactate -[kJ/g mol]
H_Mn = -2882.78  ;// Change in enthalpy of mannitol -[kJ/g mol]
mol_EtOH =1.29   ;//ethanol produced / g mol mannitol -[g mol]
mol_Ac = 0.22    ; //acetate produced / g mol mannitol -[g mol]
mol_Fr = 1.6    ; //formate produced / g mol mannitol-[g mol]
mol_Lc = 0.4    ;//lactate produced / g mol mannitol -[g mol]
mol_Mn = 1.0    ;//mannitol produced / g mol mannitol-[g mol]
B_growth = 40.5   ;// Biomass growth -[g cells/g mol mannitol]

// (a)
```

```
del_H1 = H_EtOH*mol_EtOH +H_Ac*mol_Ac + H_Fr*mol_Fr + H_Lc*mol_Lc - H_Mn*mol_Mn ;// Net enthalpy change for several products (metabolites) per g mol mannitol consumed -[kJ]
printf(' (a) Net enthalpy change for several products (metabolites) per g mol mannitol consumed is %.2f kJ.\n',del_H1) ;

//(b)
del_H2 = del_H1 / B_growth ;//Net enthalpy change for several products (metabolites) per g cells produced -[kJ]
printf('  (b) Net enthalpy change for several products (metabolites) per g cells produced is %.2f kJ.',del_H2) ;
```

Scilab code Exa 25.7 `Green Chemistry Examining Alternate Processes`

```
clear ;
clc;
// Example 25.7
printf('Example 25.7\n\n');
//page no. 777
// Solution

// Given
//Bhopal Process
//CH3NH2 + COCl2 + --> C2H3NO +2HCl                    (A)
//C2H3NO + C10H8O --> C12H11O2N                        (B)
//Alternate process
//C10H8O + COCl2 --> C11H7O2Cl                         (C)
//C11H7O2Cl + CH3NH2 --> C12H11O2N + HCl        (D)
```

```

H_Cb = -26 ;//Standard heat of formation of carbaryl(C12H11O2N) -[kJ/ g mol]
H_HCl = -92.311 ;//Standard heat of formation of HCl -[kJ/ g mol]
H_Ma = -20.0 ;//Standard heat of formation of methyl amine(CH3NH2) -[kJ/ g mol]
H_Mi = -9*10^4 ;//Standard heat of formation of methyl isocynate(C2H3NO) -[kJ/ g mol]
H_Nc = -17.9 ;//Standard heat of formation of 1-Napthalenyl chloroformate(C11H7O2Cl) -[kJ/ g mol]
H_N = 30.9 ;//Standard heat of formation of napthol(C10H8O) -[kJ/ g mol]
H_P = -221.85 ;//Standard heat of formation of phosgene(COCl2) -[kJ/ g mol]

H_rxn_a = (2*H_HCl + 1*H_Mi) - (1*(H_Ma) + 1*H_P ) ;// Heat of reaction (A)-[kJ]
H_rxn_b = (1*H_Cb ) - (1*(H_Mi) + 1*H_N )  ;// Heat of reaction (B)-[kJ]
H_rxn_c = (1*H_Nc) - (1*(H_N) + 1*H_P )  ;// Heat of reaction (C)-[kJ]
H_rxn_d = (1*H_Cb + 1*H_HCl) - (1*(H_Nc) + 1*H_Ma ) ;// Heat of reaction (D)-[kJ]

//Bhopal Process
printf(' Bhopal process .\n')  ;
printf('   (a) Heat of reaction (A) is %.1e kJ.\n',H_rxn_a) ;
printf('   (b) Heat of reaction (B) is %.1e kJ.\n',H_rxn_b) ;

//Alternate process
printf('\n  Alternate process .\n')  ;
printf('   (c) Heat of reaction (C) is %.2f kJ.\n',H_rxn_c)  ;
printf('   (d) Heat of reaction (D) is %.2f kJ.\n',H_rxn_d)  ;
```

```
printf(' \nThe above data show that capital cost of Bhopal process could be higher than alternate process.\n')  ;
```

Scilab code Exa 25.8 Calculation of the Heat of Reaction at a Temperature differen

```
clear ;
clc;
// Example 25.8
printf('Example 25.8\n\n');
//page no. 782
// Solution

// Given
P1 = 1 ;// Initial pressure - [atm]
P2 = 1 ;// Final pressure - [atm]
T1 = 500 ;// Initial temperature -[degree C]
T2 = 500 ;// Final temperature -[degree C]

// The main reaction is CO2(g) + 4H2(g) + --> 2H2O(g) + CH4(g)                                (A)
// Data obtained from above reaction
m_CO2 = 1 ;// Moles of CO2 - [ g mol]
m_H2 = 4 ;// Moles of H2 - [ g mol]
m_H2O = 2 ;// Moles of H2O - [ g mol]
m_CH4 = 1 ;// Moles of CH4 - [ g mol]

// Additional required data are obtained from CD, according to book it is a follows-
H_fCO2 = -393.250; // Heat of formation of CO2 - [kJ/g mol]
H_fH2 = 0 ;// Heat of formation of H2 - [kJ/g mol]
H_fH2O = -241.835 ;// Heat of formation of H2O - [kJ/g mol]
H_fCH4 = -74.848 ;// Heat of formation of CH4 - [kJ/
```

```
g mol]

H_CO2 = 21.425 ;// Change in enthalpy during temperature change from 25 to 500 degree C  of CO2 - [kJ/g mol]
H_H2 = 13.834 ;// Change in enthalpy during temperature change from 25 to 500 degree C of H2 - [kJ/g mol]
H_H2O = 17.010 ;// Change in enthalpy during temperature change from 25 to 500 degree C of H2O - [kJ/g mol]
H_CH4 = 23.126 ;// Change in enthalpy during temperature change from 25 to 500 degree C of CH4 - [kJ/g mol]

H_in  = (H_fCO2 + H_CO2)*m_CO2 + (H_fH2 + H_H2)*m_H2 ;// Enthalpy change for inputs -[kJ]
H_out = (H_fH2O + H_H2O)*m_H2O + (H_fCH4 + H_CH4)*m_CH4 ; // Enthalpy change for outputs -[kJ]
del_H = H_out - H_in ;// Net enthalpy change of process -[kJ]

printf('Heat of above reaction at 500 degree C and 1 atm is %.1f kJ.',del_H) ;
```

Scilab code Exa 25.9 `Calculation of the Heat Transfer when Reactants enter and Pro`

```
clear ;
clc;
// Example 25.9
printf('Example 25.9\n\n');
//page no. 783
// Solution

// Given
```

```
// The main reaction is CO(g,1 atm,298 K) + (1/2)O2(g,1 atm,400K)  --> CO2(g,1at,300 K)                                       (A)
// Conditions of input and output gases are shown in above reaction
m_CO = 1 ;// Moles of CO input- [g mol]
m1_O2 = 1.5 ;// Moles of O2 input - [g mol]
m_CO2 = 1 ;// Moles of CO2 output - [g mol]
m2_O2 = 1 ;// Moles of O2 output - [g mol]
T_in_CO = 298 ;// Temperature of entering CO -[K]
T_in_O2 = 400 ;//Temperature of entering O2 -[K]
T_out_CO2 = 300 ;// Temperature of exiting CO2 -[K]
T_out_O2 = 300 ;// Temperature of exiting O2 -[K]

// Additional data are obtained fro Appendix D and E , according to book it is as follows
H_fCO = -110.520 ;// Heat of formation of CO - [kJ/g mol]
H_fO2 = 0 ;// Heat of formation of O2 - [kJ/g mol]
H_fCO2 = -393.250 ;// Heat of formation of CO2 - [kJ/g mol]

H_CO = 0 ;// Change in enthalpy during temperature change from 298K to 298 K of CO - [kJ/g mol]
H1_O2 = 11.619 ;// Change in enthalpy during temperature change from 298K to 400 K of input O2 - [kJ/g mol]
H_CO2 = 11.644 ;// Change in enthalpy during temperature change from 298K to 300 K of CO2 - [kJ/g mol]
H2_O2 = 8.389 ;// Change in enthalpy during temperature change from 298K to 300 K of output O2 - [kJ/g mol]

H_in  = (H_fCO + H_CO)*m_CO + (H_fO2 + H1_O2)*m1_O2 ;// Enthalpy change for inputs -[kJ]
H_out  = (H_fCO2 + H_CO2)*m_CO2 + (H_fO2 + H2_O2)*m2_O2 ;// Enthalpy change for inputs -[kJ]
```

```
del_H = H_out - H_in ;// Net enthalpy change of process -[kJ]

printf('Heat of above reaction  is %.1f kJ.',del_H);
```

Scilab code Exa 25.10 `Heating Value of Coal`

```
clear ;
clc;
// Example 25.10
printf('Example 25.10\n\n');
//page no. 788
// Solution

// Given
Ex_hv = 29770.0 ;// Experimental heating value of given coal - [kJ/kg]
// Coal analysis

C = 71.0/100 ;//Fraction of C in coal
H2 = 5.6/100 ;// Fraction of H2 in coal
N2 = 1.6/100 ;// Fraction of N2 in coal
S = 2.7/100 ;// Fraction of S in coal
ash = 6.1/100 ;// Fraction of ash in coal
O2 = 13.0/100 ;//Fraction of O2 in coal

//Higher heating value (HHV) by Dulong formula
HHV = 14544*C + 62028*(H2 - O2/8) + 4050*S ;// Higher heating value (HHV) by Dulong formula -[Btu/lb]
HHV_SI = HHV *1.055/0.454 ;// HHV in SI unt - [kJ/kg]

printf('The experimental heating value -
```

```
                              %.0f kJ.\n',Ex_hv) ;
printf(' Higher heating value (HHV) by Dulong formula -          %.0f kJ.\n',HHV_SI) ;
printf('\n The two values are quite close.' ) ;
```

Scilab code Exa 25.11 Selecting a Fuel to reduce SO2 emissions

```
clear ;
clc;
// Example 25.11
printf('Example 25.11\n\n');
//page no. 789
// Solution

// Given
H_req = 10^6 ;;// Heat requirement - [Btu]

d_N6 = 60.2 ;// Density of fuel no. 6-[lb/ft^3]
d_N2 = 58.7 ;// Density of fuel no. 2-[lb/ft^3]
S_N6 = 0.72/100 ;// Sulphur content in fuel no. 6
S_N2 = 0.62/100; //Sulphur content in fuel no. 2
lhv_N6 = 155000 ;//Lower heating value of  No.6 -[Btu/gal]
lhv_N2 = 120000 ;//Lower heating value of  No.2 -[Btu/gal]

S1 = H_req*d_N6*S_N6/lhv_N6 ;// Sulphur emmited when we use fuel NO. 6-[lb]
S2 = H_req*d_N2*S_N2/lhv_N2 ;// Sulphur emmited when we use fuel NO. 2-[lb]

printf('\n Sulphur emmited when we use fuel NO. 6 is %.2f lb.',S1 ) ;
printf('\n Sulphur emmited when we use fuel NO. 2 is %.2f lb.\n',S2 ) ;
```

```
printf('Clearly fuel no. 6 should be selected because of its low SO2 emmission.');
```

Chapter 26

Energy Balances that include the Effects of Chemical Reaction

Scilab code Exa 26.1 `Analysis of the Degrees of Freedom for a Combustion Process`

```
clear ;
clc;
// Example 26.1
printf('Example 26.1\n\n');
//page no. 804
// Solution

printf('Table to carry out degree of freedom analysis:\n');
// Number of variables involved
printf('\nI. Number of variables involved.\n');
printf('    Species in F1                                        1 \n');
printf('    Species in F2                                        2\n');
```

```
printf('    Specie in F3
   5\n');
printf('    Total stream flows
   3\n');
printf('    Stream temperatures
   3\n');
printf('    Stream pressures
   3 \n');
printf('    Q
   1 \n');
printf('    Extent of reactions
   2\n');
printf('\
   ---------------------------------------------------------------
   \n');
printf('  Total
   20\n');
printf('\n\nII. Number of equations and
   specifications.\n');
printf('    Independent species material balances
                                                          6\n')
   ;
printf('    Sum of species in each of the two
   streams                                                    2
    \n');
printf('    Energy balance
   1\n');
printf('    Total stream flows
   2\n');
```

```
printf('    Species values(CO)                                    1\n');
printf('    Pressures                                             3 \n');
printf('    Temperatures                                          2 \n');
printf('    O2 to N2 ratio specified in F2                        1 \n');
printf('    Complete reaction, hence the extent of reaction is implied to both reactions      2\n');
printf('\-----------------------------------------------------------------\n');
printf('  Total                                                   20\n');
printf('\n Therefore, by analysing the above table it is clear that degrees of freedom of system is (20 - 20) = 0 \n');
```

Scilab code Exa 26.2 `Calculation of Flame Temperature`

```
clear ;
clc;
// Example 26.2
printf('Example 26.2\n\n');
//page no. 808
// Solution

// Given
// The main reaction is CO(g,1 atm,100 C) + (1/2)O2(
```

```
g,1 atm,100 C)  --> CO2(g,1at,T K)                              (A)
// Input compounds
m1_CO = 1; // Moles of CO input- [g mol]
m1_O2 = 1 ;// Moles of O2 input - [g mol]
m1_N2 = 3.76 ;// Moles of N2 input - [g mol]
//Output compounds
m2_CO2 = 1 ;// Moles of CO2 output - [g mol]
m2_O2 = .50 ;// Moles of O2 output - [g mol]
m2_N2 = 3.76 ;// Moles of N2 output - [g mol]

// Additional data is obtained from Appendix D, according to book it is as follows:
// Inputs
H1_fCO = -110520 ;// Heat of formation of CO - [J/g mol]
H1_fO2 = 0 ;// Heat of formation of O2 - [J/g mol]
H1_fN2 = 0 ;// Heat of formation of N2 - [J/g mol]
H1_CO = 2917 - 728 ;// Change in enthalpy during temperature change from 298K to 373 K of CO - [J/g mol]
H1_O2 = 2953 - 732 ;// Change in enthalpy during temperature change from 298K to 373 K of input O2 - [J/g mol]
H1_N2 = 2914 - 728 ;// Change in enthalpy during temperature change from 298K to 373 K of input N2 - [J/g mol]

H_in  = (H1_fCO + H1_CO)*m1_CO + (H1_fO2 + H1_O2)*m1_O2 + (H1_fN2 + H1_N2)*m1_N2;// Enthalpy change for inputs -[J]

//Outputs - Assume it to be at 2000 K
H2_fCO2 = -393510 ;// Heat of formation of CO2 - [J/g mol]
H2_fO2 = 0 ;// Heat of formation of O2 - [J/g mol]
H2_fN2 = 0 ;// Heat of formation of N2 - [J/g mol]
H2_CO2 = 92466 - 912 ;// Change in enthalpy during
```

```
temperature change from 298K to 2000 K of CO2 - [J/g mol]
H2_O2 = 59914-732 ;// Change in enthalpy during temperature change from 298K to 2000 K of output O2 - [J/g mol]
H2_N2 = 56902 - 728 ;// Change in enthalpy during temperature change from 298K to 2000 K of output O2 - [J/g mol]

H1_out  = (H2_fCO2 + H2_CO2)*m2_CO2 + (H2_fO2 + H2_O2)*m2_O2 + (H2_fN2 + H2_N2)*m2_N2 ;// Enthalpy change for outputs at 2000 K -[J]

del_H1 = H1_out - H_in ;// Net enthalpy change of process -[J]

//Output- Assume it to be at 1750 K
H2_fCO2 = -393510 ;// Heat of formation of CO2 - [J/g mol]
H3_CO2 = 77455 - 912 ;// Change in enthalpy during temperature change from 298K to 1750 K of CO2 - [J/g mol]
H3_O2 = 50555 -732 ;// Change in enthalpy during temperature change from 298K to 1750 K of output O2 - [J/g mol]
H3_N2 = 47940 - 728 ;// Change in enthalpy during temperature change from 298K to 1750 K of output O2 - [J/g mol]

H2_out  = (H2_fCO2 + H3_CO2)*m2_CO2 + (H2_fO2 + H3_O2)*m2_O2 + (H2_fN2 + H3_N2)*m2_N2 ;// Enthalpy change for outputs at 1750 K -[J]

del_H2 = H2_out - H_in ;// Net enthalpy change of process -[J]

printf('Heat of above reaction when output is assumed to be at 2000 K  is %.0f J.\n',del_H1) ;
```

```
printf(' Heat of above reaction  when output is assumed to be at 1750 K  is %.0f J.\n',del_H2) ;
// Energy balance here reduce to del_H = 0
printf('\n So we can see that our desired result del_H = 0 is bracketed between 2000 K and 1750 K, hence we will use interpolation to get the theoretical flame temperature.\n') ;
// Use interpolation to get the theoretical flame temperature
del_H = 0 ;// Requred condition
Ft = 1750 + ((del_H - del_H2)/(del_H1 - del_H2))*(2000 - 1750) ;// Interpolation to get Flame temperature(Ft)-[K]
printf(' Theoretical flame temperature by interpolation is %.0f K.\n',Ft) ;
```

Scilab code Exa 26.3 `Application of general Energy Balance in a Process in which M`

```
clear ;
clc;
// Example 26.3
printf('Example 26.3\n\n');
//page no. 811
// Solution Fig E26.3b

// Given
v_CH4 = 1000 ;// Volume of CH4 taken - [ cubic feet]
CH4 = 1 ;// assumed for convenience- [ g mol]
ex_air = .5 ;// Fraction of excess O2 required
hp_CaCO3 = 0.130 ;// Heat capacity of CaCO3 -[kJ/g mol]
hp_CaO = 0.062 ;// Heat capacity of CaO -[kJ/g mol]
w_CaCO3 = 100.09 ;// Mol. wt. of CaCO3 -[g]
w_CaO = 56.08 ;// Mol. wt. ofCaO - [g]

```

```
// The main reaction are -
// (a)  CaCO3(s,25 C) --> CaO(s,900 C) + CO2(g,500 C)
//  (b) CH4(g,25 C) + 2O2(g,25 C) --> CO2 (g,500 C) + 2H2O (g,500 C)

req_O2 = 2 ;// By eqn. (b), O2 required by CH4 - [g mol]
ex_O2 = ex_air*req_O2 ;// Excess O2 required - [ g mol]
O2 = req_O2 + ex_O2 ;// Total O2 entering - [ g mol]
N2 = O2 *(.79/.21) ;// Total N2 entering - [ g mol]

// By analysis DOF is zero.

// Carry out elemental balance to get the unknowns
nG_N2 = N2 ;// N2 balance - [ g mol]
nG_H2O = 4*CH4/2 ;// H2O balance - [ g mol]
nG_O2 = ex_O2 ;// [g mol]
// L = P , from Ca balance ...eqn. (A)
// 1 + L = nG_CO2 , from C balance ...eqn. (B)
// 3L + 2*O2 = 2*nG_CO2 + 2*nG_O2 + nG_H2O + P , from O balance ... eqn. (C)

// For energy balance, get required data from software in the CD of book and sensible heat data from Appendix F
// given data of outputs is taken in array in order CO2(g), O2(g),N2(g),H2O(g)  and then CaO(s)
del_Hi_out = [ -393.250,0,0,-241.835,-635.6] ;// // Heat of formation  - [kJ/g mol]
del_Hf_out = [21.425,15.043,14.241,17.010,54.25] ;// Change in enthalpy during temperature change -[kJ/g mol]
del_H_out =del_Hi_out + del_Hf_out ;// Change in enthalpy final - [kJ/g mol]

// given data of inputs is taken in array in order
```

```
CH4(g), CaCO3(s),O2(g) and N2(g)
del_Hi_in = [ -49.963,-1206.9,0,0] ;// // Heat of formation - [kJ/g mol]
del_Hf_in = [0,0,0,0] ;//Change in enthalpy during temperature change -[kJ/g mol]
del_H_in = del_Hi_in + del_Hf_in ;// Change in enthalpy final - [kJ/g mol]
// Now do energy balance , assume Q = 0 ,
// del_H_out(1)*nG_CO2 + del_H_out(2)*nG_O2 + del_H_out(3)*nG_N2 + del_H_out(4)*nG_H2O + del_H_out(5)*P = del_H_in(1)*CH4 + del_H_in(2)*L ... eqn. (D)
// Solve eqn. (A), (B), (C), and (D) to get L ,P , nG_CO2
a = [1 -1;(del_H_in(2)-del_H_out(5)) -del_H_out(1)] ;// Matrix of coefficients
b = [-1;(del_H_out(2)*nG_O2 + del_H_out(3)*nG_N2 + del_H_out(4)*nG_H2O-del_H_in(1)*CH4)] ;// Matrix of constants
x = a\b  ;// Matrix of solutions, L = x(1), nG_CO2 = x(2)
g_CaCO3 = x(1)*w_CaCO3 ;//CaCO3 processed for each g mol of CH4 - [g]
printf(' CaCO3 processed for each g mol of CH4 is %.0f g.\n',g_CaCO3) ;
m_CaCO3 = (v_CH4*g_CaCO3)/359.05 ;
printf(' Therefore, CaCO3 processed per 1000 ft^3 of CH4 is %.0f lb.\n',m_CaCO3) ;
```

Scilab code Exa 26.4 Application of general Energy Balance in a Process Composed o

```
clear;
clc;
// Example 26.4
printf('Example 26.4\n\n');
```

```
//page no. 815
// Solution Fig E26.4b

// Given
SO2_in = 2200 ;// Amount of SO2 entering reactor 2-[lb mol/hr]
// Basis : 1 lb mol CO entering reactor 1,therefore
R1_CO_in = 1 ;//CO entering reactor 1-[lb mol]
air =  .80 ;// Fraction of air used in burning

// System- reactor 2
// Given
R2_fSO2_in = 0.667 ;// Fraction of SO2 entering reactor 2
R2_fO2_in = 0.333 ;// Fraction of O2 entering reactor 2
R2_fSO3_out = 0.586 ;// Fraction of SO3 exiting reactor 2
R2_fSO2_out = 0.276 ;// Fraction of SO2 exiting reactor 2
R2_fO2_out = 0.138 ;// Fraction of O2 exiting reactor 2
// Main Reaction: CO , (1/2)*O2 ---> CO2
R1_O2_in = (1/2)*air ;//  O2 entering reactor 1-[g mol]
R1_N2_in = R1_O2_in*(79/21) ;//  N2 entering reactor 1-[g mol]

//Output of reactor 1
R1_CO_out = R1_CO_in*(1 - air) ;// [g mol]
R1_CO2_out = 1*( air) ;// [g mol]
R1_N2_out = R1_N2_in ;//[g mol]

// By analysis DOF is zero.
// Get eqn. to solve by species balance
//Unknowns - P- exit stream of reactor 2 , F - entry stream of reactor 2 , ex - extent of reaction
// P*(R2_fSO2_out)  - F*0 = 1*ex ... eqn.(a)- By SO3
```

```
balance
// P*(R2_fSO2_out) - F*(R2_fSO2_in) = -1*ex ...eqn.(b) - By SO2 balance
// By O2 balance we will get eqn. equivalent to eqn. (b), so we need  one more eqn.

// Energy balance
// For energy balance, get required data from software in the CD of book and sensible heat data from Appendix F
// given data of outputs is taken in array in order CO(g),CO2(g), N2(g),SO2(g),SO3(g)  and then O2(g)
del_Hi_out = [ -109.054,-393.250,0,-296.855,-395.263,0] ; // Heat of formation  - [kJ/g mol]
del_Hf_out = [35.332,35.178,22.540,20.845,34.302,16.313] ;// Change in enthalpy during temperature change -[kJ/g mol]
del_H_out =del_Hi_out + del_Hf_out ;// [-371.825,15.043,160.781,-449.650,-581.35]// Change in enthalpy final - [kJ/g mol]

// given data of inputs is taken in array in order CO(g),CO2(g), N2(g),SO2(g) and then O2(g)
del_Hi_in = [  -109.054,-393.250,0,-296.855,0] ;// // Heat of formation  - [kJ/g mol]
del_Hf_in = [17.177,17.753,11.981,0,0] ;//Change in enthalpy during temperature change -[kJ/g mol]
del_H_in = del_Hi_in+ del_Hf_in ;// Change in enthalpy final - [kJ/g mol]
// Now do energy balance , assume Q = 0 ,
// del_H_out(4)*P*R2_fSO2_out + del_H_out(5)*P*R2_fSO3_out - del_H_in(4)*F*R2_fSO2_in + del_Hi_out(6)*P*R2_fO2_out = 0 ... eqn. (c)

// Solve eqn. (a), (b) and (c) to get F ,P , ex
a = [(R2_fSO3_out) 0 -1;(R2_fSO2_out) -(R2_fSO2_in)
```

```
1;(del_H_out(4)*R2_fSO2_out + del_H_out(5)*R2_fSO3_out + del_Hi_out(6)*R2_fO2_out )  -(del_H_in(4)*R2_fSO2_in) 0] ;// Matrix of coefficients
b = [0;0;(del_H_in(1)*R1_CO_out+del_H_in(2)*R1_CO2_out+del_H_in(3)*R1_N2_out-(del_H_out(1)*R1_CO_out+del_H_out(2)*R1_CO2_out+ del_H_out(3)*R1_N2_out))] ;// Matrix of constants
x = a\b  ;// Matrix of solutions , P = x(1) , F = x(2) ,ex = x(3)
F = x(2) ;//exit stream of reactor 2 - [lb mol]
R2_SO2_in = R2_fSO2_in*F ;// Moles of SO2 required per lb mol of CO - [lb mol]
CO = (R1_CO_in*SO2_in)/R2_SO2_in ;//Mole of CO burned in reactor 1 - [lb mol]

printf('Mole of CO burned in reactor 1 is %.0f lb mol.\n',CO) ;
```

Scilab code Exa 26.5 Production of Citric Acid by Fungus

```
clear ;
clc;
// Example 26.5
printf('Example 26.5\n\n');
//page no. 819
// Solution

// Given
CA = 10000 ;// Produced citric acid - [kg]
f_glucose = .30 ;// Fraction of glucose in solution
con_glucose = .60 ;//  Fraction of glucose consumed
w_glucose = 180.16 ;// Mol. wt. of d,alpha glucose -[g]
H_glucose = -1266 ;// Specific enthalpy change of
```

```
glucose - [kJ/g mol]
w_CA = 192.12; // Mol. wt. of citric acid -[g]
H_CA = -1544.8  ;// Specific enthalpy change of citric acid - [kJ/g mol]
w_BM = 28.6 ;// Mol. wt. of biomass -[g]
H_BM = -91.4  ;//  Specific enthalpy change of biomass - [kJ/g mol]
H_CO2 = -393.51  ;//  Specific enthalpy change of CO2 - [kJ/g mol]

// Main reaction is :
// 3 * glucose + 7.8*O2 ---> 5.35*BM + 2.22*CA + 4.50*CO2 ..reaction (a)

// Material Balance
mol_CA = CA/w_CA ;// Mole of citric acid produced - [kg mol]
g_soln = (mol_CA*(3/2.22)*w_glucose*1)/(con_glucose*f_glucose) ;//Mass of 30 % glucose solution introduced -[kg]
i_glucose = g_soln* f_glucose / w_glucose ;// Initial moles of glucose - [kg mol]
f_glucose = (1 - con_glucose)*i_glucose ;// Final moles of glucose - [kg mol]
f_CA = mol_CA ;// Final moles of citric acid - [kg mol]
f_BM = f_CA*(5.35/2.22) ;// Using the reaction (a)- Final moles of biomass - [kg mol]
i_O2 = i_glucose*(7.8/3) ;//  Using the reaction (a) - Initial moles of O2 - [kg mol]
f_CO2 =  i_glucose*(4.5/3)*con_glucose ;//  Using the reaction (a) - Final moles of CO2 - [kg mol]

// Energy balance
// For closed system - del_U = Q + W
power = 100 ;// Power of aerator -[hp]
time = 220 ;// Time taken for reaction - [ hr ]
W = (power*745.7*time*3600)/1000 ;// Work done by
```

```
aerator - [kJ]

// Assume del_U = del_H , pv work is equal to zero , hence
// Q = del_H - W

Hi_glucose  = i_glucose*H_glucose*1000 ;// Enthalpy change of glucose input - [kJ]
Hi_O2  = i_O2*0*1000 ;// Enthalpy change of O2 input - [kJ]
H_in = Hi_glucose + Hi_O2 ;// Enthalpy change of input - [kJ]

Hf_glucose  = f_glucose*H_glucose*1000 ;// Enthalpy change of glucose output - [kJ]
Hf_BM = f_BM * H_BM*1000 ;//Enthalpy change of biomass output - [kJ]
Hf_CA = f_CA *H_CA*1000 ;//Enthalpy change of citric acid output - [kJ]
Hf_CO2 = f_CO2 *H_CO2*1000 ;//Enthalpy change of CO2 output - [kJ]
H_out = Hf_glucose + Hf_BM +Hf_CA + Hf_CO2 ;// Enthalpy change of output - [kJ]
del_H = H_out - H_in ;// Total enthalpy change in process - [kJ]
Q = del_H - W ;// Heat removed - [kJ]

printf('Heat exchange from the fermentor during production of 10,000 kg citric acid is %.2e kJ(minus sign indicates heat is removed).\n',Q) ;
```

Chapter 27

Ideal Processes Efficiency and the Mechanical Energy Balance

Scilab code Exa 27.1 `Calculation of the Work done during Evaporation of a Liquid`

```
clear ;
clc;
// Example 27.1
printf('Example 27.1\n\n');
//page no. 838
// Solution E27.1

// Given
V_w = 1 ;//  Volume of given water -[L]
P_atm = 100 ;// Atmospheric pressure - [kPa]

//W = -p*del_V
V_H2O = 0.001043 ;// Specific volume of water from
   steam table according to book- [cubic metre]
V_vap = 1.694 ;// Specific volume of vapour from
   steam table according to book- [cubic metre]
V1 = 0 ;// Initial volume of H2O in bag-[cubic metre
   ]
V2 = (V_w*V_vap)/(1000*V_H2O) ;// Final volume of
```

```
water vapour -[cubic metre]
W = -P_atm*(V2 -V1)* 1000 ;// Work done by saturated liquid water -[J]

printf(' Work done by saturated liquid water is %.3e J.\n',W) ;
```

Scilab code Exa 27.2 `Calculation of Work in a Batch Process`

```
clear ;
clc;
// Example 27.2
printf('Example 27.2\n\n');
//page no. 840
// Solution E27.2

// Given
m_N2 = 1 ;// Moles of N2 taken -[kg mol]
p = 1000;// Pressure of cylinder -[kPa]
T = 20 + 273 ;// Temperature of cylinder -[K]
a_pis = 6 ;// Area of piston - [square centimetre]
m_pis = 2 ;// Mass of pston - [kg]
R = 8.31 ;// Ideal gas constant - [(kPa*cubic metre)/(K * kgmol)]

V = (R*T)/p ;// Specific volue of gas at initial stage -[cubic metre/kg mol]
V1 = V * m_N2 ;// Initial volume of gas - [cubic metre]
V2 = 2*V1 ;// Final volume of gas according to given condition -[cubic metre]

// Assumed surrounding pressure constant = 1 atm
p_atm = 101.3 ;// Atmospheric pressure -[kPa]
del_Vsys = V2 -V1 ;// Change in volume of system -[
```

```
    cubic metre]
del_Vsurr = - del_Vsys ;// Change in volume of
    surrounding -[cubic metre]
W_surr = -p_atm*del_Vsurr ;// Work done by
    surrounding - [kJ]
W_sys = -W_surr ;// Work done by system - [kJ]

printf(' Work done by gas(actually gas + piston
    system) is %.0f kJ.\n',W_sys) ;
```

Scilab code Exa 27.3 Efficiency of Power Generation by a Hydroelectric Plant

```
clear ;
clc;
// Example 27.3
printf('Example 27.3\n\n');
//page no. 845
// Solution

// Given
p_plant = 20 ;// Power generated by plant -[MW]
h = 25 ;// Height of water level - [m]
V = 100 ;// Flow rate of water -[cubic metre/s]
d_water = 1000 ;// Density of water - [ 1000 kg /
    cubic metre]
g = 9.807 ;// Acceleration due to gravity -[m/square
    second]

M_flow = V*d_water ;// Mass flow rate of water -[kg/
    s]
del_PE = M_flow*g*h ;// Potential energy change of
    water per second -[W]
eff = (p_plant*10^6) /(del_PE) ;// Efficiency of
    plant

```

```
printf(' Efficiency of plant is %.2f .\n',eff) ;
```

Scilab code Exa 27.4 `Calculation of Plant Efficiency`

```
clear ;
clc;
// Example 27.4
printf('Example 27.4\n\n');
//page no. 845
// Solution Fig.E27.4

// Given
LHV = 36654 ;// LHV value of fuel - [kJ/ cubic metre]
Q1 = 16 ;//- [kJ/ cubic metre]
Q2 = 0 ;//- [kJ/ cubic metre]
Q3 = 2432 ;//- [kJ/ cubic metre]
Q4 = 32114 ;//- [kJ/ cubic metre]
Q41 = 6988 ;//- [kJ/ cubic metre]
Q8 = 1948 ;//- [kJ/ cubic metre]
Q9 = 2643 ;//- [kJ/ cubic metre]
Q81 = 2352 - Q8 ;// - [kJ/ cubic metre]
Q567 = 9092 ;// Sum of Q5, Q6 and Q7- [kJ/ cubic metre]

//(a)
G_ef = (LHV+ Q1 +Q2 + Q3 - Q9)/(LHV) ;// Gross efficiency
printf('(a) Gross efficiency is %.3f .\n',G_ef) ;

//(b)
T_ef = (Q567+Q8)/(LHV+ Q1 +Q2 + Q3) ;//Thermal efficiency
printf(' (b) Thermal efficiency is %.3f .\n',T_ef) ;

```

```
//(c)
C_ef = Q4/(Q4 + Q41) ;// Combustion efficiency
printf(' (c) Combustion efficiency is %.3f .\n',C_ef) ;
```

Scilab code Exa 27.5 Comparison of the Reversible Work for a Batch Process with th

```
clear ;
clc;
// Example 27.5
printf('Example 27.5\n\n');
//page no. 850
// Solution

// Given
V1 = 5 ;// Volume of gas initially - [cubic feet]
P1 = 1 ;// Initial pressure - [atm]
P2 = 10 ;// Final pressure - [atm]
T1 = 100 + 460 ;// initial temperature - [degree Rankine]
R = 0.7302 ;// Ideal gas constant -[(cubic feet*atm)/(lb mol)*(R)]
//Equation of state pV^1.4 = constant

//(a)
//Energy balance reduces to del_E = del_U = del_W
V2 = V1*(P1/P2)^(1/1.4) ;// Final volume - [cubic feet]
W1_rev = integrate('-(P1)*(V1/V)^(1.4)','V',V1,V2) ; // Reversible work done in compresion in a horizontal cylinder with piston -[cubic feet *atm]
W1 = W1_rev *1.987/.7302 ;// Conversion to Btu -[Btu]

```

```
printf('\n (a)Reversible work done in compression in a horizontal cylinder with piston is %.1f Btu .\n ',W1);

//(b)
n1 = (P1*V1)/(R*T1) ;// Number of moles of gas
W2_rev = integrate('(V1)*(P1/P)^(1/1.4)','P',P1,P2) ;// Reversible work done in compresion in a rotary compressor -[cubic feet *atm]
W2 = W2_rev *1.987/.7302 ;// Conversion to Btu -[Btu]

printf('\n (b)Reversible work done in a rotary compressor is %.1f Btu .\n ',W2);
```

Scilab code Exa 27.6 `Application of the Mechanical Energy Balance to the Pumping o`

```
clear ;
clc;
// Example 27.6
printf('Example 27.6\n\n');
//page no. 853
// Solution

// Given
m_water = 1 ;// Mass flow rate of water -[lb/min]
P1 = 100 ;// Initial pressure - [psia]
P2 = 1000 ;// Final pressure - [psia]
T1 = 80 + 460 ;// initial temperature - [degree Rankine]
T2 = 100 + 460 ;// final temperature - [degree Rankine]
h = 10 ;// Difference in water level between entry and exit of stream -[ft]
g = 32.2 ;// Accleration due to gravity - [ft/
```

```
square second]
gc = 32.2 ;//[(ft*lbm)/(lbf*square second)]

// The mechanical energy balance reduces to W = PV_work + del_PE ....(A)
// From steam table , specific volume of liquid water at 80 and 100 degree F is noted , according to book it is as follows-
v1 = .01607 ;// specific volume of liquid water at 80 degree F -[cubic feet/lbm]
v2 = .01613 ;// specific volume of liquid water at 100 degree F -[cubic feet/lbm]
// But for pratical purposes wwater is taken to be incompressible and specific volume can be taken as v, ith following value
v= 0.0161 ;// -[cubic feet/lbm]

del_PE = (h*g)/(gc*778) ;// Change in potential energy - [Btu/lbm]
PV_work = integrate('(v)*(12^2/778)','P',P1,P2) ;// PV work done -[Btu/lbm]
//From eqn. (A)
W = PV_work + del_PE ;// Work per minute required to pump 1 lb water per minute - [Btu/lbm]

printf('\n Work per minute required to pump 1 lb water per minute is %.2f Btu/lbm .\n ',W);
```

Chapter 28

Heats of Solution and Mixing

Scilab code Exa 28.1 `Application of Heats of Solution data`

```
clear ;
clc;
// Example 28.1
printf('Example 28.1\n\n');
//page no. 869
// Solution

// Given
Ref_T = 77 ;//Reference temperature -[degree F]

//(a)
mol_NH3 = 1 ;// Moles of NH3 - [lb mol]
mw_NH3 = 17 ;//Molecular t. of NH3 -[lb]
mw_H2O = 18  ;//Molecular t. of H2O -[lb]
f1_NH3 = 3/100 ;// Fraction of NH3 in solution
m_H2O = (mw_NH3/f1_NH3) - mw_NH3 ;// Mass of water
   in solution -[lb]
mol_H2O = m_H2O/mw_H2O  ;// Moles of H2O in solution
    -[lb mol]

printf('(a)  Moles of H2O in solution is %.1f  lb
```

```
mol .\n ',mol_H2O);
printf('     As we can see that moles of water is 30 lb mol(approx), hence we will see H_soln from table corresponding to 30 lb mol water .\n ');
H_soln = -14800 ;// From table given in question in book -[Btu/lb mol NH3]
printf('     The amount of cooling needed is, %.0f Btu heat removed.\n ',abs(H_soln));

//(b)
V = 100 ;// Volume of solution produced -[gal]
f2_NH3 = 32/100 ;// Fraction of NH3 in solution
// From Lange's Handbook of chemistry additional data is obtained , according to book it is as follows -
sg_NH3 = .889 ;// Specific gravity of NH3
sg_H2O = 1.003 ;// Specific gravity of H2O
d_soln = sg_NH3*62.4*sg_H2O*100/7.48 ;// Density of solution - [lb / 100 gal]
NH3 = d_soln*f2_NH3/mw_NH3 ;// Mass of NH3 - [ lb mol/ 100 gal]
m1_H2O = (mw_NH3/f2_NH3) - mw_NH3 ;// Mass of water in solution -[lb]
mol1_H2O = m1_H2O/mw_H2O  ;// Moles of H2O in solution -[lb mol]

printf('\n (b)  Moles of H2O in solution is %.1f  lb mol .\n ',mol1_H2O);
printf('     As we can see that moles of water is 2 lb mol , hence we will see H_soln from table corresponding to 2 lb mol water .\n ');
H_soln = -13700 ;// From table given in question in book -[Btu/lb mol NH3]
total_H = abs(NH3*H_soln) ;// Total heat removed from solution -[Btu]
printf('     The amount of cooling needed is, %.0f Btu heat removed.\n ',total_H);
```

Scilab code Exa 28.2 `Application of Heat of Solution Data`

```
clear ;
clc;
// Example 28.2
printf('Example 28.2\n\n');
//page no. 872
// Solution

// Given
p = 100 ;// Mass of product - [kg]
f_HCl = 25/100 ;//Fraction of HCl in product
//Product analysis
HCl = f_HCl*p ;// Mass of HCl in product - [kg]
H2O = (1-f_HCl)*p  ;// Mass of H2O in product -[kg]
mw_HCl = 36.37 ;// Molecular weight of HCl -[kg]
mw_H2O = 18.02 ;// Molecular weight of H2O -[kg]
mol_HCl = HCl /mw_HCl ;// Moles of HCl - [kg mol]
mol_H2O = H2O /mw_H2O; // Moles of H2O - [kg mol]
total_mol = mol_HCl + mol_H2O ;// Total no. of moles -[kg mol]
mf_HCl = mol_HCl / total_mol  ;// mole fraction of HCl
mf_H2O = mol_H2O / total_mol ; // mole fraction of H2O
mr = mol_H2O/mol_HCl ;// Mole ratio of H2O to HCl
MW = mf_HCl*mw_HCl + mf_H2O*mw_H2O ;// Molecular t. of solution -[kg]
Ref_T = 25 ;//Reference temperature-[degree C]

// Energy balance reduces to Q = del_H
// Additional data is obtained from Table E.1 , according to book it is a follows -
mol1_HCl = total_mol ;// Moles of HCl // Moles of
```

```
HCl output -[g mol]
Hf1_HCl = -157753 ;// Heat of formation of HCl output-[J/ g mol HCl ]
Hf_HCl = -92311 ;// Heat of formation of HCl input-[J/ g mol HCl ]
Hf_H2O = 0 ;// Heat of formation of H2O input-[J/ g mol HCl ]
H1_HCl =  556 ;// Change in enthalpy during temperature change from 25 C to 35 C of HCl - [J/g mol]
H_HCl = integrate('(29.13 - 0.134*.01*T)','T',298,393) ;// Change in enthalpy during temperature change from 25 C to 120 C of HCl - [J/g mol]

H_H2O = 0 ;// Change in enthalpy during temperature change from 25 C to 25 C of H2O - [J/g mol]

H_in = (Hf_HCl + H_HCl)*mol_HCl  + (Hf_H2O + H_H2O)*mol_H2O ;// Enthalpy change of input -[J]
H_out = Hf1_HCl*mol_HCl +H1_HCl*mol1_HCl  ;// Enthalpy change of output -[J]

del_H = H_out - H_in ;// Net enthalpy change n process - [J]
Q = del_H; // By energy balance - [J]

printf('The amount of heat removed from the absorber by cooling water is, %.0f J.\n ',Q);
```

Scilab code Exa 28.3 Application of an Enthalpy Concentration Chart

```
clear ;
clc;
// Example 28.3
```

```
printf('Example 28.3\n\n');
//page no. 875
// Solution fig. 28.3

// Given
//Input analysis
soln1 = 600 ; // Mass flow rate of entering solution 1 -[lb/hr]
c1_NaOH = 10/100 ;// Fraction of NaOH in entering solution 1
T1 = 200 ;// Temperature at entry
soln2 = 400 ;// Mass flow rate of another solution 2 entering -[lb/hr]
c2_NaOH = 50/100 ;// Fraction of NaOH in another entering solution 2

// Additional data is obtained from steam table and NaOH-H2O enthalpy-concentration chart in Appendix I at given reference temperature (del_H = 0 , 32 degree F for pure water)
F = soln1 + soln2; // Mass flow rate of final solution - [lb/hr]

 // Material balance to get composition of final solution
F_NaOH =  c1_NaOH * soln1 + c2_NaOH * soln2 ;// Mass of NaOH in final solution -[lb]
F_H2O = F - F_NaOH ;// Mass of H2O in final solution -[lb]

// Enthalpy data from H-x chart , according to book it is as follows
H_soln1 = 152 ;// Specific enthalpy change for solution 1-[Btu/lb]
H_soln2 = 290 ;// Specific enthalpy change for solution 2-[Btu/lb]

// Energy balance
```

```
H_F = (soln1*H_soln1 + soln2*H_soln2)/F ;// Specific enthalpy change for final solution -[Btu/lb]

//(a)
printf(' (a) The final temperature of the exit solution from figure E28.3 using the obtained condition of final solution is 232 degree F \n');

//(b)
cF = F_NaOH*100/F; // Concentration of final solution -[wt % NaOH ]
printf(' (b) The concentration of final solution is %.0f wt.%% NaOH . \n',cF);

//(c)
// For fraction of H2O vapour . By interpolation , draw the tie line through the point x = .26 .H = 270 (make it parallel to 220 and 250 degree F line ). The final temperature of the exit solution from figure E28.3 using the obtained condition of final solution is 232 degree ; the enthalpy of the liquid at the bubble point at this temperature is about 175 Btu/lb . The enthalpy of saturated water vapour fro the steam table at 232 degree F is 1158 Btu/lb . Let x be the water vapour evaporated , therefore
x = (F*H_F - F*175)/(1158 - 175) ;// H2O evaporated per hour -[lb]

printf(' (c) H2O evaporated per hour is %.1f lb . \n',x);
```

Chapter 29

Humidity Charts and their Uses

Scilab code Exa 29.1 Determining Properties of Moist air from Humidity Chart

```
clear ;
clc;
// Example 29.1
printf('Example 29.1\n\n');
//page no. 895
// Solution fig. E29.1

// Given
DBT = 90 ;// Dry bulb temperature - [degree F]
WBT = 70 ;// Wet bulb temperature - [degree F]

//Get point A using DBT & WBT. Following information is obtained from humidity chart, fig. E29.1

printf('(a) The Dew point is located at point B or about 60 degree F, using constant humidity line.\n');
printf(' (b) By interpolation between 40%% and 30%% RH , you can find point A is at 37%% relative
```

```
humidity .\n');
printf(' (c) You can read humidity from the righthand ordinate as 0.0112 lb H2O/lb dry air .\n');
printf(' (d) By interpolation again between 14.0 cubic feet/lb and 14.5 cubic feet/lb lines , you can find humid volume to be 14.1 cubic feet/lb dry air.\n');
printf(' (e) The enthalpy value of saturated air with WBT 70 degree F is 34.1 Btu/lb dry air .\n');
```

Scilab code Exa 29.2 `Heating at constant Humidity`

```
clear ;
clc;
// Example 29.2
printf('Example 29.2\n\n');
//page no. 897
// Solution fig. E29.2

// Given
DBT1 = 38 ;// Initial dry bulb temperature - [degree C]
DBT2 = 86 ;// Final dry bulb temperature - [degree C]
RH1 = 49 ;// Relative humidity - [%]

//A is initial and B is final  point , see fig. E29.2 . Dew point is obtained graphically and it is 24.8 degree C,therefore

printf('The Dew point is unchanged in the process because humidity is unchanged, and it is located at 24.8 degree C.\n');
```

```

// Additional data is obtained from humidity chart , according to book data is as follows
A_Hsat = 90.0 ;// Enthalpy of saturation at point A-[kJ/kg]
A_dH = -0.5 ;//Enthalpy deviation -[kJ/kg]
A_Hact = A_Hsat + A_dH ;// Actual enthalpy at point A -[kJ/kg]
B_Hsat = 143.3 ;// Enthalpy of saturation at point B - [kJ/kg]
B_dH = -3.3 ;//Enthalpy deviation -[kJ/kg]
B_Hact = B_Hsat + B_dH ;// Actual enthalpy at point B -[kJ/kg]

// Energy balance reduces to Q = del_H
del_H = B_Hact - A_Hact ;// Total change in enthalpy - [kJ/kg]
v = 0.91 ;// Specific volume of moist air at point A -[cubic metre / kg]
Q = del_H/v ;// Heat added per cubic metre of inital moist air -[kJ]
printf('\n Heat added per cubic metre of inital moist air is %.1f kJ.\n',Q);
```

Scilab code Exa 29.3 Cooling and Humidification using a Water Spray

```
clear ;
clc;
// Example 29.3
printf('Example 29.3\n\n');
//page no. 898
// Solution fig. E29.3b

// Given
DBT1 = 40 ;// Initial dry bulb temperature - [degree
```

```
C]
DBT2 = 27 ;// Final dry bulb temperature - [degree C]

// Process is assumed to be adiabatic, therefore wet bulb temperature is constant
WBT1 = 22 ;// Initial wet bulb temperature - [degree C]
WBT2 = WBT1 ;// Final wet bulb temperature - [degree C]

//A is initial and B is final  point , see fig. E29.3b . Humidity is obtained from humidity chart, according to book the respective humidities are as follows
H_B = 0.0145 ;// Humidity at point B -[kg H2O/kg dry air]
H_A = 0.0093 ;// Humidity at point A -[kg H2O/kg dry air]
Diff = H_B - H_A ;// Moisture added in kg per kilogram of dry air going through humidifier -[kg H2O/kg dry air]

printf('Moisture added per kilogram of dry air going through humidifier is %.4f kg H2O.\n',Diff);
```

Scilab code Exa 29.4 `Combined Material and Energy Balance for a Cooling Tower`

```
clear ;
clc;
// Example 29.4
printf('Example 29.4\n\n');
//page no. 900
// Solution fig. E29.4

```

```
// Given
c_bl = 8.30 * 10^6 ;// Capacity of blower - [cubic feet/hr]
DBT_A = 80 ;// Initial dry bulb temperature of moist air - [degree F]
DBT_B = 95 ;// Final dry bulb temperature of exit air - [degree F]
WBT_A = 65 ;// Initial wet bulb temperature of moist air - [degree F]
WBT_B = 90 ;// Final wet bulb temperature of exit air - [degree F]
T1_H2O = 120 ;// Initial temperature of water - [degree F]
T2_H2O = 90 ;// Final temperature of water - [degree F]

//A is initial and B is final point , see fig. E29.4 . Humidity is obtained from humidity chart, according to book the respective humidities are as follows
H_A = 0.0098; // Humidity of air at A - [lb H2O / lb dry air]
H1_A = 69 ;// Humidity of air at A - [grains H2O / lb dry air]
delH_A = 30.05 - 0.12; // Enthalpy of entering air -[Btu/lb dry air]
v_A = 13.82 ;// Specific volume of entering air -[cubic feet/lb dry air]
H_B = 0.0297;// Humidity of air at B - [lb H2O / lb dry air]
H1_B = 208 ;// Humidity of air at B - [grains H2O / lb dry air]
delH_B = 55.93 - 0.10  ;// Enthalpy of exit air -[Btu/lb dry air]
v_B = 14.65 ;// Specific volume of exit air -[cubic feet/lb dry air]
Eq_A = c_bl /v_A ;// Entering dry air equivalent of capacity of blower -[lb dry air]
```

```

// Reference temperature for water stream is 32 degree F
del_H1_H2O = 1*(T1_H2O - 32) ;//Enthalpy of entering water -[Btu/lb H2O]
del_H2_H2O = 1*(T2_H2O - 32) ;//Enthalpy of exit water -[Btu/lb H2O]
tr_H2O = H_B - H_A ;// Transfer of water to air -[lb H2O / lb dry air]

// Energy balance around the entire process yields W -
W = (delH_B - del_H2_H2O*tr_H2O - delH_A)/(del_H1_H2O - del_H2_H2O) ;// Water entering tower - [lb H2O/lb dry air]
W1 = W - tr_H2O ;// Water leaving tower -[lb H2O/lb dry air]
Total_W1 = W1* Eq_A ;// Total water leaving tower -[lb/hr]

printf('Amount of water cooled per hour is %.2e lb/hr .\n',Total_W1);
```

Scilab code Exa 29.5 Drying of Chlorella

```
clear ;
clc;
// Example 29.5
printf('Example 29.5\n\n');
//page no. 902
// Solution fig. E29.5

// Given
W = 100 ;// Amount of entering water -[lb/hr]
H1 = .020 ;// Humidity of entering air -[lb H2O / lb
```

```
dry air]
T1 = 155 ;//Temperature of entering air -[degree F]
DTB = 110 ;// Dry bulb temperature of exit air -[degree F]
WTB = 100 ;// Wet bulb temperature of exit air -[degree F]

// Additional data is obtained from humidity chart, it is as follows
H2 = .0405 ;//Humidity of exit air -[lb H2O / lb dry air]

del_H = H2 - H1 ;// Change in humidity betwween two states -[lb H2O / lb dry air]
air_in = (W*1.02)/(del_H * 1)    ;// Amount of wet air entering -[lb]

mol_air = 29 ;// Molecular wt. of air -[lb]
Ref_T = 32 + 460 ;// Reference temperature - [degree R]
gi_T = 90 + 460; // Given temperature on which calculation is based - [degree R]
air = (air_in *359*gi_T)/( mol_air*Ref_T) ;// Air consumption of dryer at 90 degree F and 1 atm -[cubic feet]

printf('Air consumption of dryer at 90 degree F and 1 atm  is %.2e cubic feet .\n',air);
```