

Scilab Textbook Companion for
Basic Principles And Calculations In Chemical
Engineering
by D. M. Himmelblau And J. B. Riggs¹

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

May 20, 2016

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

Book Description

Title: 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	4
1 Dimensions units and their conversion	5
2 Moles Density and Concentration	12
3 Choosing a Basis	22
4 Temperature	28
5 Pressure	30
6 Introduction to Material Balances	34
7 A General Strategy for Solving Material Balance Problems	37
8 Solving Material Balance Problems for Single Units without Reaction	41
9 The Chemical Reaction Equation and Stoichiometry	50
10 Material Balances for Processes Involving Reaction	61
11 Material Balance Problems involving Multiple Units	80
12 Recycle Bypass Purge and the Industrial Application of Material Balance	92
13 Ideal Gases	102

14 Real Gases Compressibility	111
15 Real Gases Equations of State	115
16 Single Component Two Phase Systems	117
17 Two Phase Gas Liquid Systems Saturation Condensation and Vaporization	124
18 Two Phase Gas Liquid Systems Partial Saturation and Humidity	132
19 The Phase Rule and Vapor Liquid Equilibria	139
20 Liquid and Gases in Equilibrium with Solids	146
21 Energy Terminology Concepts and Units	149
22 Introduction to Energy Balances for Process without Reaction	155
23 Calculation of Enthalpy Changes	162
24 Applications of Energy Balances in the Absence of Chemical Reactions	175
25 Energy Balances How to Account for Chemical Reaction	192
26 Energy Balances that include the Effects of Chemical Reaction	208
27 Ideal Processes Efficiency and the Mechanical Energy Balance	221
28 Heats of Solution and Mixing	228
29 Humidity Charts and their Uses	234

List of Scilab Codes

Exa 1.1	Dimensions and Units	5
Exa 1.2	Conversion of Units	6
Exa 1.3	Nanotechnology	6
Exa 1.4	A conversion involving lbm and lbf	7
Exa 1.5	Conversion associated with Biological Materials	8
Exa 1.6	Dimensional Consistency	8
Exa 1.7	Dimensional Consistency	9
Exa 1.8	Retention of significant Figures	10
Exa 1.9	Micro dissection of DNA	10
Exa 2.1	Calculation of Molecular Weight	12
Exa 2.2	Use of Molecular Weight to Convert Mass to Moles	13
Exa 2.3	Use of Molecular Weight to Convert Moles to Mass	13
Exa 2.4	Calculation of Density	14
Exa 2.5	Calculation of Mass and Moles	15
Exa 2.6	Conversion between Mass Fraction and Mole Fraction	15
Exa 2.7	Nitrogen Requirements for the Growth of Cells	16
Exa 2.8	Use of ppm	17
Exa 2.9	Calculation of Mole Fraction and ppm from a Concentration	18
Exa 2.10	Evaluation of Alternate Processes for the Production of Methyl Methacrylate	19
Exa 3.1	Choosing a Basis	22
Exa 3.2	Choosing a Basis	23
Exa 3.3	Choosing a Basis for Cell Growth	24
Exa 3.4	Calculation of the Mass Fraction of the Components in Nanoparticles	25
Exa 3.5	Changing Bases	26
Exa 4.1	Temperature Conversion	28

Exa 4.2	Temperature Conversion	29
Exa 5.1	Pressure Conversion	30
Exa 5.2	Pressure Conversion	31
Exa 5.3	Vacuum Pressure Reading	31
Exa 5.4	Calculation of Pressure Difference	32
Exa 5.5	Pressure Conversion	32
Exa 6.1	A Material Balance for the blending of Gasoline	34
Exa 6.2	Concentration of cells using a Centrifuge	35
Exa 6.3	Discharge of Tank Residuals to the Environment	36
Exa 7.1	Understanding the Problem	37
Exa 7.2	Drawing a Sketch of a Mixing Process	38
Exa 7.3	Placing the unknown Information on the Diagram	38
Exa 7.4	Analysis of the Degrees of Freedom	38
Exa 7.5	Analysis of Degree of Freedom in the Production of Biomass	39
Exa 8.1	Extraction of Streptomycin from a Fermentation Broth	41
Exa 8.2	Separation of gases Using a Membrane	42
Exa 8.3	Overall analysis for a continuous Distillation Column	43
Exa 8.4	Mixing of Battery Acid	44
Exa 8.5	Drying	45
Exa 8.6	Crystallization	46
Exa 8.7	Hemodialysis	48
Exa 9.1	Balancing a reaction for a Biological Reaction	50
Exa 9.2	Use of Chemical Reaction to Calculate the Mass of Reactants given the Mass of Products	51
Exa 9.3	Application of Stoichiometry when more than one Reaction occurs	51
Exa 9.4	Calculation of extent of Reaction	53
Exa 9.5	Calculation of Limiting and Excess Reactants	54
Exa 9.6	Yields in the Reaction of Glucose to produce Ethanol	56
Exa 9.7	Selectivity in the Production of Nanotubes	56
Exa 9.8	Calculation of various terms Pertaining to Reaction	57
Exa 10.1	Reaction in which Fraction Conversion is specified	61
Exa 10.2	A reaction in which Fraction Conversion is to be Calculated	63
Exa 10.3	Material Balances for Process in which two Simultaneous Reactions Occur	65
Exa 10.4	Analysis of Bioreactor	67

Exa 10.5	Solution using Elemental Balances	69
Exa 10.6	Use of Elemental Balance to Solve a Hydrocracking Problem	73
Exa 10.7	Excesss Air	74
Exa 10.8	A Fuel Cell to generate Electricity from Methane	75
Exa 10.9	Combustion of Coal	76
Exa 11.1	Determination of the Number of Independent Material Balance in a process with Multiple Units	80
Exa 11.2	Material Balances for Multiple Units in which no Reaction occurs	82
Exa 11.3	Material Balance for problems Involving Multiple units and Reactions	84
Exa 11.4	Analysis of Sugar Recovery Process involving Multiple Serial Units	87
Exa 11.5	Production of a Hormone in connected Reactor	89
Exa 12.1	A Continuous Crystallizer involving a Recycle Stream	92
Exa 12.2	Recycle in a Process in Which Reaction Occurs	94
Exa 12.3	Recycle in a Process with a Reaction Occurring	95
Exa 12.4	A Bioreactor with Recycle	97
Exa 12.5	Bypass Calculation	98
Exa 12.6	Purge	99
Exa 13.1	Use of Standard Conditions to Calculate Volume from Mass	102
Exa 13.2	Calculation of R using the Standard Conditions	102
Exa 13.3	Application of Ideal Gas Law to Calculate Volume	103
Exa 13.4	Calculation of Gas Density	104
Exa 13.5	Calculation of Specific Gravity of a Gas	105
Exa 13.6	Calculation of the Partial Pressures of the Components in a Gas	106
Exa 13.7	Material Balance for a Process Involving Combustion	107
Exa 13.8	Material Balance without Reaction	109
Exa 14.1	Use of Compressiblity Factor to Calculate a Specific Volume	111
Exa 14.2	Use of Compressiblity Factor to Calculate Pressure	112
Exa 14.3	Calculation of Properties of Real Gas Mixture	113
Exa 15.1	Application of Van der Walls Equation to Calculate a Temperature	115
Exa 15.2	Solution of van der Waals Equation for V	116

Exa 16.1	Vaporization of Metals for Thin Film Deposition	117
Exa 16.2	Interpolating in Steam Tables	118
Exa 16.3	Extrapolation of Vapour Pressure data	120
Exa 16.4	Solvent selection based on OSHA PEL Limits and Potential Hazard	122
Exa 17.1	Calculation of Dew Point of the Products of Combustion	124
Exa 17.2	Condensation of Benzene from a Vapour Recovery Unit	125
Exa 17.3	Smokestack Emissions and Pollution	126
Exa 17.4	Material Balance involving Condensation	128
Exa 17.5	Vaporization to Saturate Dry air	129
Exa 17.6	Vaporization of a Hazardous Component of an Oil Slick	130
Exa 18.1	Application of Relative Humidity to Calculate the Dew Point	132
Exa 18.2	Calculation involving Various Partial Saturation Terms	133
Exa 18.3	Dehydration of Moist air	134
Exa 18.4	Humidification of Air	135
Exa 18.5	Condensation of Water from air	136
Exa 19.1	Applications of Phase Rule to Systems without Reaction	139
Exa 19.2	Application of the Phase Rule to Systems in which Reactions can Occur	140
Exa 19.3	Bubble Point Calculation	141
Exa 19.4	Flash Calculation for a Binary Liquid Mixture	143
Exa 19.5	Separation of Virus from a Culture	144
Exa 20.1	Fitting Adsorption Isotherms to Experimental Data .	146
Exa 20.2	Separation of Biochemicals by Solvent Extraction . . .	147
Exa 20.3	Combination of an Adsorption Isotherm with a Material Balance	147
Exa 21.1	Calculation of Mechanical Work by a Gas on a Piston Showing How the Path affects the Value of the Work .	149
Exa 21.2	Calculation of the Specific Kinetic Energy for a Flowing Fluid	150
Exa 21.3	Calculation of Potential Energy Change in Water . . .	151
Exa 21.4	Calculation of Internal Energy Change using the Heat Capacity	151
Exa 21.5	Calculation of Internal Energy Change using Different Paths	152
Exa 21.6	Calculation of the Change in Enthalpy by two Different Paths	153

Exa 21.7	Calculation of an Enthalpy Change	153
Exa 22.1	Application of the Energy Balance to a Closed System	155
Exa 22.2	Calculation of delta U using American Engineering Units	156
Exa 22.3	Energy Balance to analyze an Open Unsteady State System	157
Exa 22.4	Application of Energy Balance to an Open Steady State System	159
Exa 22.5	Calculation of Power needed to Pump Water	160
Exa 23.1	Graph Showing the Heat of Vaporization of Water . . .	162
Exa 23.2	Comparison of Various Sources to Estimate the Heat of Vapourization	163
Exa 23.3	Conversion of Units in a Heat Capacity Equation . . .	164
Exa 23.4	Fitting Heat Capacity Equation to Heat Capacity Data	165
Exa 23.5	Calculation of Change in Specific Enthalpy for a Gas Mixture using Heat Capacity Equations for Each Component	167
Exa 23.6	Calculation of the Change in Enthalpy for a Gas using Tabulated Enthalpy Values	169
Exa 23.7	Use of Steam Tables to Calculate Change in Enthalpy	170
Exa 23.8	Use of Steam Table when a Phase Change is involved to Calculate the final State of Water	172
Exa 23.9	Calculate Enthalpy Difference between two States by Pressure Enthalpy Chart for Butane	173
Exa 24.1	Simplification of General Energy Balance	175
Exa 24.2	Degree of Freedom Analysis Including an Energy Balance	177
Exa 24.3	Application of Energy Balance	179
Exa 24.4	Applications of Energy Balance to Plasma Etching . .	180
Exa 24.5	Energy Balance applied to a Batch Process	181
Exa 24.6	Applications of Energy Balance to Pumping Water . .	185
Exa 24.7	Applications of Energy Balance to Heating a Biomass	186
Exa 24.8	Sterilization of a Fermentation Medium	188
Exa 24.9	Use of Combined Material and Energy Balances to Solve a Distillation Problem	189
Exa 25.1	Determination of a Heat of Formation from Heat Transfer Measurements	192
Exa 25.2	Retrieval of Heats of Formation from Reference Data .	193
Exa 25.3	Calculation of the Standard Heat of Reaction from the Standard Heats of Formation	194

Exa 25.4	Calculation of the Heat of Reaction at a Temperature different from the Standard Conditions	195
Exa 25.5	Calculation of the Heat Transfer using Heat of Reaction in a Process in which Reactants enter and Products leave at Different Temperatures	196
Exa 25.6	Calculation of the Enthalpy Change in an Anerobic Culture	199
Exa 25.7	Green Chemistry Examining Alternate Processes . . .	200
Exa 25.8	Calculation of the Heat of Reaction at a Temperature different from the Standard Conditions with Heat of Formation Merged with Sensible Heat	202
Exa 25.9	Calculation of the Heat Transfer when Reactants enter and Products leave at Different Temperatures	203
Exa 25.10	Heating Value of Coal	205
Exa 25.11	Selecting a Fuel to reduce SO ₂ emissions	206
Exa 26.1	Analysis of the Degrees of Freedom for a Combustion Process	208
Exa 26.2	Calculation of Flame Temperature	210
Exa 26.3	Application of general Energy Balance in a Process in which More than one Reaction Occurs	213
Exa 26.4	Application of general Energy Balance in a Process Composed of Multiple Units	215
Exa 26.5	Production of Citric Acid by Fungus	218
Exa 27.1	Calculation of the Work done during Evaporation of a Liquid	221
Exa 27.2	Calculation of Work in a Batch Process	222
Exa 27.3	Efficiency of Power Generation by a Hydroelectric Plant	223
Exa 27.4	Calculation of Plant Efficiency	224
Exa 27.5	Comparison of the Reversible Work for a Batch Process with that of a Flow Operating under Same Conditions	225
Exa 27.6	Application of the Mechanical Energy Balance to the Pumping of Water	226
Exa 28.1	Application of Heats of Solution data	228
Exa 28.2	Application of Heat of Solution Data	230
Exa 28.3	Application of an Enthalpy Concentration Chart . . .	231
Exa 29.1	Determining Properties of Moist air from Humidity Chart	234
Exa 29.2	Heating at constant Humidity	235
Exa 29.3	Cooling and Humidification using a Water Spray . . .	236

Exa 29.4	Combined Material and Energy Balance for a Cooling Tower	237
Exa 29.5	Drying of Chlorella	239

Chapter 1

Dimensions units and their conversion

Scilab code Exa 1.1 Dimensions and Units

```
1 clear;
2 clc;
3
4 // Example 1.1
5 printf('Example 1.1\n\n');
6 //Page no. 13
7 // Solution
8
9 //(a)
10 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');
11
12 //(b)
13 // Converting all terms to same unit
14 hp = 746;//[watts]
15 total = 1*hp+300;//[watts]
16
```

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

Scilab code Exa 1.2 Conversion of Units

```
1 clear ;
2 clc;
3
4 // Example 1.2
5 printf('Example 1.2\n\n');
6 //Page no. 17
7 // Solution
8
9 // (a)
10 // Converting all terms to same unit
11 ml = 1.61;//[km]
12 km = (2*1)/(ml);//[miles]
13 printf('(a) 2 kilometers is equal to %.2f miles.\n',
    km);
14
15 //(b)
16 in = 2.54;//[cm]
17 dy = 24*60;//[min]
18 nw_unit = (400*(in)^3*1)/(1*dy);//[cubic centimetre/
    min]
19 printf(' (b) 400 cubic in./day is equal to %.2f
    cubic centimetre/min.',nw_unit);
```

Scilab code Exa 1.3 Nanotechnology

```
1 clear;
2 clc;
3
4 // Example 1.3
```

```

5 printf('Example 1.3\n\n');
6 //Page no. 17
7 // Solution
8
9 //(a)
10 // Converting all terms to same unit
11 nm = 10^(-9);//[meters]
12 m1 = 10;//[decimeters]
13 dm = (1.8*nm*m1)/(1*1);//[decimeters]
14 printf('(a) 1.8 nanometers is equal to %.2e dm.\n',
        dm);
15
16 //(b)
17 m2 = 39.37;//[inches]
18 in = (1.8*nm*m2)/(1*1);//[inches]
19 printf('(b) 1.8 nanometers is equal to %.2e in.\n',
        in);

```

Scilab code Exa 1.4 A conversion involving lbm and lbf

```

1 clear ;
2 clc;
3
4 // Example 1.4
5 printf('Example 1.4\n\n');
6 //Page no. 19
7 // Solution
8
9 // Potential Energy = mgh
10 // Assume 100 lb means 100 lb mass
11 m = 100;//[lb]
12 g = 32.2 ;//[ft/second square]
13 h = 10 ;//[ft]
14 gc = 32.174 ;//[ (ft*lbm)/(second square/lbf)]
15 pe = (m*(g/gc)*h) ;//[ft*lbf]

```

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

Scilab code Exa 1.5 Conversion associated with Biological Materials

```
1 clear;
2 clc;
3
4 // Example 1.5
5 printf('Example 1.5\n\n');
6 //Page no.20
7 // Solution
8
9 // Basis 1 min
10 // Assume 100lb means 100 lb mass
11 g = 10^6 ;//[ug mol]
12 lb = 454 ;//[g mol]
13 ml = .001 ;//[L]
14 L = 3.531*10^(-2);//[ft ^3]
15 hr = 60 ;//[min]
16 dy = 24 ;//[hr]
17 pr_rate = (0.6*1*1*1*hr*dy/(g*lb*ml*L)); //[ft*lbf]
18
19 printf('Production rate of glucose is %.4f lb mol/(
    cubic feet*day).\n',pr_rate);
```

Scilab code Exa 1.6 Dimensional Consistency

```
1 clear;
2 clc;
3
4 // Example 1.6
5 printf('Example 1.6\n\n');
```



```

6 //Page no. 22
7 // Solution
8
9 // using suitable conversion factors inside front
  cover of book
10 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');
11 printf(' The exponential must be dimensionless so
  that 0.021 must have the associated units of s
  (-1).\n');
12
13 m = 39.27 ;//[inches]
14 um = 10(-6) ;//[meters]
15 c1 = 16.2*m*um ;//[inches]
16 mn = 60 ;//[seconds]
17 c2 = 0.021*60;//[min(-1)]
18 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

```

1 clear;
2 clc;
3
4 // Example 1.7
5 printf('Example 1.7\n\n');
6 // Page no. 23
7 // Solution
8
9 printf('By analysing dimensionally , both a and x
  have same units(from 1+(x2/a2)),thus left hand
  side of equation has units of 1/x(from d/dx),\n

```

```

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

```

Scilab code Exa 1.8 Retention of significant Figures

```

1 clear ;
2 clc;
3
4 // Example 1.8
5 printf('Example 1.8\n\n');
6 // Page no. 28
7 // Solution
8
9 // Using Scientific notation
10 x = 2.24 * 10^4 ; //[kg]
11 y = 2.01 * 10^4 ;//[kg]
12 D = x - y ;// Difference obtained by using
    scientific notation //[kg]
13
14 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

```

1 clear;
2 clc;
3
4 // Example 1.9
5 printf('Example 1.9\n\n');
6 //Page no. 29

```

```
7 // Solution
8
9 um = 3 ;//[kb]
10 kb = 1000 ;//[bp]
11 bs_prs = (3*um*kb)/(1*1);
12 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

```
1 clear ;
2 clc;
3
4 // Example 2.1
5 printf('Example 2.1\n\n');
6 //Page no. 45
7 // Solution
8
9 // Count the number of each element from fig. E2.1.
10 // Look for the atomic weights of elements from
    Appendix B
11 // Assume the one cell is a molecule
12 n_Ba = 2 ;// Number of atoms of Ba
13 n_Cu = 16 ;// Number of atoms of Cu
14 n_O = 24 ;// Number of atoms of O
15 n_Y = 1 ;// Number of atoms of Y
16 m_Ba = 137.34 ;//Atomic wt. -[g]
17 m_Cu = 63.546 ;//Atomic wt. -[g]
18 m_O = 16.00 ;//Atomic wt. -[g]
```

```

19 m_Y = 88.905; //Atomic wt.-[g]
20 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]
21
22 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

```

1 clear ;
2 clc;
3
4 // Example 2.2
5 printf('Example 2.2\n\n');
6 // Page no. 46
7 // Solution
8
9 //(a)
10 m_NaOH = 40.0 ;//[lb]
11 pnd_mol = 2*1/m_NaOH ;//[lb mol]
12 printf('(a) The number of pound moles of NaOH in
    2.00 lb NaOH is %.2f lb mol.\n',pnd_mol);
13
14 //(b)
15 grm_mol = pnd_mol*454 ;//[g mol]
16 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

```

1 clear;
2 clc;
3

```

```

4 // Example 2.3
5 printf('Example 2.3\n\n');
6 //Page no. 46
7 // Solution
8
9 //Basis 7.5 g mol of NaOH
10 m_NaOH = 40.0 ;//[lb]
11 lb = 454 ;//[g mol]
12 n = 7.50*1*m_NaOH/(lb*1);
13 printf('Number of pounds of NaOH is %.3f lb.\n',n);

```

Scilab code Exa 2.4 Calculation of Density

```

1 clear ;
2 clc;
3
4 // Example 2.4
5 printf('Example 2.4\n\n');
6 //Page no. 53
7 // Solution
8
9 // (a)
10 d_w1=1000 ;//[kg/cubic metre]
11
12 d1=(1.184*d_w1*1000)/(10^6) ;//[g/cubic centimetre]
13 printf('(a) Density in g/cubic centimetre is %.4f g/
    cubic centimetre.\n',d1);
14
15 // (b)
16 d_w2= 62.4 ;//[lbm/cubic feet]
17
18 d2=1.184*d_w2/1 ;//[lbm/cubic feet]
19
20 printf(' (b) Density in lbm/cubic feet is %.1f lbm/
    cubic feet.\n',d2);

```

```

21
22 // (c)
23 d3=1.184*d_w1 ;//[kg/cubic metre]
24
25 printf(' (c) Density in kg/cubic metre is %.1f kg/
      cubic metre.\n',d3);

```

Scilab code Exa 2.5 Calculation of Mass and Moles

```

1 clear ;
2 clc;
3
4 // Example 2.5
5 printf('Example 2.5\n\n');
6 //Page no. 54
7 // Solution
8
9 m_wt=192 ;//[kg]
10 d_sol=1.024*1000 ;//[kg/cubic metre]
11 // 1000L=1 cubic metre
12 c_sol=d_sol/1000 ;//[kg/L]
13 c_drug=c_sol*.412 ;//[kg/L]
14 printf('Concentration of drug in solution is %.3f kg
      /L .\n',c_drug);
15
16 Q=10.5 ;//[L/min]
17 Qmol=10.5*c_drug/m_wt ;//[kg mol/min]
18 printf(' Flow rate of drug is %.3f kg mol/min. \n',
      Qmol);

```

Scilab code Exa 2.6 Conversion between Mass Fraction and Mole Fraction

```

1 clear ;

```

```

2  clc;
3
4  // Example 2.6
5  printf('Example 2.6\n\n');
6  //Page no.57
7  // Solution
8
9  // Let component 1 be water and component 2 be NaOH
10 // Basis 10 kg total solution
11 m1 = 5.0 ;//[kg]
12 m2 = 5.0; //[kg]
13 total = m1 + m2 ;//[kg]
14 m_fr1 = m1/total ;//mass fraction of water
15 m_fr2 = m2/total ;//mass fraction of NaOH
16 mw1 = 18.0 ;//molecular weight of water
17 mw2 = 40.0 ;//molecular weight of NaOH
18 mol1 = m1/mw1;
19 mol2 = m2/mw2;
20 mol_fr1 = mol1/(mol1 + mol2) ;//mol fraction of
    water
21 mol_fr2 = mol2/(mol1 + mol2) ;//mol fraction of
    NaOH
22 printf(' Component      kg      Mass fraction      Mol
    .Wt.      kg mol      Mole fraction\n');
23 printf('n Water          %.2f      %.3f              %.1
    f          %.3f      %.2f\n',m1,m_fr1,mw1,mol1,
    mol_fr1);
24 printf(' NaOH          %.2f      %.3f              %.1f
    %.3f      %.2f\n',m2,m_fr2,mw2,mol2,
    mol_fr2);
25 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


```

1 clear ;
2 clc;
3
4 // Example 2.7
5 printf('Example 2.7\n\n');
6 //Page no.58
7 // Solution
8
9 // Basis 500 L solution containing 35g/L
10 // (NH4)2SO4 is the only nitrogen source
11 cn = 35 ;//[g/L]
12 wt = 9 ;//[wt % N]
13 m_wt1 = 132 ;//[g]
14 m_wt2 = 14 ;//[g]
15 amt = (500*(35)*.09*1*1*m_wt1)/(1*m_wt2*1*1);
16 printf('Total amount of (NH4)2SO4 consumed is %.1f
      g. ',amt);

```

Scilab code Exa 2.8 Use of ppm

```

1 clear ;
2 clc;
3
4 // Example 2.8
5 printf('Example 2.8\n\n');
6 //Page no. 63
7 // Solution
8
9 // 1 kg of the air/HCN mixture
10 // (a)
11 m1 = 27.03 ;//[g]
12 m2 = 29.0 ;//[g]
13 cn = (10*m1*1000*1000)/(10^6*m2) ;//[mgHCN/kg air]
14 printf('(a) 10.0 ppm HCN is %.2f mg HCN/kg air.\n',
      cn);

```

```

15
16 // (b)
17 ld = 300 ;//[mg/kg air]
18 fr = cn/ld;
19 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

```

1 clear ;
2 clc;
3
4 // Example 2.9
5 printf('Example 2.9\n\n');
6 //Page no. 64
7 // Solution
8
9 // Let component 1 be water and component 2 be HNO3
10 // Basis 1L solution
11 c = 15 ;//[g/L]
12 sg = 1.10 ;
13 L = 1000 ;//[cubic centimetre]
14 m1 = 18.0 ;//[g]
15 m2 = 63.02 ;//[g]
16 cn2 = (15*1)/(L*sg) ;//[gHNO3/g soln]
17 // Basis 1g soln
18 cn1 = 1-cn2 ;// Mass of water in 1 g soln
19 mg1 = cn1/m1;
20 mg2 = cn2/m2;
21 ml_fr1 = mg1/(mg1+mg2);
22 ml_fr2 = mg2/(mg1+mg2);
23 printf(' (a) Component      g(per 1g soln)
        Mol.Wt.      g mol      Mole fraction\n'
        )

```

```

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

```

Scilab code Exa 2.10 Evaluation of Alternate Processes for the Production of Methyl Methacrylate

```

1 clear;
2 clc;
3
4 // Example 2.10
5 // Page no. 64
6 // Solution
7
8 // Given
9 // Process a
10 // 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
11 Lb1 = [0.68,0.32,0.37,1.63,1] ;// Mass of
      compounds -[lb]
12 Value1 = [0.43, 0.67,0.064,0.04,0.78] ;// Cost of
      compounds -[$/lb]
13 TLV1 = [750,10,200,2,100] ;// TLV value of
      compounds -[ppm]
14 OITF1 = [0,1000,10,10000,10] ;// Note : (?) mark
      values are neglected as they are nearly equal to
      zero

```

```

15
16 // Process b
17 // Let us take array of given values for compounds
    in following order 1- Isobutylene , 2 - Methanol ,
    3- Pentane , 4-Sulphuric acid , 5 - Methyl
    methacrylate
18 Lb2 = [1.12,0.38,0.03,0.01,1.00] ;// Mass of
    compounds -[lb]
19 Value2 = [0.31,0.64,0.112,0.04,0.78] ;// Cost of
    compounds -[$/lb]
20 TLV2 = [200,200,600,2,100] ;// TLV value of
    compounds -[ppm]
21 OITF2 = [0,10,0,10000,10] ;// Note : (?) mark
    values are neglected as they are nearly equal to
    zero
22
23 NetV1 = Lb1(5)*Value1(5) - Lb1(2)*Value1(2) - Lb1
    (3)*Value1(3) - Lb1(4)*Value1(4) - Lb1(1)*Value1
    (1); // Net Value for process (a) -[$]
24 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) -[$]
25
26 printf('1.With respect to cost criteria\n');
27 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);
28
29 // With respect to two environmental criteria
30 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
31 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
32
33 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
34 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
35
36 printf('\n 2.With respect to two environmental
    criteria\n');
37 printf('    Process (a)\n    TLV index for process a
    is %.2f .\n    OITF index process a is %.2f . \n'
    ,TLV_index1,OITF_index1);
38 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

```
1 clear ;
2 clc;
3
4 // Example 3.1
5 printf('Example 3.1\n\n');
6 //Page no. 79
7 // Solution
8
9 // Let component 1 be Ce and component 2 be O
10 // Basis 2kg mol CeO
11 mol1 = 1.0 ;//[kg mol]
12 mol2 = 1.0 ;//[kg mol]
13 total = mol1+mol2 ;//[kg mol]
14 mol_fr1 = mol1/total ;//mole fraction of Ce
15 mol_fr2 = mol2/total ;//mole fraction of O
16 mw1 = 140.12; //molecular weight of Ce
17 mw2 = 16.0 ;//molecular weight of O
18 m1 = mw1*mol1;
19 m2 = mw2*mol2;
20 m_fr1 = m1/(m1+m2) ;//mass fraction of Ce
21 m_fr2 = m2/(m1+m2) ;//mass fraction of O
```

```

22
23 printf('Component      kg mol      Mole fraction
      Mol.Wt.      kg.      Mass fraction\n')
24 printf('\n Ce          %.2 f      %.3 f
      %.2 f      %.3 f      %.2 f\n',mol1,
      mol_fr1,mw1,m1,m_fr1);
25 printf(' O          %.2 f      %.3 f
      %.2 f      %.3 f      %.2 f\n',mol2,mol_fr2,mw2,m2
      ,m_fr2);
26 printf(' Total          %.2 f      %.3 f
      %.2 f      %.3 f      %.2 f',mol1+mol2,mol_fr1+
      mol_fr2,mw1+mw2,m1+m2,m_fr1+m_fr2);

```

Scilab code Exa 3.2 Choosing a Basis

```

1 clear ;
2 clc;
3
4 // Example 3.2
5 printf('Example 3.2\n\n');
6 //Page no. 80
7 // Solution
8
9 // Basis 100kg mol gas
10 m11 = 20.0 ;//[kg mol]
11 m12 = 30.0 ;//[kg mol]
12 m13 = 40.0 ;//[kg mol]
13 m14 = 10.0 ;//[kg mol]
14 mw1 = 44.0 ;//molecular weight of CO2
15 mw2 = 28.0 ;//molecular weight of CO
16 mw3 = 16.04 ; //molecular weight of CH4
17 mw4 = 2.02 ;//molecular weight of H2
18 m1 = mw1*m11;
19 m2 = mw2*m12;
20 m3 = mw3*m13;

```

```

21 m4 = mw4*m14;
22 printf(' Component      kg mol      Mol.Wt.      kg.
        \n')
23 printf(' CO2            %.2 f      %.2 f      %.0 f
        \n',m11,mw1,m1);
24 printf(' CO             %.2 f      %.2 f      %.0 f
        \n',m12,mw2,m2);
25 printf(' CH4            %.2 f      %.2 f      %.0 f
        \n',m13,mw3,m3);
26 printf(' H2             %.2 f      %.2 f      %.0 f
        \n',m14,mw4,m4);
27 printf('\n Total      %.2 f      %.2 f      %.0 f
        \n',m11+m12+m13+m14,mw1+mw2+mw3+mw4,m1+m2+
        m3+m4);
28 av_m = (m1+m2+m3+m4)/100 ;//[kg]
29 printf('\nAverage molecular mass of gas is %.1 f kg.\n
        n',av_m);

```

Scilab code Exa 3.3 Choosing a Basis for Cell Growth

```

1 clear ;
2 clc;
3
4 // Example 3.3
5 printf('Example 3.3\n\n');
6 //Page no. 81
7 // Solution
8
9 // Basis 1 hour
10 rc = 5000 ;//[cpm-counts per minute]
11 cg = 10000/24 ;//[cells/hr]
12 k = cg/rc ;//[cells/cpm]
13 n_rc = 8000 ;//[cpm]
14 n_cg = k*n_rc ;//[cells/hr]
15 printf('New average cell growth rate is %.0 f cells/

```



```

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

```

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

```

1 clear;
2 clc;
3
4 // Example 3.4
5 printf('Example 3.4\n\n');
6 //Page no. 82
7 // Solution
8
9 // Basis 100 g mol of Nd(4.5)Fe(77)B(18.5)
10 //(a)
11 n_Fe = 77-0.2;
12 printf('(a) Molecular formula after adding Cu is Nd
    (4.5)Fe(%.1f)B(18.5)Cu(.2).\n',n_Fe);
13
14 //(b)
15 o_m11 = 4.5 ;//[kg mol]
16 o_m12 = 77.0 ;//[kg mol]
17 o_m13 = 18.5 ;//[kg mol]
18 o_m14 = 0.0 ;//[kg mol]
19 f_m11 = 4.5 ;//[kg mol]
20 f_m12 = 77.0-0.2 ;//[kg mol]
21 f_m13 = 18.5 ;//[kg mol]
22 f_m14 = 0.2 ;//[kg mol]
23 mw1 = 144.24 ;//molecular weight of Nd
24 mw2 = 55.85 ;//molecular weight of Fe
25 mw3 = 10.81 ; //molecular weight of B
26 mw4 = 63.55 ;//molecular weight of Cu

```

```

27 m1 = mw1*f_m11;
28 m2 = mw2*f_m12;
29 m3 = mw3*f_m13;
30 m4 = mw4*f_m14;
31 f1 = f_m11/100;
32 f2 = f_m12/100;
33 f3 = f_m13/100;
34 f4 = f_m14/100;
35 tf = f1+f2+f3+f4;
36 printf('\n (b) Component      Original g mol      Final
      g mol      Mol.Wt.      g.      Mass
      fraction\n')
37 printf('      Nd      %.2 f      %.2 f
      %.2 f      %.2 f      %.3 f\n',
      o_m11, f_m11, mw1, m1, f1);
38 printf('      Fe      %.2 f      %.2 f      %.2 f
      %.2 f      %.2 f      %.3 f\n',
      o_m12, f_m12, mw2, m2, f2);
39 printf('      B      %.2 f      %.2 f      %.2 f
      %.2 f      %.2 f      %.3 f\n',
      o_m13, f_m13, mw3, m3, f3);
40 printf('      Cu      %.2 f      %.2 f      %.2 f
      %.2 f      %.2 f      %.3 f\n',
      o_m14, f_m14, mw4, m4, f4);
41 printf('\n      Total      100.0      100.0
      %.2 f      %.3 f\n',
      m1+m2+m3+m4, tf);

```

Scilab code Exa 3.5 Changing Bases

```

1 clear ;
2 clc;
3
4 // Example 3.5
5 printf('Example 3.5\n\n');

```

```

6 //Page no. 84
7 // Solution
8
9 // Basis 100 kg coal
10 m1_r = 9;
11 wt_r = (9*1.008)/(1*12) ;//conversion of mole ratio
    to wt.ratio
12 m1 = 2 ;//[kg] wt.of sulphur
13 m2 = 1 ;//[kg] wt. of nitrogen
14 m3 = 6 ;//[kg] wt. of oxygen
15 m4 = 11 ;//[kg] wt. of ash
16 m5 = 3 ;//[kg] wt. of water
17 m6 = (1*77)/(wt_r+1) ;//[kg] wt. of carbon
18 m7 = wt_r*m6 ;//[kg] wt. of hydrogen
19 wc = 100-(m4+m5) ;//[kg] wt. of coal excluding ash
    and water
20 wf1 = m1/wc;
21 wf2 = m2/wc;
22 wf3 = m3/wc;
23 wf4 = m4/wc;
24 wf6 = m6/wc;
25 wf7 = m7/wc;
26 tf = wf1+wf2+wf3+wf6+wf7;
27 printf(' Component          kg.           Mass
    fraction ');
28 printf('\n C                %.2 f          %.2 f\n',m6
    ,wf6);
29 printf(' H                %.2 f          %.2 f\n',m7,
    wf7);
30 printf(' S                %.2 f          %.2 f\n',m1,
    wf1);
31 printf(' N                %.2 f          %.2 f\n',m2,
    wf2);
32 printf(' O                %.2 f          %.2 f\n',m3,
    wf3);
33 printf('\n Total          %.2 f          %.2 f\n',wc
    ,tf);

```

Chapter 4

Temperature

Scilab code Exa 4.1 Temperature Conversion

```
1 clear ;
2 clc ;
3
4 // Example 4.1
5 printf('Example 4.1\n\n');
6 //Page no. 92
7 // Solution
8
9 //(a)
10 Temp_c=100 ;//[degree Celsius]
11 Temp_k=Temp_c+273 ;//[K]
12 printf('(a) Temperature in kelvin is %.2f K\n',
        Temp_k);
13
14 //(b)
15 Temp_f=(100*(1.8/1)) +32 ;//[degree Fahrenheit]
16 printf(' (b) Temperature in degree Fahrenheit is %.2
        f \n',Temp_f);
17
18 //(c)
19 Temp_r= Temp_f + 460 ;//[degree Rankine ]
```

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

Scilab code Exa 4.2 Temperature Conversion

```
1 clear ;  
2 clc;  
3  
4 // Example 4.2  
5 printf('Example 4.2\n\n');  
6 // Page no. 93  
7 // Solution  
8  
9 // Given  
10 // Heat capacity =  $139.1 + (1.56 \times 10^{-1})T_c$  J/(g mol*  
    degree C), T is in degree C  
11 // First convert Tc (Temperature in degree celsius)  
    to TR (in degree R) to get c + dTR, where  
12 c =  $139.1 + (1.56 \times 10^{-1}) * (-460 - 32) / 1.8$  ;  
13 d =  $(1.56 \times 10^{-1}) / 1.8$ ;  
14  
15 //Now convert c +dTR to (Btu/lb mol*degree R) to get  
    answer of form a + bTR, where  
16 a =  $c * (454 / (1055 * 1.8))$  ;  
17 b =  $d * (454 / (1055 * 1.8))$  ;  
18  
19 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

```
1 clear ;
2 clc;
3 // Example 5.1
4 printf('Example 5.1\n\n');
5 //Page no.109
6 // Solution
7
8 P = 60 ;//[Gpa]
9
10 //(a)
11 p_atm = (P*(10^6))/101.3 ;//[atm]
12 printf('(a) Pressure in atmospheres is %.2e atm\n',
    p_atm);
13
14 //(b)
15 p_s = (P*(10^6)*14.696)/101.3 ;//[psia]
16 printf(' (b) Pressure in psia is %.2e psia\n',p_s);
17
18 // (c)
19 p_in = (P*(10^6)*29.92)/101.3 ;//[inches of Hg]
20 printf(' (c) Pressure in inches of Hg is %.2e in. Hg
```

```

    \n',p_in);
21
22 // (d)
23 p_mm = (P*(10^6)*760)/101.3 ;//[mm of Hg]
24 printf(' (d) Pressure in mm of Hg is %.2e mm Hg\n',
    p_mm);

```

Scilab code Exa 5.2 Pressure Conversion

```

1 clear;
2 clc;
3 // Example 5.2
4 printf('Example 5.2\n\n');
5 //Page no. 110
6 // Solution
7
8 b_rd = 28.0 ;//[in. Hg]
9 p_rd = 51.0 ;//[psia]
10 p_atm = b_rd*14.7/29.92 ;//[psia]
11 p_tnk = p_atm+p_rd ;//[psia]
12 printf(' Pressure in tank in psia is %.1f psia\n',
    p_tnk);

```

Scilab code Exa 5.3 Vacuum Pressure Reading

```

1 clear ;
2 clc;
3 // Example 5.3
4 printf('Example 5.3\n\n');
5 //Page no. 111
6 // Solution
7
8 b_rd = 100.0 ;//[kPa]

```

```

9 gp = 64.5*101.3/76.0 ;//[kPa]
10 p_tnk = b_rd-gp ;//[kPa]
11 printf(' Absolute Pressure in tank in is %.1f kPa\n'
        ,p_tnk);
12 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

```

1 clear ;
2 clc;
3 // Example 5.4
4 printf('Example 5.4\n\n');
5 //Page no. 115
6 // Solution
7
8 df = 1.10*10^3 ;//[kg/m^3]
9 d = 1.0*10^3 ;//[kg/m^3]
10 g = 9.8 ;//[m/s^2]
11 h = 22.0 ;//[mm]
12 dP = (df-d)*g*(h*10^(-3)) ;//[Pa]
13 printf('Pressure difference across the orifice plate
        is %.1f Pa.\n',dP);

```

Scilab code Exa 5.5 Pressure Conversion

```

1 clear ;
2 clc;
3 // Example 5.5
4 printf('Example 5.5\n\n');
5 //Page no. 117
6 // Solution

```



```
7
8 p_atm=730.0*29.92/760.0 ;//[in. Hg]
9 gp= (4.0*29.92)/(2.54*12*33.91) ;//[in. Hg]
10 p_air=p_atm-gp ;//[in. Hg]
11 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

```
1 clear ;
2 clc;
3 // Example 6.1
4 printf('Example 6.1\n\n');
5 // Page no. 142
6 // Solution
7
8 // Given
9 P_0 = 89 ;// Premium octane -[octane/gal]
10 S_0 = 93 ;// Supreme octane - [octane/gal]
11 R_0 = 87 ;// Regular octane - [octane/gal]
12 CP = 1.269 ;// Cost of premium octane -[$/gal]
13 SP = 1.349 ;// Cost of supreme octane -[$/gal]
14 RP = 1.149 ;// Cost of regular octane -[$/gal]
15
16 // Let x and y fraction of regular octane and
    supreme octane is blended respectively , therefore:
    x + y = 1 ... (a)
17 // and 89 = 87x + 93y ... (b)
```

```

18 // Solve equations (a) and (b) simultaneously
19 a = [1 1;87 93] ;// Matrix of coefficients of
    unknown
20 b = [1;89] ;// Matrix of constant
21 c = a\b ;// Matrix of solutions- x = c(1) , y = c(2)
22 cost = c(1)*RP + c(2)*SP ;// Cost after blending -
    [$/gal]
23 sv = CP - cost ;// Save on blending - [$/gal]
24
25 // Check whether there is loss or save
26 if (sv<0)
27     then
28         printf('We will not save money by blending. ');
29
30     else
31         printf('We will save money by blending , and
                save is %.3f $/gal. ',sv);

```

Scilab code Exa 6.2 Concentration of cells using a Centrifuge

```

1 clear ;
2 clc;
3 // Example 6.2
4 printf('Example 6.2\n\n');
5 //Page no. 147
6 // Solution
7
8 // Basis 1 hour
9 fd= 1000.0 ;//feed rate-[L/hr]
10 cfd= 500.0; //Weight of cells per litre- [mg/L]
11 dn= 1.0 ;//Density of feed-[g/cm^3]
12 wp= 50.0 ;// Weight percent of cells in product
    stream
13 Pg=(fd*cfd*dn)/(1000*wp*.01) ;// Mass balance for
    cells

```

```

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

```

Scilab code Exa 6.3 Discharge of Tank Residuals to the Environment

```

1 clear ;
2 clc;
3 // Example 6.3
4 printf('Example 6.3\n\n');
5 //Page no. 154
6 // Solution
7
8 //Basis 10000 gal motor oil at an assumed 77 degree
    fahrenheit
9 dn = 0.80 ;//Density of motor oil -[g/cm^3]
10 in_ms = (10000*(0.1337)*62.4*dn) ;// Initial mass of
    motor oil in the tank -[lb]
11 printf(' Initial mass of motor oil in the tank is %
    .1f lb\n',in_ms);
12 m_fr = .0015 ;//Mass fractional loss
13 printf(' Mass fractional loss is %.4f \n',m_fr);
14 Dsg = m_fr*in_ms ;// Mass balance for the fluid
15 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

```
1 clear ;
2 clc;
3 // Example 7.1
4 printf('Example 7.1\n\n');
5 //Page no.169
6 // Solution
7
8 v_ts = 105.0 ;// velocity of train wrt station -[cm/s
   ]
9 v_mt = 30.0 ;// velocity of man wrt train -[cm/s]
10 v_hm = 2.0 ;// velocity of hot dough wrt man-[cm/s]
11 v_am = 1.0 ;// velocity of ant wrt man- [cm/s]
12 // 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
13 v_as = v_ts + v_mt + v_am;
14 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

```
1 clear ;
2 clc;
3 // Example 7.2
4 printf('Example 7.2\n\n');
5 // Page no. 169
6 // Solution Fig. E7.2
7
8 printf("Drawing as in fig E7.2 is not possible with
   scilab.")
```

Scilab code Exa 7.3 Placing the unknown Information on the Diagram

```
1 clear ;
2 clc;
3 // Example 7.3
4 printf('Example 7.3\n\n');
5 // Page no. 171
6 // Solution Fig. E7.3
7
8 printf("Drawing as in fig E7.3 is not possible with
   scilab.")
```

Scilab code Exa 7.4 Analysis of the Degrees of Freedom

```
1 clear ;
2 clc;
3 // Example 7.4
```

```

4 printf('Example 7.4\n\n');
5 //Page no. 180
6 // Solution
7
8 n_un= 7 ;// Number of unknowns in the given problem–
      3 values of xi and 4 values Fi
9 n_ie = 5 ;// Number of independent equations
10 // Summary of independent equations
11 // Three material balances:CH4,C2H6 and N2
12 // One specified ratio: moles of CH4 to C2H6 equal
      1.5
13 // One summation of mole fraction in mixture equals
      1
14 d_o_f = n_un-n_ie ;// No. of degree of freedom
15
16 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

```

1 clear ;
2 clc;
3 // Example 7.5
4 printf('Example 7.5\n\n');
5 //Page no. 182
6 // Solution
7
8 n_un=8 ;// Number of unknowns in the given problem–
      8 values of mole fractions
9 n_ie =6 ;// Number of independent equations– six
      elemental balances
10 d_o_f= n_un-n_ie ;// Number of degree of freedom
11 printf('Number of degree of freedom for the given
      system is %i .\n',d_o_f);

```

- 12 //Note: Experiments show that the change in $CH_{1.8}O_{0.5}N_{0.16}S_{0.0045}P_{0.0055}$ 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

```
1 clear ;
2 clc;
3 // Example 8.1
4 printf('Example 8.1\n\n');
5 //Page no. 197
6 // Solution
7
8 // Basis : 1 min
9 d_w = 1.0 ;// Density of aqueous solution -[g/cubic
    metre]
10 d_sol = 0.6 ;// Density of organic solvent -[g/cubic
    metre]
11
12 n_un = 8 ;// Number of unknowns in the given problem
13 n_ie = 8 ;// Number of independent equations
14 d_o_f = n_un-n_ie ;// Number of degree of freedom
15 printf('Number of degree of freedom for the given
```

```

    system is  %i .\n',d_o_f);
16
17 // Material balance of Strep.
18 x = (200*10+10*0-200*0.2)/10;//[g]
19 printf('Strep per litre of solvent is  %.1f g .\n',x
    );
20
21 cnc = x/(1000*d_sol) ;//[g Strep/g of S]
22 printf('Strep per gram of solvent is  %.4f g Strep/g
    of S .\n',cnc);
23
24 m_fr = cnc/(1+cnc) ;//Mass fraction
25 printf('Mass fraction of Strep is  %.3f g .\n',m_fr)
    ;

```

Scilab code Exa 8.2 Separation of gases Using a Membrane

```

1 clear ;
2 clc;
3 // Example 8.2
4 printf('Example 8.2\n\n');
5 // Page no. 199
6 // Solution Fig. E8.2b
7
8 F_O2 = 0.21 ;// fraction of O2 in feed(F)
9 F_N2 = 0.79 ;// fraction of N2 in feed(F)
10 P_O2 = 0.25 ;// fraction of O2 in product(P)
11 P_N2 = 0.75 ;// fraction of N2 in product(P)
12 F = 100 ;// Feed - [g mol]
13 w = 0.80 ;// Fraction of waste
14 W = w*F ;// Waste -[g mol]
15
16 // By analysis for degree of freedom , DOF comes to
    be zero
17 P = F - W ;// By overall balance - [g mol]

```

```

18 W_O2 = (F_O2*F - P*P_O2)/100 ;// Fraction of O2 in
    waste stream by O2 balance
19 W_N2 = (W - W_O2*100)/100 ;//Fraction of N2 in waste
    stream
20
21 printf('Composition of Waste Stream\n' );
22 printf('\n Component           Fraction in waste
    stream\n' );
23 printf(' O2                       %.2 f\n',W_O2 );
24 printf(' N2                       %.2 f\n',W_N2 );

```

Scilab code Exa 8.3 Overall analysis for a continuous Distillation Column

```

1 clear;
2 clc;
3 // Example 8.3
4 printf('Example 8.3\n\n');
5 // Page no. 202
6 // Solution
7
8 // Basis : 1 hr so F = 1000 kg
9 F = 1000 ;// feed rate -[kg/hr]
10 P = F/10 ;// product mass flow rate -[kg/hr]
11
12 n_un = 9 ;// Number of unknowns in the given problem
13 n_ie = 9 ;// Number of independent equations
14 d_o_f = n_un-n_ie ;// Number of degree of freedom
15 printf('Number of degree of freedom for the given
    system is %i .\n',d_o_f);
16
17 // Overall mass balance: F = P+B
18 B = F-P ;// bottom mass flow rate -[kg/hr]
19 printf('\n Bottom mass flow rate -           %.1 f
    kg \n',B);
20

```

```

21 // Composition of bottoms by material balances
22 m_EtOH = 0.1*F-0.6*P ;// By EtOH balance -[kg]
23 m_H2O = 0.9*F - 0.4*P ;// By H2O balance -[kg]
24 total = m_EtOH+m_H2O ;//[kg]
25 f_EtOH = m_EtOH/total ;// Mass fraction of EtOH
26 f_H2O = m_H2O/total ;// Mass fraction of H2O
27
28 printf(' Mass of EtOH in bottom -           %.1 f
      kg \n',m_EtOH);
29 printf(' Mass of H2O in bottom -           %.1 f
      kg \n',m_H2O);
30 printf(' Mass fraction of EtOH in bottom -   %.3 f \
      n',f_EtOH);
31 printf(' Mass fraction of H2O in bottom -   %.3 f \
      n',f_H2O);

```

Scilab code Exa 8.4 Mixing of Battery Acid

```

1 clear ;
2 clc;
3 // Example 8.4
4 printf('Example 8.4\n\n');
5 // Page no. 205
6 // Solution Fig E8.4
7
8 // Given
9 A = 200 ;// Mass of added solution [kg]
10 P_H2SO4 = .1863 ;//Fraction of H2SO4 in P(Final
      solution)
11 P_H2O = .8137 ;//Fraction of H2O in P(Final solution
      )
12 A_H2SO4 = .777 ;//Fraction of H2SO4 in A(Added
      solution)
13 A_H2O = .223 ;//Fraction of H2O in A(Added solution)
14 F_H2SO4 = .1243 ;//Fraction of H2SO4 in F(Original

```

```

        solution)
15 F_H2O = .8757 ;//Fraction of H2O in F(Original
        solution)
16
17 // By analysis for degree of freedom , DOF comes to
        be zero
18 // Solve following equations simultaneously for F
        and P,
19 // P*P_H2O - F*F_H2O = A*A_H2O - By H2O balance
20 // P - F = A - By overall balance
21 a = [P_H2O -F_H2O;1 -1] ;// Matrix of coefficient
22 b = [A*A_H2O;A] ;// Matrix of contants
23 x = a\b ;// Matrix of solutions- P = x(1) and F = x
        (2)
24
25 printf(' Original solution taken-
                                %.0i kg\n
        ',x(2) );
26 printf(' Final solution or kilograms of battery
        acid formed-
                                %.0i kg\n',x(1) );

```

Scilab code Exa 8.5 Drying

```

1 clear ;
2 clc;
3 // Example 8.5
4 printf('Example 8.5\n\n');
5 // Page no. 207
6 // Solution Fig E8.5
7
8 // Given
9 W = 100 ;// Water removed - [kg]
10 A_H2O = 0.80 ;// Fraction of water in A(intial fish
        cake)
11 A_BDC = 0.20 ;// Fraction of BDC(bone dry cake) in

```

```

    B(final dry fish cake)
12 B_H2O = 0.40 ;// Fraction of water in A(intial fish
    cake)
13 B_BDC = 0.60 ;// Fraction of BDC(bone dry cake) in
    B(final dry fish cake)
14
15 // By analysis for degree of freedom , DOF comes to
    be zero
16 // Solve following equations simultaneously for A
    and B,
17 // A*A_H2O = B*B_H2O + W - By H2O balance
18 // A = B + W - By overall balance
19 a = [A_H2O -B_H2O;1 -1] ;// Matrix of coefficient
20 b = [W;W] ;// Matrix of contants
21 x = a\b; // Matrix of solutions- A = x(1) and B = x
    (2)
22
23 printf('Weight of the fish cake originally put into
    dryer - %.0i kg\n',x(1) );

```

Scilab code Exa 8.6 Crystallizaton

```

1 clear ;
2 clc;
3 // Example 8.6
4 printf('Example 8.6\n\n');
5 // Page no. 209
6 // Solution
7
8 // Composition of initial solution at 30 degree C
9 s_30 = 38.8 ;// solublity of Na2CO3 at 30 degree C,
    by using the table for solublity of Na2CO3-[g
    Na2CO3/100 g H2O]
10 If_Na2CO3 = s_30/(s_30+100) ;// Initial mass
    fraction of Na2CO3

```

```

11 If_H2O = 1-If_Na2CO3 ;// Initial mass fraction of
    H2O
12
13 // Composition of crystals
14 // Basis : 1g mol Na2CO3.10H2O
15 n_mol_Na2CO3 = 1 ;// Number of moles of Na2CO3
16 n_mol_H2O = 10 ;// Number of moles of H2O
17 mwt_Na2CO3 = 106 ;// mol. wt of Na2CO3
18 mwt_H2O = 18 ;// mol. wt of H2O
19 m_Na2CO3 = mwt_Na2CO3*n_mol_Na2CO3 ;// Mass of
    Na2CO3
20 m_H2O = mwt_H2O*n_mol_H2O ;// Mass of H2O
21 Cf_Na2CO3 = m_Na2CO3/(m_Na2CO3+m_H2O) ;// mass
    fraction of Na2CO3
22 Cf_H2O = 1-Cf_Na2CO3 ;// mass fraction of H2O
23
24 n_un = 9 ;// Number of unknowns in the given problem
25 n_ie = 9 ;// Number of independent equations
26 d_o_f = n_un-n_ie ;// Number of degree of freedom
27 printf('Number of degree of freedom for the given
    system is %i .\n',d_o_f);
28
29 // Final composition of tank
30 //Basis :I = 10000 kg
31 // Material balance reduces to Accumulation =
    final -initial = in-out (but in = 0)
32 I = 10000 ;//initial amount of saturated solution -[
    kg]
33 amt_C = 3000 ;// Amount of crystals formed -[kg]
34 Fm_Na2CO3 = I*If_Na2CO3 -amt_C*Cf_Na2CO3 ;// Mass
    balance of Na2CO3
35 Fm_H2O = I*If_H2O -amt_C*Cf_H2O ;// Mass balance of
    H2O
36
37 //To find temperature ,T
38 s_T = (Fm_Na2CO3/Fm_H2O)*100 ;// Solubility of
    Na2CO3 at temperature T
39 s_20 = 21.5 ;//Solubility of Na2CO3 at temperature 20

```

```

    degree C ,from given table -[g Na2CO3/100 g H2O]
40 // Find T by interpolation
41 T = 30 - ((s_30 - s_T) / (s_30 - s_20)) * (30 - 20) ; //
    Temperature -[degree C]
42 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

```

1 clear ;
2 clc;
3 // Example 8.7
4 printf('Example 8.7\n\n');
5 // Page no. 213
6 // Solution
7
8 // Write given data
9 B_in = 1.1 ; // Flow rate in of blood -[L/min]
10 B_out = 1.2; // Flow rate out of blood -[L/min]
11 S_in = 1.7; // Flow rate in of solution -[L/min]
12
13 // Composition of input blood
14 B_in_CR = 2.72 ; // [g/L]
15 B_in_UR = 1.16 ; // [g/L]
16 B_in_U = 18 ; // [g/L]
17 B_in_P = 0.77 ; // [g/L]
18 B_in_K = 5.77 ; // [g/L]
19 B_in_Na = 13.0 ; // [g/L]
20 B_in_water = 1100 ; // [mL/min]
21
22 // Composition of output blood
23 B_out_CR = 0.120 ; // [g/L]
24 B_out_UR = 0.060; // [g/L]
25 B_out_U = 1.51 ; // [g/L]

```



```

26 B_out_P = 0.040 ;//[g/L]
27 B_out_K = 0.120 ;//[g/L]
28 B_out_Na = 3.21 ;//[g/L]
29 B_out_water = 1200 ;//[mL/min]
30
31 n_un = 7 ;// Number of unknowns in the given problem
32 n_ie = 7 ;// Number of independent equations
33 d_o_f = n_un-n_ie ;// Number of degree of freedom
34 printf('Number of degree of freedom for the given
        system is %i .\n\n',d_o_f);
35
36 // Water balance in grams, assuming 1 ml is
        equivalent to 1 g
37 S_in_water = 1700 ;//[ml/min]
38 S_out_water = B_in_water+ S_in_water - B_out_water;
39 S_out = S_out_water/1000 ;//[L/min]
40 printf(' Flow rate of water in output solution is %
        .2f L/min.\n\n',S_out);
41
42 // The component balance in grams for CR,U,P,K
        and Na are
43 S_out_CR = (B_in*B_in_CR - B_out*B_out_CR)/S_out;
44 S_out_UR = (B_in*B_in_UR - B_out*B_out_UR)/S_out;
45 S_out_U = (B_in*B_in_U - B_out*B_out_U)/S_out;
46 S_out_P = (B_in*B_in_P - B_out*B_out_P)/S_out;
47 S_out_K = (B_in*B_in_K - B_out*B_out_K)/S_out;
48 S_out_Na = (B_in*B_in_Na - B_out*B_out_Na)/S_out;
49 printf(' Component          Concentration(g/L) in output
        Dialysis solution \n');
50 printf(' UR          %.2f \n',S_out_UR);
51 printf(' CR          %.2f \n',S_out_CR);
52 printf(' U          %.2f \n',S_out_U);
53 printf(' P          %.2f \n',S_out_P);
54 printf(' K          %.2f \n',S_out_K);
55 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

```
1 clear;
2 clc;
3 // Example 9.1
4 printf('Example 9.1\n\n');
5 // Page no. 228
6 // Solution
7
8 // Given
9 //Main eqn. C6H12O6 + aO2 ----> bCO2 + cH2O
10 // By carbon balance
11 b = 6 ;
12
13 //By hydrogen balance
14 c = 6;
15
16 //Balancing oxygen in reaction
17 a = (c*1+b*2-6)/2;
18 printf('Value of a is %i\n',a);
19 printf('Value of b is %i\n',b);
```

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

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

```
1 clear ;
2 clc;
3 // Example 9.2
4 printf('Example 9.2\n\n');
5 // Page no. 229
6 // Solution
7
8 m_CO2 = 44.0 ;//molecular wt-[g]
9 m_C7H16 = 100.1 ;//molecular wt-[g]
10 p_con = 50 ;// percentage conversion of CO2 to dry
    ice
11 amt_di = 500 ;// amount of dry ice to be produce per
    hour-[kg]
12 // By using the given equation
13 amt_C7H16 = (amt_di*m_C7H16)/((p_con/100)*m_CO2*7) ;
    // [kg]
14 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 occurs

```
1 clear ;
2 clc;
3 // Example 9.3
4 printf('Example 9.3\n\n');
5 // Page no. 230
```

```

6 // Solution
7
8 m_CaCO3 = 100.1 ;//molecular wt-[g]
9 m_MgCO3 = 84.32 ;//molecular wt-[g]
10 m_CaO = 56.08 ;//molecular wt-[g]
11 m_MgO = 40.32 ;//molecular wt-[g]
12 m_CO2 = 44.0 ;//molecular wt-[g]
13
14
15 // Limestone analysis
16 p_CaCO3 = 92.89 ;// percentage of CaCO3
17 p_MgCO3 = 5.41 ;// percentage of MgCO3
18 inrt = 1.7 ;//percentage of inert
19
20 //(a)
21 amt_CaO = (((p_CaCO3/100)*m_CaO)/m_CaCO3)*2000 ;//
    Pounds of CaO produced from 1 ton(2000lb) of
    limestone
22 printf(' Amount of CaO produced from 1 ton(2000lb)
    of limestone is %.0f lb.\n',amt_CaO);
23
24 //(b)
25 mol_CaCO3 = (p_CaCO3/100)/m_CaCO3 ;// lb mol of
    CaCO3
26 mol_MgCO3 = (p_MgCO3/100)/m_MgCO3 ;// lb mol of
    MgCO3
27 total_mol = mol_CaCO3+mol_MgCO3;
28 amt_CO2 = total_mol*m_CO2 ;// Amount of CO2
    recovered per pound of limestone-[lb]
29 printf(' Amount of CO2 recovered per pound of
    limestone is %.3f lb.\n',amt_CO2);
30
31 //(c)
32 amt_CaO = m_CaO*mol_CaCO3 ;// since lb mol of CaO =
    CaCO3
33 amt_MgO = m_MgO*mol_MgCO3 ;// since lb mol of MgO =
    MgCO3
34 total_lime = amt_CaO+amt_MgO+(inrt)/100 ;// total

```

```

    amount of lime per pound limestone
35 amt_lmst = 2000/total_lime ;// Amount of limestone
    required to make 1 ton(2000lb) of lime
36 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

```

1 clear ;
2 clc;
3 // Example 9.4
4 printf('Example 9.4\n\n');
5 // Page no. 235
6 // Solution
7
8 f_NH3 = 5 ;// NH3 in feed-[g]
9 f_N2 = 100 ;// N2 in feed-[g]
10 f_H2 = 50 ;// H2 in feed-[g]
11 p_NH3 = 90 ;// NH3 in product-[g]
12 m_NH3 = 17 ;// Molecular wt. of NH3-[g]
13 m_N2 = 28 ;// Molecular wt. of N2-[g]
14 m_H2 = 2 ;// Molecular wt. of H2-[g]
15
16 // Extent of reaction can be calculated by using eqn
    . 9.3
17 // For NH3
18 ni = p_NH3/m_NH3 ;//[g mol NH3]
19 nio = f_NH3/m_NH3 ;//[g mol NH3]
20 vi = 2 ;// coefficient of NH3
21 ex_r = (ni-nio)/vi ;// Extent of reaction - moles
    reacting
22
23 //Determine H2 and N2 in product of reaction by Eqn.
    9.4
24 // For N2

```

```

25 nio_N2 = f_N2/m_N2 ;//[g mol N2]
26 vi_N2 = -1 ;// coefficient of N2
27 ni_N2 = nio_N2 + vi_N2*ex_r ;//N2 in product of
    reaction-[g moles ]
28 m_N2 = ni_N2*m_N2 ;// mass of N2 in product of
    reaction-[g]
29 printf(' N2 in product of reaction is %.2f g moles
    \n',ni_N2);
30 printf(' Mass of N2 in product of reaction is %.2
    f g \n',m_N2);
31 // For H2
32 nio_H2 = f_H2/m_H2 ;//[g mol H2]
33 vi_H2 = -3 ;// coefficient of H2
34 ni_H2 = nio_H2 + vi_H2*ex_r ;//H2 in product of
    reaction-[g moles ]
35 m_H2 = ni_H2*m_H2 ;// mass of H2 in product of
    reaction-[g]
36 printf(' \n H2 in product of reaction is %.2f g
    moles \n',ni_H2);
37 printf(' Mass of H2 in product of reaction is %.2f
    g \n',m_H2);
38
39 // ARP
40 m_SO2 = 64 ;// Molecular wt.of SO2-[g]
41 mol_SO2 = 2 ;// moles of SO2
42 ARP = (1/m_NH3)/(mol_SO2/m_SO2);
43 printf(' \n ARP is %.2f \n',ARP);

```

Scilab code Exa 9.5 Calculation of Limiting and Excess Reactants

```

1 clear ;
2 clc;
3 // Example 9.5
4 printf('Example 9.5\n\n');
5 // Page no. 238

```

```

6 // Solution
7
8 f_N2 = 10 ;// N2 in feed-[g]
9 f_H2 = 10 ;// H2 in feed-[g]
10 m_NH3 = 17.02; // Molecular wt. of NH3-[g]
11 m_N2 = 28 ;// Molecular wt. of N2-[g]
12 m_H2 = 2 ;// Molecular wt. of H2-[g]
13
14 // Extent of reaction can be calculated by using eqn
    . 9.3
15 // Based on N2
16 nio_N2 = f_N2/m_N2 ;//[g mol N2]
17 vi_N2 = -1 ;// coefficient of N2
18 ex_N2 = -(nio_N2)/vi_N2 ;// Max. extent of reaction
    based on N2
19
20 // Based on H2
21 nio_H2 = f_H2/m_H2 ;//[g mol H2]
22 vi_H2 = -3 ;// coefficient of H2
23 ex_H2 = -(nio_H2)/vi_H2 ;// Max. extent of reaction
    based on H2
24
25 //(a)
26 vi_NH3 = 2 ;// coefficient of NH3
27 mx_NH3 = ex_N2*vi_NH3*m_NH3 ;// Max. amount of NH3
    that can be produced
28 printf(' (a) Max. amount of NH3 that can be produced
    is %.1f g\n',mx_NH3);
29
30 //(b) and (c)
31 if (ex_H2 > ex_N2 )
32     printf(' (b) N2 is limiting reactant \n');
33     printf(' (c) H2 is excess reactant \n');
34     ex_r = ex_N2;
35     else
36     printf(' (b) H2 is limiting reactant \n');
37     printf(' (c) N2 is excess reactant \n');
38     ex_r = ex_H2 ;

```

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

```
1 clear;
2 clc;
3 // Example 9.6
4 printf('Example 9.6\n\n');
5 // Page no. 242
6 // Solution
7
8 //(a)
9 mol_bms = 0.59 ;// Biomass produced per g mol of
    glucose-[g mol biomass/ g mol glucose]
10 mw_bms = 23.74 ;// molecular wt. of biomass -[g]
11 mw_gls = 180.0 ;// molecular wt. of glucose -[g]
12 ms_bms = (mol_bms*mw_bms)/mw_gls ;// Biomass
    produced per gram of glucose-[g biomass/ g
    glucose]
13 printf('(a) Biomass produced per gram of glucose is
    %.4f g biomass/ g glucose.',ms_bms);
14
15 //(b)
16 mol_etol = 1.3 ;//Ethanol produced per g mol of
    glucose-[g mol ethanol/ g mol glucose]
17 mw_etol = 46.0 ;// molecular wt. of ethanol -[g]
18 ms_etol = (mol_etol*mw_etol)/mw_gls ;// Ethanol
    produced per gram of glucose-[g ethanol/ g
    glucose]
19 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


```

1 clear ;
2 clc;
3 // Example 9.7
4 printf('Example 9.7\n\n');
5 // Page no. 243
6 // Solution
7
8 //Basis: 3 g mol H2 by reaction (a)
9 // 0.50 g mol C2H4 by reaction (b)
10 // by analysing reaction (a) 0.50 g mol C2H4
    corresponds to 0.50 g mol H2 produced in reaction
    (b)
11 // By using reaction (a)
12 H2_a = 3-0.50 ;// H2 produced in reaction (a)
13 C_a = (2/3)*H2_a ;// Nanotubes(the C) produced by
    reaction (a)
14 sel = C_a/0.50 ;// Selectivity of C relative to C2H4
    -[g mol C/ g mol C2H4]
15 printf('Selectivity of C relative to C2H4 is %.2f g
    mol C/ g mol C2H4.\n',sel)

```

Scilab code Exa 9.8 Calculation of various terms Pertaning to Reaction

```

1 clear;
2 clc;
3 // Example 9.8
4 printf('Example 9.8\n\n');
5 // Page no. 244
6 // Solution
7
8 m_C3H6 = 42.08; // molecular wt. of propene -[g]
9 m_C3H5Cl = 76.53 ; // molecular wt. of C3H5Cl-[g]
10 m_C3H6Cl2 = 112.99 ; // molecular wt. of C3H6Cl2-[g]
11 // Product analysis
12 pm1_C12 = 141.0 ; // [g mol]

```

```

13 pml_C3H6 = 651.0 ;//[g mol]
14 pml_C3H5Cl = 4.6 ;// [g mol]
15 pml_C3H6Cl2 = 24.5 ;// [g mol]
16 pml_HCL = 4.6 ;//[g mol]
17
18 //(a)
19 a_Cl = pml_C3H5Cl; // Chlorine reacted by eqn.(a)
20 b_Cl = pml_C3H6Cl2 ;// Chlorine reacted by eqn.(b)
21 fed_Cl = pml_Cl2+a_Cl+b_Cl ;// Total chlorine fed to
    reactor-[g mol]
22 //by analysing reaction (a) and (b)
23 a_C3H6 = a_Cl+b_Cl ;// C3H6 reacted by reaction (a)
24 fed_C3H6 = pml_C3H6+a_C3H6 ;//Total C3H6 fed to
    reactor-[g mol]
25 printf('(a) Total chlorine fed to reactor is %.2f g
    mol \n',fed_Cl);
26 printf('    Total C3H6 fed to reactor is %.2f g
    mol \n',fed_C3H6);
27
28 //(b) and (c)
29 // Extent of reaction can be calculated by using eqn
    . 9.3
30 // Based on C3H6
31 nio_C3H6 = fed_C3H6 ;//[g mol C3H6]
32 vi_C3H6 = -1 ;// coefficient of C3H6
33 ex_C3H6 = -(nio_C3H6)/vi_C3H6 ;// Max. extent of
    reaction based on C3H6
34
35 // Based on Cl2
36 nio_Cl2 = fed_Cl; //[g mol Cl2]
37 vi_Cl2 = -1 ;// coefficient of Cl2
38 ex_Cl2 = -(nio_Cl2)/vi_Cl2 ;// Max. extent of
    reaction based on Cl2
39
40 if (ex_Cl2 > ex_C3H6 )
41     printf(' \n (b) C3H6 is limiting reactant \n');
42     printf('    (c)Cl2 is excess reactant \n');
43     ex_r = ex_C3H6;

```

```

44     else
45     printf(' \n (b) Cl2 is limiting reactant \n');
46     printf(' (c) C3H6 is excess reactant \n');
47     ex_r = ex_Cl2;
48     end
49
50     //(d)
51     fr_cn = pml_C3H5Cl/fed_C3H6 ;//Fractional conversion
           of C3H6 to C3H5Cl
52     printf(' \n (d) Fractional conversion of C3H6 to
           C3H5Cl is %.2e \n',fr_cn);
53
54     //(e)
55     sel = pml_C3H5Cl/pml_C3H6Cl2 ;// Selectivity of
           C3H5Cl relative to C3H6Cl2
56     printf(' \n (e) Selectivity of C3H5Cl relative to
           C3H6Cl2 is %.2f g mol C3H5Cl/g mol C3H6Cl2 \n',
           sel);
57
58     //(f)
59     yld = (m_C3H5Cl*pml_C3H5Cl)/(m_C3H6*fed_C3H6) ;//
           Yield of C3H5Cl per g C3H6 fed to reactor
60     printf(' \n (f) Yield of C3H5Cl per g C3H6 fed to
           reactor is %.3f g C3H5Cl/g C3H6 \n',yld);
61
62     //(g)
63     vi_C3H5Cl = 1 ;// coefficient of C3H5Cl
64     vi_C3H6Cl2 = 1 ;// coefficient of C3H6Cl2
65     ex_a = (pml_C3H5Cl-0)/vi_C3H5Cl ;// Extent of
           reaction a as C3H5Cl is produced only in reaction
           a
66     ex_b = (pml_C3H6Cl2-0)/vi_C3H6Cl2 ;// Extent of
           reaction b as C3H6Cl2 is produced only in
           reaction b
67     printf(' \n (g) Extent of reaction a as C3H5Cl is
           produced only in reaction a is %.1f \n',ex_a);
68     printf('           Extent of reaction b as C3H6Cl2 is
           produced only in reaction b %.1f \n',ex_b);

```

```
69
70 //(h)
71 in_Cl = fed_Cl*2 ;//Entering Cl -[g mol]
72 out_Cl = pml_HCL ;// Exiting Cl in HCl-[g mol]
73 ef_w = out_Cl/in_Cl ;// Mole efficiency of waste
74 ef_pr = 1-ef_w ;// Mole efficiency of product
75 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

```
1 clear;
2 clc;
3 // Example 10.1
4 printf('Example 10.1\n\n');
5 //Page no. 264
6 // Solution
7
8 F = 100 ;// feed to the reactor -[g mol]
9 // Composition of feed
10 CH4 = 0.4*F ;// [g mol]
11 C12 = 0.5*F ;// [g mol]
12 N2= 0.1*F ;//[g mol]
13
14 // Extent of reaction can be calculated by using eqn
   . 9.3
15 // Based on CH4
16 nio_CH4 = CH4 ;//[g mol CH4]
17 vi_CH4 = -1 ;// coefficient of CH4
18 ex_CH4 = -(nio_CH4)/vi_CH4 ;// Max. extent of
```

```

    reaction based on CH4
19
20 // Based on Cl2
21 nio_Cl2 = Cl2 ;//[g mol Cl2]
22 vi_Cl2 = -1 ;// coefficient of Cl2
23 ex_Cl2 = -(nio_Cl2)/vi_Cl2 ;// Max. extent of
    reaction based on Cl2
24
25 if (ex_Cl2 > ex_CH4 )
26     printf(' \n CH4 is limiting reactant \n');
27     else
28     printf(' \n (b) Cl2 is limiting reactant \n');
29 end
30 // By execution of above block its clear that CH4 is
    limiting reactant , therefore extent of reaction
    is
31 cn_CH4 = 67/100 ;// percentage conversion of CH4
32 ex_r = (-cn_CH4)*CH4/vi_CH4 ;// extent of reaction
33 printf(' extent of reaction is %.1f g moles reacting
    \n',ex_r);
34
35 n_un = 11 ;// Number of unknowns in the given
    problem
36 n_ie = 11 ;// Number of independent equations
37 d_o_f = n_un-n_ie ;// Number of degree of freedom
38 printf(' Number of degree of freedom for the given
    system is %i \n',d_o_f);
39
40 // Product composition using species balance using
    eqn.10.2
41 vi_CH3Cl = 1;
42 vi_HCl = 1;
43 vi_N2 = 0;
44 p_CH4 = CH4+(vi_CH4*ex_r);// [g mol]
45 p_Cl2 = Cl2+(vi_Cl2*ex_r);// [g mol]
46 p_CH3Cl = 0+(vi_CH3Cl*ex_r);// [g mol]
47 p_HCl = 0+(vi_HCl*ex_r);// [g mol]
48 p_N2 = N2+(vi_N2*ex_r);// [g mol]

```

```

49 // As we have taken F = 100 so answers we are
    getting can be directly used as percentage
    composition
50 printf('\n\nComposition of product stream in %% g
    mol of products\n');
51 printf('\nProduct           Percentage g mol\n');
52 printf('\nCH4              %.1f%% g mol\n',p_CH4);
53 printf('\nC12              %.1f%% g mol\n',p_C12);
54 printf('\nCH3Cl            %.1f%% g mol\n',p_CH3Cl
    );
55 printf('\nHCl              %.1f%% g mol\n',p_HCl);
56 printf('\nN2              %.1f%% g mol\n',p_N2);

```

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

```

1 clear ;
2 clc;
3 // Example 10.2
4 printf('Example 10.2\n\n');
5 // Page no. 266
6 // Solution
7
8 S = 5000 ;// Sulphur [lb]
9 // Composition of feed
10 CH4 = 80 ;// [%]
11 H2S = 20 ;// [%]
12
13 n_un = 11 ;// Number of unknowns in the given
    problem
14 n_ie = 11 ;// Number of independent equations
15 d_o_f = n_un-n_ie ;// Number of degree of freedom
16 printf('Number of degree of freedom for the given
    system is %i \n',d_o_f);
17 m_S = 32.0 ;//molecular wt. of S -[lb]

```

```

18 mol_S = S/32.0;
19 // Extent of reaction can be calculated by using eqn
   . 9.3
20 // Based on S
21 nio_S = 0 ;//[g mol S]
22 ni_S = mol_S ;//[g mol S]
23 vi_S = 3 ;// coefficient of S -from given reaction
24 ex_r = (ni_S-nio_S)/vi_S ;// Extent of reaction
   based on S
25 printf(' Extent of reaction is %.1f g moles reacting
   \n',ex_r);
26
27 // Product composition
28 vi_H2O = 2 ;// coefficient of H2O
29 vi_H2S = -2 ;// coefficient of H2S
30 vi_SO2 = -1 ;//coefficient of SO2
31 vi_CH4 = 0 ;//coefficient of CH4
32 P_H2O = 0+(vi_H2O*ex_r);// [lb mol]
33 P_H2S = P_H2O/10 ;//[lb mol]
34 P_SO2 = 3*P_H2S ;//[lb mol]
35
36 F = (P_H2S-vi_H2S*ex_r)/(H2S/100) ;// total feed-[lb
   mol]
37 F_SO2 = P_SO2-(vi_SO2*ex_r);// feed rate of SO2- [lb
   mol]
38 F_CH4 = (CH4/100)*F+vi_CH4*ex_r ;//feed rate of CH4-
   [lb mol]
39 F_H2S = ((H2S/100)*F) ;// feed rate of H2S-[lb mol]
40
41 // 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)
42 f_cn = -(vi_H2S*ex_r)/((H2S/100)*F) ;// Fractional
   conversion of limiting reagent
43
44 printf('\n(1) Feed rate of H2S- %.1f lb mol\n',F_H2S
   );

```



```

45 printf('(2)Feed rate of SO2- %.1f lb mol\n',F_S02);
46 printf('(3)Fractional conversion of limiting reagent
    - %.2f \n',f_cn);

```

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

```

1 clear ;
2 clc;
3 // Example 10.3
4 printf('Example 10.3\n\n');
5 // Page no. 270
6 // Solution
7
8 F = 1 ;//CH3OH -[gmol]
9 // Extent of reactions can be calculated by using
    eqn. 10.5
10 // For reaction 1 based on CH3OH is limiting reagent
11 f_cn = 90 ;//[%]
12 vi_CH3OH = -1 ;//coefficint of CH3OH
13 ex_r1 = (-90/100)/vi_CH3OH ;// Extent of reaction
    based on CH3OH
14 printf(' Extent of reaction 1 is %.2f g moles
    reacting \n',ex_r1);
15 //For reaction 2
16 yld = 75 ;//[%]
17 ex_r2 = ex_r1-(F*(yld/100));
18 printf(' Extent of reaction 2 is %.2f g moles
    reacting \n',ex_r2);
19
20 // For amount of air
21 // Entering O2 is twice the O2 required by reaction
    1,therefore
22 f_O2 = 0.21 ;// mol. fraction of O2
23 f_N2 = 0.79 ;// mol. fraction of N2

```

```

24 n_O2 = 2*((1/2)*F) ;// entering oxygen -[g mol]
25 air = n_O2/f_O2 ;// Amount of air entering
26 n_N2 = air-n_O2 ;// entering nitrogen -[g mol]
27
28 // Degree of freedom analysis
29 n_un = 11 ;// Number of unknowns in the given
    problem
30 n_ie = 11 ;// Number of independent equations
31 d_o_f = n_un-n_ie ;// Number of degree of freedom
32 printf(' Number of degree of freedom for the given
    system is %i \n',d_o_f);
33
34 // Reaction 1
35 v1_CH3OH = -1 ;//coefficint of CH3OH
36 v1_O2 = -1/2 ;//coefficint of O2
37 v1_CH2O = 1 ;//coefficint of CH2O
38 v1_H2O = 1 ;//coefficint of H2O
39 v1_CO = 0 ;//coefficint of CO
40 //Reaction 2
41 v2_O2 = -1/2 ;//coefficint of O2
42 v2_CH2O = -1 ;//coefficint of CH2O
43 v2_H2O = 1 ;//coefficint of H2O
44 v2_CO = 1 ;//coefficint of CO
45 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]
46
47 no_CH3OH = F+(v1_CH3OH*ex_r1)+0 ;// [g mol]
48 no_O2 = n_O2+(v1_O2*ex_r1)+v2_O2*ex_r2 ;// [g mol]
49 no_CH2O = 0 + v1_CH2O*ex_r1 +v2_CH2O*ex_r2 ;//[g mol
    ]
50 no_CO = 0+v1_CO*ex_r1 +v2_CO*ex_r2 ;//[g mol]
51 no_H2O = 0+v1_H2O*ex_r1+v2_H2O*ex_r2 ;// [g mol]
52 no_N2 = n_N2-0-0 ;// [g mol]
53
54 // Composition of product
55 y_CH3OH = (no_CH3OH/P )*100 ;// mole %
56 y_O2 = (no_O2/P)*100 ;// mole %

```

```

57 y_CH2O = (no_CH2O/P)*100 ;// mole %
58 y_CO = (no_CO/P)*100 ;// mole %
59 y_H2O = (no_H2O/P)*100 ;// mole %
60 y_N2 = (no_N2/P )*100; // mole %
61
62 printf('\nComposition of product\n');
63 printf('Component      mole percent\n');
64 printf(' CH3OH           %.1 f %%\n', y_CH3OH);
65 printf(' O2                %.1 f %%\n', y_O2);
66 printf(' CH2O              %.1 f %%\n', y_CH2O);
67 printf(' CO                 %.1 f %%\n', y_CO);
68 printf(' H2O                %.1 f %%\n', y_H2O);
69 printf(' N2                 %.1 f %%\n', y_N2);

```

Scilab code Exa 10.4 Analysis of Bioreactor

```

1 clear;
2 clc;
3 // Example 10.4
4 printf('Example 10.4\n\n');
5 // Page no. 273
6 // Solution
7
8 F = 4000 ;//[kg]
9 m_H2O = 18.02 ;// molecular masss of water
10 m_C6H12O6 = 180.1 ;// molecular mass of glucose
11 m_CO2 = 44 ;//molecular mass of CO2
12 m_C2H3CO2H = 72.03 ;// molecular mass of C2H3CO2H
13 m_C2H5OH = 46.05 ;// molecular mass of ethanol
14
15 p_H2O = 88 ;// [%]
16 p_C6H12O6 = 12; // [%]
17 ni_H2O = (F*p_H2O/100)/m_H2O ;// initial moles of
   water
18 ni_C6H12O6 = (F*(p_C6H12O6/100))/m_C6H12O6 ;//

```

```

    initial moles of glucose
19
20 // Degree of freedom analysis
21 n_un = 9 ;// Number of unknowns in the given problem
22 n_ie = 9 ;// Number of independent equations
23 d_o_f = n_un-n_ie ;// Number of degree of freedom
24 printf('Number of degree of freedom for the given
    system is %i \n',d_o_f);
25
26 ur_C6H12O6 = 90 ;//[kg]
27 pr_CO2 = 120 ;//[kg]
28 nf_C6H12O6 = ur_C6H12O6/m_C6H12O6 ;// [kmoles]
29 nf_CO2 = pr_CO2/m_CO2 ;// [kmoles]
30
31 // solve following eqn. (b) and (e)simultaneously
32 //(b):  nf_C6H12O6 = ni_C6H12O6+-1*ex_r1+-1*ex_r2
33 //(e):  nf_CO2 = 0+2*ex_r1+ 0*ex_r2
34 a = [-1 -1;2 0];// matrix formed by coefficients of
    unknowns
35 b = [(nf_C6H12O6-ni_C6H12O6);nf_CO2];//matrix formed
    by constant
36 x = a^(-1)*b;//matrix formed by solution
37 printf(' Extent of reaction 1 is %.3f kg moles
    reacting \n',x(1));
38 printf(' Extent of reaction 2 is %.3f kg moles
    reacting \n',x(2));
39
40 nf_H2O = ni_H2O+0*x(1) +2*x(2);// from eqn. (a)-[
    kmoles]
41 nf_C2H5OH = 0+2*x(1)+0*x(2);// from eqn.(c)-[kmoles]
42 nf_C2H3CO2H = 0+0*x(1)+2*x(2) ;//from eqn.(d)-[
    kmoles]
43 total_wt = m_H2O*nf_H2O + m_C6H12O6*nf_C6H12O6 +
    m_CO2*nf_CO2 + m_C2H3CO2H*nf_C2H3CO2H + m_C2H5OH*
    nf_C2H5OH;
44 mp_C2H5OH = (m_C2H5OH*nf_C2H5OH*100)/total_wt ;//
    Mass percent of ethanol -[%]
45 mp_C2H3CO2H = (m_C2H3CO2H*nf_C2H3CO2H*100)/total_wt

```

```

    ;//Mass percent of propenoic acid -[%]
46
47 printf(' \n Mass percent of ethanol in broth at end
    of fermentation process is %.1f %%\n',mp_C2H5OH)
    ;
48 printf(' Mass percent of propenoic acid in broth at
    end of fermentation process is %.1f %%\n',
    mp_C2H3C02H);

```

Scilab code Exa 10.5 Solution using Elemental Balances

```

1 clear ;
2 clc;
3 // Example 10.5
4 printf('Example 10.5\n\n');
5 // Page no. 279
6 // Solution
7
8 //(a)Solution of Example 10.1 using element balance
9 printf('(a)Solution of Example 10.1 using element
    balance\n');
10 F = 100 ;// feed to the reactor-[g mol]
11 // Composition of feed
12 CH4 = 0.4*F ;// [g mol]
13 C12 = 0.5*F ;// [g mol]
14 N2 = 0.1*F ;//[g mol]
15
16 n_un = 10 ;// Number of unknowns in the given
    problem(excluding extent of reaction)
17 n_ie = 10 ;// Number of independent equations
18 d_o_f = n_un-n_ie ;// Number of degree of freedom
19 printf('    Number of degree of freedom for the
    given system is %i \n',d_o_f);
20
21 // Extent of reaction can be calculated by using eqn

```

```

. 9.3
22 // Based on CH4
23 nio_CH4 = CH4 ;//[g mol CH4]
24 vi_CH4 = -1; // coefficient of CH4
25 ex_CH4 = -(nio_CH4)/vi_CH4 ;// Max. extent of
    reaction based on CH4
26
27 // Based on Cl2
28 nio_Cl2 = Cl2 ;//[g mol Cl2]
29 vi_Cl2 = -1 ;// coefficient of Cl2
30 ex_Cl2 = -(nio_Cl2)/vi_Cl2 ;// Max. extent of
    reaction based on Cl2
31
32 if (ex_Cl2 > ex_CH4 )
33     printf('    CH4 is limiting reactant \n');
34 else
35     printf(' \n (b) Cl2 is limiting reactant \n');
36 end
37 // By execution of above block its clear that CH4 is
    limiting reactant ,therefore
38 cn_CH4 = 67/100 ;// percentage conversion of CH4(
    limiting reagent)
39 no_CH4 = CH4-(cn_CH4*CH4) ;//CH4 in product -[g mol]
40
41 // Product composition using element balance
42 // By N2 balance
43 no_N2 = N2; //N2 in product -[g mol]
44
45 C = CH4 ;//moles of CH4 = moles of C (by molecular
    formula)
46 H = 4*CH4 ;// moles of H = 4*moles of CH4 (by
    molecular formula)
47 Cl = 2*Cl2 ;// moles of Cl = 2* moles of Cl2 (by
    molecular formula)
48 // Solving following 3 eqn. obtained from balance of
    C,H,Cl for 3 unknowns
49 //1. C-no_CH4*1 = 1*no_CH3Cl
50 //2. H-4*no_CH4 = 3*no_CH3Cl+no_HCl*1

```

```

51 //3. Cl = no_Cl2*2 + no_HCl*1+1*no_CH3Cl
52 a = [0 0 1;0 1 3;2 1 1] ;// matrix formed by
    coefficients of unknowns
53 b = [C-no_CH4*1;H-4*no_CH4;Cl] ;//matrix formed by
    constant
54 x = a^(-1)*b ;// matrix of solution
55
56 // As we have taken F = 100 so answers we are
    getting can be directly used as percentage
    composition
57 printf('\nComposition of product stream in %% g mol
    of products\n');
58 printf('Product          Percentage g mol\n');
59 printf('\nCH4            %.1f%% g mol\n',no_CH4)
    ;
60 printf('\nCl2            %.1f%% g mol\n',x(1));
61 printf('\nCH3Cl          %.1f%% g mol\n',x(3));
62 printf('\nHCl            %.1f%% g mol\n',x(2));
63 printf('\nN2             %.1f%% g mol\n',no_N2);
64
65 //(b)Solution of Example 10.3 using element balance
66 printf('
    -----
    ');
67 printf('\n\n(b)Solution of Example 10.3 using
    element balance\n');
68 F = 1 ;//CH3OH -[gmol]
69 yld = 75 ;//[ %]
70 cnv = 90 ;//conversion of methanol-[ %]
71
72 // For amount of air
73 // Entering O2 is twice the O2 required by reaction
    1,therefore
74 f_O2 = 0.21 ;// mol. fraction of O2
75 f_N2 = 0.79 ;// mol. fraction of N2
76 n_O2 = 2*((1/2)*F) ;// entering oxygen -[g mol]
77 air = n_O2/f_O2 ;// Amount of air entering
78 n_N2 = air-n_O2 ;// entering nitrogen -[g mol]

```

```

79
80 // Degree of freedom analysis
81 n_un = 9 ;// Number of unknowns in the given problem
      (excluding extent of reactions)
82 n_ie = 9 ;// Number of independent equations
83 d_o_f = n_un-n_ie ;// Number of degree of freedom
84 printf(' Number of degree of freedom for the given
      system is %i \n',d_o_f);
85
86 // Product composition using element balance
87 // By N2 balance
88 no_N2 = n_N2 ;// inert ,therefore input = output
89 C = 1*F ;//moles of C = moles of CH3OH (by
      molecular formula)
90 H = 4*F ;//moles of H = 4*moles of CH3OH (by
      molecular formula)
91 O = 1*F +2*n_o2 ;// moles of O = 1*moles of CH3OH +
      O in air
92 no_CH2O = yld/100 ;//[g mol]
93 no_CH3OH = F-((cnv/100)*F) ;// [g mol]
94
95 // Solving following 3 eqn. obtained from balance of
      C,H,O for 3 unknowns
96 a = [0 0 1;0 2 0;2 1 1] ;// matrix formed by
      coefficients of unknowns
97 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
98 x = a\b ;// matrix of solution
99
100 P = no_CH2O+no_CH3OH+no_N2+x(1)+x(2)+x(3);
101
102 // Composition of product
103 y_CH3OH = (no_CH3OH/P )*100 ;// mole %
104 y_o2 = ((x(1))/P)*100 ;// mole %
105 y_CH2O = (no_CH2O/P)*100 ;// mole %
106 y_CO = (x(3)/P)*100 ;// mole %
107 y_H2O = (x(2)/P)*100 ;// mole %

```



```

108 y_N2 = (no_N2/P )*100; // mole %
109
110
111 printf( '\nComposition of product\n');
112 printf( 'Component      mole percent\n');
113 printf( ' CH3OH          %.1 f %%\n', y_CH3OH);
114 printf( ' O2              %.1 f %%\n', y_O2);
115 printf( ' CH2O             %.1 f %%\n', y_CH2O);
116 printf( ' CO                 %.1 f %%\n', y_CO);
117 printf( ' H2O                 %.1 f %%\n', y_H2O);
118 printf( ' N2                  %.1 f %%\n', y_N2);

```

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

```

1 clear ;
2 clc ;
3 // Example 10.6
4 printf( 'Example 10.6\n\n');
5 // Page no. 281
6 // Solution
7
8 // Basis: P=100 // Product from the reactor -[g mol]
9 P=100 ; //Product from the reactor -[g mol]
10 // Composition of product
11 C3H8 = 0.195*P ; // [g mol]
12 C4H10 = 0.594*P ; // [g mol]
13 C5H12 = 0.211*P; // [g mol]
14
15 n_un = 3 ; // Number of unknowns in the given problem
    (excluding extent of reaction)
16 n_ie = 3 ; // Number of independent equations
17 d_o_f = n_un-n_ie ; // Number of degree of freedom
18 printf( 'Number of degree of freedom for the given
    system is %i \n', d_o_f);

```

```

19
20 C = C3H8*3+C4H10*4+C5H12*5 ;// moles of C on product
    side
21 H = C3H8*8+C4H10*10+C5H12*12 ;// moles of H on
    product side
22 // Solve following eqn.( obtained by element balance
    of C & H) for F and G
23 //8F+0G = C
24 //18F+2G = H
25 a = [8 0;18 2] ;// matrix formed by coefficients of
    unknowns
26 b = [C;H] ;//matrix formed by constant
27 x = a\b ;// matrix of solution
28
29 R = x(2)/x(1) ;// Ratio of H2 consumed to C8H18
    reacted = G/F
30 printf(' Molar ratio of H2 consumed to C8H18 reacted
    is %.3f \n',R);

```

Scilab code Exa 10.7 Excesss Air

```

1 clear ;
2 clc;
3 // Example 10.7
4 printf('Example 10.7\n\n');
5 // Page no. 286
6 // Solution
7
8 C3H8 = 20 ;//C3H8 burned in a test -[kg]
9 m_C3H8 = 44.09 ;// mol. wt . of 1 kmol C3H8
10 cf_O2 = 5 ;// coefficient of O2 in given reaction
11 air = 400 ;// Air given -[kg]
12 m_air = 29 ;// molecular wt. of 1kmol air -[kg]
13 O2p = 21 ;// percentage of O2 in air -[%]
14 p_CO2 = 44 ;// CO2 produced -[kg]

```

```

15 p_CO = 12 ;// CO produced -[kg]
16 O2 = (air*O2p/100)/(m_air) ;// amount of entering O2
    -[k mol]
17 rqO2 = (C3H8*cf_O2)/(m_C3H8) ;// Required O2 for
    given reaction
18 ex_air = ((O2-rqO2)*100)/rqO2 ;// Excess air percent
    -[%]
19 printf('Excess air percent is %.0f %%.\n',ex_air);

```

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

```

1 clear;
2 clc;
3 // Example 10.8
4 printf('Example 10.8\n\n');
5 // Page no. 287
6 // Solution
7
8 F = 16 ;// feed of CH4 -[kg]
9 CH4p = 100 ;// [%]
10 m_CH4 = 16 ;// mass of kmol of CH4-[kg]
11 mol_CH4 = (F*CH4p/100)/m_CH4 ;//k moles of CH4 in
    feed -[kmol]
12 air = 300 ;// Air given -[kg]
13 m_air = 29 ;// molecular wt. of 1kmol air -[kg]
14 mol_air = air/m_air ;// kmoles of air -[kmol]
15 O2p = 21 ;// percentage of O2 in air -[%]
16 O2 = (mol_air*O2p/100) ;// amount of entering O2-[k
    mol]
17 N2 = mol_air-O2 ;// amount of entering N2-[k mol]
18
19 // Degree of freedom analysis
20 n_un = 8 ;// Number of unknowns in the given problem
    (excluding extent of reactions)
21 n_ie = 8 ;// Number of independent equations

```

```

22 d_o_f = n_un-n_ie ;// Number of degree of freedom
23 printf('Number of degree of freedom for the given
        system is %i \n',d_o_f);
24
25 // Product composition analysis using element
        balance of C,H,O and N
26 p_N2 = N2 ;// inert
27 C_in = 1*mol_CH4 ;// kmoles of carbon in input-[kmol
        ]
28 H_in = 4*mol_CH4 ;// kmoles of hydrogen in input-[
        kmol]
29 O_in = 2*O2 ;// kmoles of oxygen in input-[kmol]
30 p_CO2 = C_in/1 ;//kmoles of CO2 in product obtained
        by carbon balance-[kmol]
31 p_H2O = H_in/2 ;//kmoles of H2O in product obtained
        by hydrogen balance-[kmol]
32 p_O2 = (O_in-(2*p_CO2+p_H2O))/2 ;//kmoles of O2 in
        product obtained by oxygen balance-[kmol]
33 p_CH4 = 0 ;// Complete reaction occurs
34 P = p_CH4 + p_N2+ p_CO2 + p_H2O + p_O2;
35
36 y_N2 = p_N2*100/P ;//[mol %]
37 y_CO2 = p_CO2*100/P ;//[mol %]
38 y_H2O = p_H2O*100/P ;//[mol %]
39 y_O2 = p_O2*100/P ;//[mol %]
40 y_CH4 = p_CH4*100/P ;//[mol %]
41
42 printf('\nComposition of product\n');
43 printf('Component           mole percent\n');
44 printf(' CH4                %.1f %%\n',y_CH4);
45 printf(' O2                  %.1f %%\n',y_O2);
46 printf(' CO2                  %.1f %%\n',y_CO2);
47 printf(' H2O                    %.1f %%\n',y_H2O);
48 printf(' N2                     %.1f %%\n',y_N2);

```

Scilab code Exa 10.9 Combustion of Coal

```
1 clear;
2 clc;
3 // Example 10.9
4 printf('Example 10.9\n\n');
5 // Page no. 290
6 // Solution
7
8 F = 100 ;// feed of coal -[lb]
9 // given coal composition-given
10 C = 83.05 ;//[%]
11 H = 4.45 ;//[%]
12 O = 3.36 ;//[%]
13 N = 1.08 ;//[%]
14 S = 0.70 ;//[%]
15 ash = 7.36;//[%]
16 H2O = 3.9 ;//[%]
17 w_C = 12 ;// mol. wt. of C
18 w_H = 1.008; //mol. wt. of H
19 w_O = 16 ;// mol. wt. of O
20 w_N = 14 ;// mol. wt. of N
21 w_S = 32 ;//mol. wt. of S
22
23 //given stack gas analysis-given
24 CO2 = 15.4 ;//[%]
25 CO = 0.0 ;//[%]
26 O2 = 4.0 ;//[%]
27 N2 = 80.6 ;//[%]
28 //given refuse analysis
29 ash_R = 86 ;//[%]
30 odr = 14 ;//[%]
31
32 H2O_air = .0048 ;// [lb H2O/lb dry air]
33 m_air = 29 ;// mol. wt. of air
34 mf_O2 = 0.21 ;// mole fraction of O2 in air
35 mf_N2 = 0.79 ;//mole fraction of N2 in air
36 m_H2O = 18 ;// mol. wt. of H2O
```

```

37
38 H_cl = (H2O*2)/m_H2O ;// lb mol of H in coal
    moisture
39 O_cl = H_cl/2 ;// lb mol of O in coal moisture
40
41 H_air = (H2O_air*m_air )/m_H2O;// lb mol of H per lb
    mol air
42 O_air = H_air/2 ;// lb mol of O per lb mol air
43
44 // Ash balance to get refuse(R)
45 R = ash/(ash_R/100) ;// Refuse-[lb]
46 //refuse composition
47 pub_cl = 14 ;// percentage of unburned coal in
    refuse-[%]
48 ub_cl = (14/100)*R ;// amount of unburned coal in
    refuse
49 C_p = (C/(100-ash))*ub_cl ;// C in unburned coal-[
    lb]
50 H_p = (H/(100-ash))*ub_cl ;// H in unburned coal-[
    lb]
51 O_p = (O/(100-ash))*ub_cl ;// O in unburned coal-[
    lb]
52 N_p = (N/(100-ash))*ub_cl ;// N in unburned coal-[
    lb]
53 S_p = (S/(100-ash))*ub_cl ;// S in unburned coal-[
    lb]
54 mol_C = C_p/w_C;// lb mol of C
55 mol_H = H_p/w_H ;// lb mol of H
56 mol_N = N_p/w_N ;// lb mol of N
57 mol_O = O_p/w_O ;// lb mol of O
58 mol_S = S_p/w_S ;// lb mol of S
59
60 // Degree of freedom analysis
61 n_un = 4 ;// Number of unknowns in the given problem
    (excluding extent of reactions)
62 n_ie = 4 ;// Number of independent equations
63 d_o_f = n_un-n_ie ;// Number of degree of freedom
64 printf('Number of degree of freedom for the given

```

```

        system is %i \n\n',d_o_f);
65
66 //Using element balance of C+S, N& H
67 P = (C/w_C + S/w_S - (mol_C+mol_S ))/.154 ;// mol of
        stack gas-[lb mol]
68 A = (2*P*.806 +2*mol_N-N/w_N)/(2*mf_N2) ;// mol of
        air -[lb mol]
69 W = (H/w_H +H_cl+H_air*A-mol_H)/2 ;// moles of exit
        water-[lb mol]
70 printf(' Moles of stack gas(P)      -      %.1f  lb mol
        \n',P);
71 printf(' Moles of air (A)          -      %.1f lb mol
        \n',A);
72 printf(' Moles of exit water(W)    -      %.1f lb mol
        \n',W);
73 // by using P,W , A and O2 balance we get 19.8 =
        20.3 , therefore difference is about 1%
74
75 //Calculation of excess air
76 // For O2 required
77 C_req = (C/w_C)/1 ;// O2 required by entering C
        given by reaction C+O2---->CO2 -[lb mol]
78 H_req = (H/w_H)/4 ;// O2 required by entering H by
        given reaction H2+(1/2)*O2---->H2O-[lb mol]
79 N_req = 0 ;// inert
80 O_req = (O/w_O)/2 ;// O2 required by entering O-[lb
        mol]
81 S_req = (S/w_S)/1 ;// O2 required by entering S-
        given by S+O2---->SO2 -[lb mol]
82 total_O2_req = C_req+H_req+N_req+O_req +S_req ;//
        Total oxygen required-[lb mol]
83 O2_in = A*mf_O2 ;// O2 entering in air
84 ex_air = 100*((O2_in-total_O2_req)/total_O2_req) ;//
        [% of excess air]
85 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 in a process with Multiple Units

```
1 clear ;
2 clc;
3 // Example 11.1
4 printf('Example 11.1\n\n');
5 // Page no. 311
6 // Solution
7
8 // Composition of each stream
9 w_A1 = 1 ; //concentration of A in 1
10 w_B2 = 1 ; // concentration of B in 2
11 w_A3 = 0.8 ; // concentration of A in 3
12 w_B3 = 0.2 ; // concentration of B in 3
13 w_C4 = 1 ; // concentration of C in 4
14 w_A5 = 0.571 ; //concentration of A in 5
15 w_B5 = 0.143 ; //concentration of B in 5
16 w_C5 = 0.286 ; //concentration of C in 5
17 w_D6 = 1 ; // concentration of D in 6
18 w_A7 = 0.714 ; // concentration of A in 7
```



```

19 w_B7 = 0.286 ;// concentration of B in 7
20 w_B8 = 0.333 ;//concentration of B in 8
21 w_C8 = .667 ;//concentration of C in 8
22
23 us1 = 2 ;// Species involved in unit 1
24 us2 = 3 ;// Species involved in unit 2
25 us3 = 4 ;// Species involved in unit 3
26 total_sp = us1+us2+us3 ;// Total species in system
27
28 // Element balance of all systems
29 printf('Number of possible equations are 9, they are
      as follows - \n');
30 printf(' Subsystem 1\n');
31 printf('      A: F1*w_A1+F2*0 = F3*w_A3      (a)\n
      ');
32 printf('      B:F1*0 + F2*w_B2 = F3*w_B3      (b)\n
      ');
33 printf(' Subsystem 2\n');
34 printf('      A: F3*w_A3+F4*0 = F5*w_A5      (c)\n
      ');
35 printf('      B:F3*w_B3 + F4*0 = F5*w_B5      (d)\n
      ');
36 printf('      C: F3*0+F4*w_C4 = F5*w_C5      (e)\n
      ');
37 printf(' Subsystem 3\n');
38 printf('      A: F5*w_A5+F6*0 = F7*w_A7+F8*0
      (f)\n');
39 printf('      B:F5*w_B5 + F6*0 = F7*0+F8*w_B8
      (g)\n');
40 printf('      C: F5*w_C5+F6*0 = F7*0+F8*w_C8
      (h)\n');
41 printf('      D:F5*w_C5+F6*0 = F7*0+F8*w_C8
      (i)\n');
42 printf('\n The above equations do not form a unique
      set\n');
43
44 // By inspection we can see that only 7 equations
      are independent

```

```

45 //Independent Equations are:
46 // Subsystem 1
47 //A: F1*w_A1+F2*0 = F3*w_A3 (a)
48 //B:F1*0 + F2*w_B2 = F3*w_B3 (b)
49 //Subsystem 2
50 //A: F3*w_A3+F4*0 = F5*w_A5 (c)
51 // C: F3*0+F4*w_C4 = F5*w_C5 (e)
52 // Subsystem 3
53 //A: F5*w_A5+F6*0 = F7*w_A7+F8*0 (f)
54 //B:F5*w_B5 + F6*0 = F7*0+F8*w_B8 (g)
55 //D:F5*w_C5+F6*0 = F7*0+F8*w_C8 (i)
56
57 printf('\n Number of independent equations are 7 \n'
);

```

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

```

1 clear;
2 clc;
3 // Example 11.2
4 printf('Example 11.2\n\n');
5 // Page no.315
6 // Solution
7
8 //Basis:1 hr
9 G = 1400 ;//[kg]
10 //Unit 1
11 // Degree of freedom analysis
12 n_un = 16 ;// Number of unknowns in the given
    problem(excluding extent of reactions)
13 n_ie = 16 ;// Number of independent equations
14 d_o_f = n_un-n_ie ;// Number of degree of freedom
15 printf('For unit 1 number of degree of freedom for
    the given system is %i .\n',d_o_f);

```

```

16 //Given
17 o1_air = 0.995 ;// Mass fraction of air at out of
    unit 1 in A
18 i1_air = 0.95 ;// Mass fraction air at in of unit 1
    in G
19 i1_wtr = 0.02; // Mass fraction water at in of unit 1
    in G
20 F1_wtr = 0.81 ;// Mass fraction of water at out of
    unit 1 in F
21 o1_wtr = 0.005 ;// Mass fraction of water at out of
    unit 1 in A
22 o2_wtr = 0.96 ;// Mass fraction of water at out of
    unit 2 in B
23 o3_wtr = 0.01; // Mass fraction of water at out of
    unit 3 in D
24 i1_act = 0.03 ;// Mass fraction of acetone at in of
    unit 1 in G
25 F1_act = 0.19 ;// Mass fraction of acetone at out
    of unit 1 in F
26 o3_act = 0.99 ;// Mass fraction of acetone at out
    of unit 3 in D
27 o2_act = 0.04 ;// Mass fraction of acetone at out
    of unit 2 in B
28
29 //Mass balance to get A ,W & F
30 A = G*i1_air/o1_air ;//air -[kg]
31 F = G*i1_act/F1_act ;//[kg]
32 W = (F*F1_wtr+A*o1_wtr-G*i1_wtr)/1 ;//Pure water in
    -[kg]
33 // unit 2 and 3
34 // Degree of freedom analysis
35 n_un = 9 ;// Number of unknowns in the given problem
    (excluding extent of reactions)
36 n_ie = 9 ;// Number of independent equations
37 d_o_f = n_un-n_ie ;// Number of degree of freedom
38 printf(' For unit 2 and 3 number of degree of
    freedom for the given system is %i .\n',d_o_f);
39 // Mass balance

```

```

40 // solving eqn (d)& (e) simultaneously for D and B
41 a = [o3_act o2_act;o3_wtr o2_wtr]; // Matrix formed
    by coefficients of unknown
42 b = [F*F1_act;F*F1_wtr]; // Matrix formed by constant
43 x = a\b ; // Solution matrix-x(1) = D and x(2) = B
44 printf('\n W-Pure water in to unit 1 - %.2f kg/hr\n\
    n',W);
45 printf(' A-Air out of unit 1 - %.2f kg/hr\n\
    ,A);
46 printf(' F-out of unit 1 - %.2f kg/hr\n\
    ,F);
47 printf(' B-out of unit 2 - %.2f kg/hr\n\
    ,x(2));
48 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 Reactions

```

1 clear ;
2 clc;
3 // Example 11.3
4 printf('Example 11.3\n\n');
5 //Page no. 318
6 // Solution
7
8 P = 6205 ;//[lb mol/hr]
9 //Given
10 amt_F = 560 ;//[bbl]
11 // Fuel oil(F) analysis
12 C_F = 0.50 ;// [mol fraction]
13 H2_F = 0.47 ;//[mol fraction]
14 S_F = 0.03 ;//[mol fraction]
15 // Natural Gas(G) analysis
16 CH4_G = 0.96 ;//[mol fraction]

```

```

17 C2H2_G = 0.02 ;//[mol fraction]
18 CO2_G = 0.02 ;//[mol fraction]
19 // Analysis of air into Gas furnace(A)
20 O2_A = 0.21 ;//[mol fraction]
21 N2_A = 0.79 ;//[mol fraction]
22 // Analysis of air into Oil furnace(A1)
23 O2_A1 = 0.20 ;//[mol fraction]
24 N2_A1 = 0.76 ;//[mol fraction]
25 CO2_A1 = 0.04 ;//[mol fraction]
26 //Stack gas(P) analysis
27 N2_P = .8493 ;//[mol fraction]
28 O2_P = .0413 ;//[mol fraction]
29 SO2_P = .0010 ;// [mol fraction]
30 CO2_P = .1084 ;//[mol fraction]
31
32 // Degree of freedom analysis
33 n_un = 5; // Number of unknowns in the given problem(
    excluding extent of reactions)
34 n_ie = 5 ;// Number of independent equations
35 d_o_f = n_un-n_ie; // Number of degree of freedom
36 printf('Number of degree of freedom for the given
    system is %i .\n',d_o_f);
37
38 // Elemental mole balance for 2N,2H,2O,S and C
39 // Use S balance to get F
40 F = P* SO2_P/S_F ;// [lb mol/hr]
41 //Solve other four balances to get G
42 //2H: G*(2*CH4_G+C2H2_G)+F*H2_F-W*1
43 //2N: A*N2_A+A1*N2_A1 = P*N2_P
44 //2O: A*(O2_A)+A1*(O2_A1+CO2_A1)+G*CO2_G-W*(1/2) =
    P*(O2_P+CO2_P+SO2_P)
45 //C: G*(CH4_G+2*C2H2_G+CO2_G)+F*C_F+A1*CO2_A1 = P*
    CO2_P
46 //Solving above eqns. by matrix method[G W A A1]
47 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
48 b = [-F*H2_F;P*N2_P;P*(O2_P+CO2_P+SO2_P);(P*CO2_P-F*

```

```

    C_F)];// matrix of constants
49 x = a\b ;// matrix of solutions x(1) = G,x(2) = W,x
    (3) = A & x(3) = A1
50 G = x(1);//[lb mol/hr]
51 m_F = 7.91 ;// Molecular wt. of fuel oil-[lb]
52 Fc = (F*m_F)/(7.578*42);// Fuel gas consumed -[bbl/
    hr]
53 time = amt_F/Fc ;// Time for which available fuel
    gas lasts -[hr]
54 printf('(1) Fuel gas consumed(F) is %.2f bbl/hr .\n
    ',Fc);
55 printf('(2) Time for which available fuel gas lasts
    is %.0f hr .\n',time);
56
57 // For increase in arsenic and mercury level
58 F_oil = Fc*42; //[gal/hr]
59 Em_ars2 = (3.96 *10^(-4))/1000 ;// [lb/gal]
60 Em_Hg2 = (5.92 *10^(-4))/1000 ;// [lb/gal]
61 ars_F = F_oil*Em_ars2 ;// Arsenic produced on
    burning oil-[lb]
62 Hg_F = F_oil*Em_Hg2 ;//Mercury produced on burning
    oil-[lb]
63 G_gas = G*359 ;//[ft^3/hr]
64 Em_ars1 = (2.30 *10^(-4))/10^6 ;// [lb/ft^3]
65 Em_Hg1 = (1.34 *10^(-4))/10^6 ;// [lb/ft^3]
66 ars_G = G_gas*Em_ars1; // Arsenic produced on
    burning Natural gas-[lb]
67 Hg_G = G_gas*Em_Hg1 ;//Mercury produced on burning
    Natural Gas-[lb]
68 in_ars = ((ars_F-ars_G)/ars_G)*100 ;//[% increase
    in Arsenic emission]
69 in_Hg = ((Hg_F-Hg_G)/Hg_G)*100 ; //[% increase in
    Mercury emission]
70 printf('(3) Increase in Arsenic emission is %.1f %%
    .\n',in_ars);
71 printf('(4) Increase in Mercury emission is %.1f %%
    .\n',in_Hg);

```

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

```
1 clear ;
2 clc ;
3 // Example 11.4
4 printf('Example 11.4\n\n');
5 // Page no. 322
6 // Solution fig E11.4
7
8 //Basis : M = 1000 lb
9 M = 1000 ;//[lb]
10 //Given
11 F_s = 16/100 ;// Fraction of sugar in F
12 F_w = 25/100 ;// Fraction of water in F
13 F_p = 59/100 ; // Fraction of pulp in F
14 D_p = 80/100 ; // Fraction of pulp in D
15 E_s = 13/100 ;// Fraction of sugar in E
16 E_p = 14/100 ;// Fraction of pulp in E
17 G_p = 95/100 ;// Fraction of pulp in G
18 H_s = 15/100 ;// Fraction of sugar in H
19 K_s = 40/100 ;// Fraction of sugar in K
20
21 // For crystallizer equations are
22 K_w = 1 - K_s ;// summation of wt. fraction is 1
23 K = M/K_s ;// By sugar balance -[lb]
24 L = K_w*K ;// By water balance -[lb]
25
26 // For evaporator equations are
27 H_w = 1- H_s ;//summation of wt. fraction is 1
28 H = K_s*K/H_s ;//By sugar balance -[lb]
29 J = H - K; //By overall balance -[lb]
30
31 // For screen equations are
```

```

32 E_w = 1 - (E_p + E_s) ; // summation of wt. fraction
    is 1
33 // solve E and G by simultaneous eqn. obtained by
    overall and pulp balance
34 a1 = [1 -1;E_p -G_p] ;// Matrix of coefficients of
    unknown
35 b1 = [H;0] ;//Matrix of constants
36 x1 = a1\b1 ;// Matrix of solutions ,x1(1) = E, x1(2)
    = G
37 E = x1(1) ;//[lb]
38 G = x1(2) ;//[lb]
39 G_s = (E_s*E - H_s *H )/G ;// By sugar balance
40 G_w = 1 -(G_s + G_p) ;// summation of wt. fraction
    is 1
41
42 // For mill equations are
43 // solve F and D by simultaneous eqn. obtained by
    overall and pulp balance
44 a2 = [1 -1;F_p -D_p] ;// Matrix of coefficients of
    unknown
45 b2 = [E;E_p*E] ;//Matrix of constants
46 x2 = a2\b2 ;// Matrix of solutions ,x2(1) = F, x2(2)
    = D
47 F = x2(1) ;//[lb]
48 D = x2(2) ;//[lb]
49 D_s = (F_s*F - E_s *E )/D ;// By sugar balance
50 D_w = 1 -(D_s + D_p) ; // summation of wt. fraction
    is 1
51
52 S_rec = M/(F*F_s) ; // Fraction of sugar recovered
53
54 printf('\nFlow streams and their respective
    compositions.\n');
55 printf('\n M = %.0f lb \n',M);
56 printf(' Sugar: %.2f \n',1);
57
58 printf('\n L = %.0f lb \n',L);
59 printf(' Water: %.2f\n',1);

```



```

60
61 printf( '\n K = %.0 f lb \n',K);
62 printf( '  Sugar: %.2 f\n',K_s);
63 printf( '  Water: %.2 f\n',K_w);
64
65 printf( '\n J = %.0 f lb \n',J);
66 printf( '  Water: %.2 f \n',1);
67
68 printf( '\n H = %.0 f lb \n',H);
69 printf( '  Sugar: %.2 f\n',H_s);
70 printf( '  Water: %.2 f\n',H_w);
71
72 printf( '\n G = %.0 f lb \n',G);
73 printf( '  Sugar: %.3 f\n',G_s);
74 printf( '  Water: %.3 f\n',G_w);
75 printf( '  Pulp  : %.2 f\n',G_p);
76
77 printf( '\n E = %.0 f lb \n',E);
78 printf( '  Sugar: %.2 f\n',E_s);
79 printf( '  Water: %.2 f\n',E_w);
80 printf( '  Pulp  : %.2 f\n',E_p);
81
82 printf( '\n D = %.0 f lb \n',D);
83 printf( '  Sugar: %.3 f\n',D_s);
84 printf( '  Water: %.3 f\n',D_w);
85 printf( '  Pulp  : %.2 f\n',D_p);
86
87 printf( '\n F = %.0 f lb \n',F);
88 printf( '  Sugar: %.2 f\n',F_s);
89 printf( '  Water: %.2 f\n',F_w);
90 printf( '  Pulp  : %.2 f\n',F_p);

```

Scilab code Exa 11.5 Production of a Hormone in connected Reactor

```
1 clear ;
```

```

2  clc;
3  // Example 11.5
4  printf('Example 11.5\n\n');
5  // Page no.324
6  // Solution
7
8  // Option 1
9  F = 15 ;//[L/hr]
10 cs_in = 10 ;//Nutrient conc. input vessel - [g
    nutrient/L substrate]
11 V1 = 100 ;// [L]
12 V2 = 50 ;//[L]
13 Yxs = 0.2 ;// [cells/g]
14 umax = 0.4 ;//[hr-1]
15 Ks = 2 ;//[g/L] - Monod constant
16 // Use eqn. 10.1 for balances and Monod eqns.
    applies to each vessel
17 //Cells: 0 - F/V * x_out + u * x_out - 0 = 0....(a)
18 //Nutrient: F/V * cs_in - F/V * cs_out + 0 - (u *
    x_out)/(Yxs) = 0.....(b)
19 //From eqn.(a) F/V = u(dilution rate)...(c)
20 // From eqn. (b) x_out = Yxs(cs_in - cs_out)....(d)
21 u1 = F/V1 ;//[hr-1] // [hr-1]
22 cs_out = (Ks * u1/umax)/(1 - (u1/umax)) ;//Nutrient
    conc. output vessel - [g nutrient/L substrate]
23 // Find x_out by eqn. (d)
24 x_out = Yxs * (cs_in - cs_out) ;//[g cells / L
    substrate]
25
26 //Option 2
27 //For vessel 1
28 u2 = F/V2;
29 cs_out1 = (Ks * u2/umax)/(1 - (u2/umax)) ;//
    Nutrient conc. output vessel - [g nutrient/L
    substrate]
30 x_out1 = Yxs * (cs_in - cs_out1) ;//[g cells / L
    substrate]
31 // For vessel 2

```

```

32 // Eqn. (a) is now  $F/V * x_{out1} - F/V * x_{out2} + u3$ 
    *  $x_{out2} = 0 \dots (e)$ 
33 // Eqn. (b) is now  $F/V * cs_{out1} - F/V * cs_{out2} +$ 
     $(u3 * x_{out2})/Y_{xs} = 0 \dots (f)$ 
34 //  $u3 = (u_{max} * cs_{out2}) / (K_s + cs_{out2}) \dots$  Monod
    Eqn  $\dots (g)$ 
35 // (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
36  $x_{out2} = 1.73$  ; // From eqn. (e),(f) and (g) - [g
    cells / L substrate]
37
38 printf('g cells/L from option 1 is %.2f.\n',x_out);
39 printf(' g cells/L from option 2 is %.2f.\n',x_out2)
    ;
40 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

```
1 clear;
2 clc;
3 //Page No.349
4 // Example 12.1
5 printf('Example 12.1\n\n');
6 // Solution
7
8 //(a) fig.E12.1a
9 F = 10000 ;//[lb/hr]
10 //Given
11 NaOH_F = 40/100 ;//[wt. fraction]
12 NaOH_P1 = 95/100 ;//[wt. fraction of NaOH filter
    cake]
13 NaOH_P2 = (0.05 * 45)/100 ;//[wt. fraction of NaOH
    in NaOH soln.]
14 H2O_P2 = (0.05 * 55)/100 ;//[wt. fraction of H2O in
    NaOH soln.]
```

```

15 NaOH_R = 45/100; //[wt. fraction]
16 NaOH_G = 50/100; //[wt. fraction]
17 //Get P from overall NaOH balance
18 P = (NaOH_F * F)/[NaOH_P1 + NaOH_P2] ;//[lb/hr]
19 // Get W from overall total balance
20 W = F-P ;// [lb/hr]
21
22 // Solve following eqn. simultaneously to get G & R
23 // NaOH_G * G = F * NaOH_F + NaOH_R * R (NaOH
    balance on crystallizer)
24 //G = R + P (overall balance)
25 a = [NaOH_G -NaOH_R;1 -1] ;// matrix formed of
    coefficients of unknown
26 b = [F * NaOH_F;P]; // matrix formed by constant
27 x = a\b ;// matrix of solutions . x(1) = G, x(2) = R
28 G = x(1) ;// [lb/hr]
29 R = x(2) ;// [lb/hr]
30 printf('(a) Flow rate of water removed by
    evaporator is %.1f lb/hr\n',W);
31 printf(' The recycle rate of the process is %.1
    f lb/hr\n',R);
32
33 // (b) fig.E12.1b
34 //given
35 NaOH_H = 45/100 ;//[wt fraction]
36 H2O_H = 55/100 ;//[wt fraction]
37 // Get H & G by solving following eqn.
    simultaneously
38 //NaOH_G * G = [NaOH_P1 + NaOH_P2] * P + NaOH_H *
    H (NaOH balance on crystallizer)
39 //H2O_G * G = H2O_P2 * P + H2O_H * H (H2O
    balance on crystallizer)
40 a1 = [NaOH_G -NaOH_H;NaOH_G -H2O_H] ;// matrix
    formed of coefficients of unknown
41 b1 = [((NaOH_P1 + NaOH_P2) * P);(H2O_P2) * P]; //
    matrix formed by constant
42 x1 = ((a1)^-1) * b1 ;// matrix of solutions nw_G =
    x1(1);H = x1(2)

```

```

43 nw_G1 = x1(1) ;// [lb/hr]
44 H = x1(2);// [lb/hr]
45 // By overall NaOH balance
46 nw_F = (NaOH_H * H + (NaOH_P1 + NaOH_P2) * P)/
    NaOH_F ;//[lb/hr]
47 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

```

1 clear ;
2 clc;
3 //Page No.357
4 // Example 12.2
5 printf('Example 12.2\n\n');
6 // Solution fig.E12.2
7
8 // Given
9 // Main reaction - C6H6 + 3H2 --> C6H12
10 F_Bz = 100 ;// Fresh benzene feed / basis - [mol]
11 con_Bz = .95 ;// Fraction of conversion of benzene
12 sp_con = .20 ;// Fraction of single pass conversion
13 ex_H2 = .20 ;// Fraction of exces H2 used in fresh
    feed
14 R_Bz = 22.74 ;// Benzene in Recycle stream - [mol %]
15 R_H2 = 78.26 ;// H2 in Recycle stream - [mol %]
16 TLV_Bz = 0.5 ;// TLV value of benzene -[ppm]
17 TLV_C6H12 = 300 ;// TLV value of cyclohexane -[ppm]
18 TLV_H2 = 1000 ;// TLV value of H2 -[ppm]
19
20 // Feed composition
21 F_H2 = F_Bz*3*(1+ex_H2) ;// H2 in Feed - [mol]
22 F = F_Bz + F_H2 ;// Total feed - [mol]
23
24 // Use Eqn. 12.1 to get extent of reaction -(ex_r)

```

```

25 ex_r = con_Bz*F_Bz/(-(-1)) ;// Extent of reaction
26
27 // get composition of P by using overall species
    balances
28 P_Bz = F_Bz -1*(ex_r) ;// Benzene in P ,by benzene
    balance - [mol]
29 P_H2 = F_H2 + -3*(ex_r) ;// H2 in P ,by H2 balance -
    [mol]
30 P_C6H12 = 0 + 1*(ex_r) ;// Cyclohexane in P ,by
    cyclohexane balance - [mol]
31 P = P_Bz + P_H2 + P_C6H12 ;// Total Product - [ mol]
32
33 // Use single pass conversion information to get
    recycle stream(R)
34 R = ((-(-ex_r))/(sp_con) - F_Bz)/(R_Bz/100) ;//
    Recycle stream - [mol]
35 R_by_F = R/F ;// Ratio of R to F
36
37 printf('Ratio of R to F is %.2f .\n',R_by_F);
38
39 TLV = (P_Bz/P)*(1/TLV_Bz) + (P_H2/P)*(1/TLV_H2) + (
    P_C6H12/P)*(1/TLV_C6H12) ;// TLV (environmental
    index)
40
41 printf('\n TLV (environmental index) is %.3f .\n',
    TLV);

```

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

```

1 clear;
2 clc;
3 //Page No.359
4 // Example 12.3
5 printf('Example 12.3\n\n');
6 // Solution fig.E12.3a and fig.E12.3b

```

```

7
8 // Given
9 // Main reaction - C6H12O6(d-glucose) --> C6H12O6(d-
    fructose)
10
11 RR = 8.33 ;// Recycle ratio
12 F = 100 ;// Overall feed/basis - [lb]
13 F_g = 0.40 ;// Fraction of glucose in overall feed
14 F_w = 0.60 ;// Fraction of water in overall feed
15 F_dash_f = 0.04 ; // Fraction of fructose in feed to
    reactor
16 P = F ;// By overall balance -[lb]
17 R = P/RR ;// Recycle stream - [lb]
18 P_w = (F_w * F)/ P ;// Fraction of water in product(
    P), by overall water balance
19 R_w = P_w ;//Fraction of water in recycle (R), since
    both R and P has same composition
20
21 // Mixing point 1
22 F_dash = F +R ;// Feed to reactor ,by total balance
    -[lb]
23 R_f = (F_dash*F_dash_f)/R ;// Fraction of fructose
    in recycle stream
24 R_g = 1 - (R_f + R_w) ;// Fraction of glucose in
    recycle stream
25 F_dash_g = (F*F_g + R*R_g)/F_dash ;// Fraction of
    glucose i feed to reactor
26
27 // Make glucose balance in reactor to get fraction
    of conversion (f_con)
28 f_con = ((F_dash*F_dash_g) - (R + P)*R_g)/(F_dash*
    F_dash_g) ;// Fraction of conversion of glucose
    in reactor
29
30 printf('Fraction of conversion of glucose in reactor
    is %.2f .\n',f_con);

```

Scilab code Exa 12.4 A Bioreactor with Recycle

```
1 clear ;
2 clc;
3 //Page No.362
4 // Example 12.4
5 printf('Example 12.4\n\n');
6 // Solution fig.E12.4
7
8 // Given
9 F = 100 ;// Overall feed/basis – [kg]
10 F_com = 0.10 ;// Mass fraction of component in fresh
    feed
11 F_w = 0.90 ;// Mass fraction of water in fresh feed
12 P_w = 0.10 ;// Mass fraction of water in product
13 P_com = 0.90 ;//Mass fraction of component in
    product
14 F_dash_com = 0.03 ;//Mass fraction of component in
    feed to reactor
15 W_w = 1 ;// Mass fraction of water in W(waste)
16 C_con = .40 ;// Fraction of conversion of component
    in reactor
17
18 // By analysis DOF is zero
19 // Take overall process as system
20 P = F_com*F/P_com ;//By component balance– Product –
    [kg]
21 W = F - P ;// By overall balance – waste(W)– [kg]
22
23 //Take reactor plus product recovery unit as system
24 // Use Eqn. 10.1 for component balance
25 R_w = (F*F_com - F*F_com*C_con)/C_con ;// Mass of
    component in recycle(R) – [kg]
26
```

```

27 // Take mixer a system
28 F_dash = ( F*F_com + R_w )/F_dash_com ;// By
    component balance – feed to reactor(F') –[kg]
29 R = F_dash - F ;// Recycle(R) – By total balance –[
    kg]
30 w = R_w/R ;// Mass fraction of component in recycle(R
    )
31
32 printf('Recycle(R) stream– %.0f kg \n',R);
33 printf(' Mass fraction of component in recycle(R)–
    %.4f .\n',w);

```

Scilab code Exa 12.5 Bypass Calculation

```

1 clear ;
2 clc;
3 //Page No.367
4 // Example 12.5
5 printf('Example 12.5\n\n');
6 // Solution fig.E12.5
7
8 // Given
9 F = 100 ;// Overall feed/basis – [kg]
10 F_n_C5H12 = 0.80 ;// Fraction of n_C5H12 in overall
    feed
11 F_i_C5H12 = 0.20 ;// Fraction of i_C5H12 in overall
    feed
12 S_i_C5H12 = 1 ;// Fraction of i_C5H12 in isopentane
    stream
13 P_n_C5H12 = .90 ;// Fraction of n_C5H12 in overall
    product
14 P_i_C5H12 = .10 ;// Fraction of i_C5H12 in overall
    product
15
16 // Overall Balances

```

```

17 P = (F*F_n_C5H12)/P_n_C5H12 ;//Product Material
    Balance of n.C5H12 -[kg]
18 S = F - P ;// Isopentane stream (S) from overall
    material balance - [kg]
19
20 // Balance around isopentane tower
21 // Let x be kg of butane free gas going to
    isopentane tower , y be the n-C5H12 stream
    leaving the isopentane tower
22 // Solve following Equations by Matrix method
23 // x = S + y - By Total material balance
24 // x*F_n_C5H12 = y
25 a = [1 -1;F_n_C5H12 -1] ;// Matrix of coefficients
    of unknown
26 b = [S;0] ;// Matrix of constants
27 x = a\b ;// Matrix of solutions , x(1) = x , x(2) = y
28
29 xf = x(1)/F ;// Fraction of butane-free gas going to
    isopentane tower
30
31 printf('Fraction of butane-free gas going to
    isopentane tower is %.3f .\n',xf);

```

Scilab code Exa 12.6 Purge

```

1 clear;
2 clc;
3 //Page No.369
4 // Example 12.6
5 printf('Example 12.6\n\n');
6 // Solution fig.E12.6
7
8 // Given
9 F = 100 ;// Overall feed/basis - [mole]
10 F_H2 = 0.673 ;// Mole fraction of H2 in overall feed

```

```

11 F_CO = 0.325 ;// Mole fraction of i_C5H12in overall
    feed
12 F_CH4 = .002 ;// Mole fraction of CH4 in overall
    feed
13 E_CH3OH = 1 ;// Mole fraction of CH3OH in Exit(E)
14 // Let x , y and z be Mole fraction of H2,CO and CH4
    respectively in recycle(R) and purge(P)
15 z = .032 ;
16 CO_con = .18 ;// Fraction of conversion of CO in
    reactor
17
18 // Following eqn. are obtained by Material balances
19 // x + y + z = 1 eqn.(a)
20 // F_H2*F + F_CH4*F*2 = E*2 + P*(x + 2z) - By H2
    balance eqn.(b)
21 // F_CO*F + F_CH4*F = E + P*(y + z) -By C balance
    eqn.(c)
22 // F_CO*F = E + P*y - By O balance eqn.(d)
23 // F_CO*F + Ry - Ry - Py = (F_CO*F + Ry)*CO_con -
    By CO balance eqn.(e)
24
25 //By using eqn.(c) and (d)
26 P = F_CH4*F/z ;// Purge stream - [mole]
27
28 // Using eqn.(a) , (b) and (c)
29 x_plus_y = 1 - z ;// x + y
30 E = (F_H2*F + F_CO*F + 3*F_CH4*F - P*(x_plus_y + 3*z
    ))/3 ;// Exit stream - [mole]
31
32 // By using eqn. (d)
33 y = ( F_CO*F - E )/P ;// Mole fraction of CO
34
35 // By using eqn. (a)
36 x = 1 - z - y ;// Mole fraction of H2
37
38 // Lastly by using eqn.(e)
39 R = ( F_CO*F - P*y - F_CO*F*CO_con )/(y*CO_con) ;//
    Recycle steam - [mole]

```

```

40
41 printf('Moles of recycle(R) per mole of feed(F) -
          %.4f \n',R/F);
42 printf(' Moles of CH3OH(E) per mole of feed(F) -
          %.4f \n',E/F);
43 printf(' Moles of Purge(P) per mole of feed(F) -
          %.4f \n',P/F);
44 printf('\n Composition of Purge \n');
45 printf(' Component Mole fraction \n
          ');
46 printf(' H2 %.3f \n',x);
47 printf(' CO %.3f \n',y);
48 printf(' CH4 %.3f \n',z);

```

Chapter 13

Ideal Gases

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

```
1 clear;
2 clc;
3 // Example 13.1
4 printf('Example 13.1\n\n');
5 //Page No. 404
6 // Solution
7
8 m_CO2 = 40 ;// Mass of CO2-[kg]
9 mol_wt_CO2 = 44 ;// Molecular mass of 1kmol CO2 -[kg
10 mol_V = 22.42 ;// Molar of ideal gas at standard
11 condition -[cubic metre/kg mol]
12 V_CO2 = (m_CO2 * mol_V)/(mol_wt_CO2);// volume of
13 CO2-[cubic metre]
14 printf('Volume occupied by 40 kg CO2 at standard
15 condition is %.1f cubic metre.',V_CO2);
```

Scilab code Exa 13.2 Calculation of R using the Standard Conditions

```

1 clear ;
2 clc;
3 // Example 13.2
4 printf('Example 13.2\n\n');
5 //Page No. 405
6 // Solution
7
8 p =1 ;// Pressure -[atm]
9 V = 22415 ;// Molar valume -[cubic centimetre/g mol]
10 T = 273.15 ;// Temperature-[K]
11 R = (p*V/T);// Universal gas constant -[(cubic
    centimetre.atm)/(K.g mol)]
12 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

```

1 clear ;
2 clc;
3 // Example 13.3
4 printf('Example 13.3\n\n');
5 // Page No.406
6 // Solution
7
8 m_CO2 = 88 ;// Mass of CO2-[lb]
9 mol_wt_CO2 = 44 ;// Molecular mass of 1 lb mol
    CO2 -[lb]
10 mol_V = 359 ; // Molar volume-[cubic feet]
11
12 // State 1-standard condition
13 P1 = 33.91 ; // Pressure -[ft of water]
14 T1 = 273 ;// Temperature-[K]
15
16 // State 2
17 P2 = 32.2 ;// Pressure -[ft of water]

```

```

18 Tc = 15 ;// Temperature-[degree C]
19 T2 = Tc+273 ;// Temperature-[K]
20
21 // Use eqn. 13.2 to get final volume
22 V1 = (m_CO2 * mol_V) / (mol_wt_CO2);
23 V2 = (V1 * T2 * P1) / (T1 * P2);
24 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

```

1 clear ;
2 clc;
3 // Example 13.4
4 printf('Example 13.4\n\n');
5 //Page No. 408
6 // Solution
7
8 mol_wt_N2 = 28 ;// Molecular mass of 1 kg mol N2 -[
kg]
9 mol_V = 22.42 ;// Molar of ideal gas at standard
condition-[cubic metre/kg mol]
10 Tc = 27 ;// Temperature-[degree C]
11 T = Tc + 273 ;//Temperature-[K]
12 P = 100 ;//Pressure-[kPa]
13
14 //Standard condition
15 Ps = 101.3 ;// Pressure -[kPa]
16 Ts = 273 ;//Temperature-[K]
17
18 V = (T * Ps * mol_V)/(Ts * P) ;// Volume occupied
by N2-[cubic metre]
19 D_N2 = mol_wt_N2/V ;// Density of N2 at given
condition-[kg/cubic metre]
20 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

```
1 clear ;
2 clc;
3 // Example 13.5
4 printf('Example 13.5\n\n');
5 //Page No. 409
6 // Solution
7
8 mol_wt_N2 = 28 ;// Molecular mass of 1 lb mol N2 -[
    lb]
9 mol_wt_air = 29 ;// Molecular mass of 1 lb mol air
    -[lb]
10 mol_V = 359 ;// Molar volume of ideal gas-[cubic
    feet]
11 //Given condition
12 Tf = 80 ;// Temperature-[degree F]
13 T = Tf + 460 ;//Temperature-[degree Rankine]
14 P = 745 ;//Pressure-[mm of Hg]
15
16 //Standard condition
17 Ps = 760 ;// Pressure -[mm of Hg]
18 Ts = 492 ;//Temperature-[degree Rankine]
19
20 D_air = (Ts * P * mol_wt_air)/(T * Ps * mol_V) ;
    // Density of air at given condition -[lb/cubic
    feet]
21 D_N2 = (Ts * P * mol_wt_N2)/(T * Ps * mol_V) ;//
    Density of N2 at given condition -[lb/cubic feet]
22 sg_N2 = D_N2/D_air ;// Specific gravity of N2
    compared to air at given condition
23 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 Gas

```
1 clear ;
2 clc ;
3 // Example 13.6
4 printf('Example 13.6\n\n');
5 //Page No. 414
6 // Solution
7
8 F_gas = 1 ;// Flue gas [kg mol]
9 mf_CO2 = 14/100 ;// [mol fraction]
10 mf_O2 = 6/100 ;// [mol fraction]
11 mf_N2 = 80/100 ;// [mol fraction]
12 P = 765 ;//Pressure-[mm of Hg]
13 T = 400 ;// Temperature-[degree F]
14 p_CO2 = P * mf_CO2 ;// Partial pressure of CO2-[mm
    of Hg]
15 p_O2 = P * mf_O2 ;// Partial pressure of O2-[mm of
    Hg]
16 p_N2 = P * mf_N2 ;// Partial pressure of N2-[mm of
    Hg]
17
18 printf(' Component                pi(Partial pressure-[
    mm of Hg]) \n');
19 printf(' CO2                        %.1f mm of Hg\n ',
    p_CO2);
20 printf(' O2                          %.1f mm of Hg\n ',
    p_O2);
21 printf(' N2                          %.1f mm of Hg\n ',
    p_N2);
```

Scilab code Exa 13.7 Material Balance for a Process Involving Combustion

```
1 clear ;
2 clc;
3 // Example 13.7
4 printf('Example 13.7\n\n');
5 //Page no. 416
6 // Solution fig E13.7
7
8 G = 100 ;// Basis: Pyrolysis Gas-[lb mol]
9 ub_CO = 10/100 ;// fraction of CO left unburnt
10 ex_air = 40/100 ;// fraction of excess air
11 m_vol = 359 ;// molar volume of gas at std. cond.-[
    cubic feet]
12 Ts = 492 ;// Standard temperature -[degree Rankine]
13 Ps = 29.92 ;//Standard pressure -[in. Hg]
14
15 // Analysis of entering gas of entering gas
16 Tf1 = 90 ;// Temperature of gas-[degree F]
17 T_gas = Tf1 + 460 ;//Temperature of gas-[degree
    Rankine]
18 P_gas = 35 ;//Pressure -[in. Hg]
19 CO2 = 6.4/100 ;// mol fraction of CO2
20 O2 = 0.1/100 ;// mol fraction of O2
21 CO = 39/100 ;// mol fraction of CO
22 H2 = 51.8/100 ;// mol fraction of H2
23 CH4 = 0.6/100 ;// mol fraction of CH4
24 N2 = 2.1/100 ;// mol fraction of N2
25
26 // Analysis of entering air
27 Tf2 = 70 ;// Temperature of air -[degree F]
28 T_air = Tf2 + 460 ;//Temperature of air -[degree
    Rankine]
29 P_air = 29.4 ;//Pressure of air [in. Hg]
30 f_N2 = 79/100 ;// mol fraction of N2
31 f_O2 = 21/100 ;// mol fraction of O2
32
```

```

33 // Get O2 required for combustion of CO,H2 & CH4
    according to the following equation
34 // CO + 1/2O2—>CO2
35 //H2 + 1/2O2—>H2O
36 //CH4 + 2O2—> CO2 + 2H2O
37 O2r_O2 = O2 * G ;// O2 required by O2-[lb mol]
38 O2r_CO = CO * G/2 ;// O2 required by CO-[lb mol]
39 O2r_H2 = H2 * G/2 ;// O2 required by H2-[lb mol]
40 O2r_CH4 = G * CH4 * 2 ;// O2 required by CH4-[lb mol
    ]
41 O2r_total = O2r_O2 + O2r_CO + O2r_H2 + O2r_CH4 ;
    // Total O2 required-[lb mol]
42 ex_O2 = ex_air * O2r_total ;// Excess O2-[lb mol]
43 total_O2 = ex_O2 + O2r_total ;// Total amt of O2 in
    air-[lb mol]
44 total_N2 = total_O2 * (f_N2/f_O2);// Total amt of in
    air-[lb mol]
45 air = total_O2 + total_N2 ;// Total air entering -[
    lb mol]
46
47 // Product analysis
48 P_CO = ub_CO * CO * G ;//Unburnt CO in P-[lb mol]
49 //Element balance of 2N
50 P_N2 = N2 * G + total_N2 ;// N2 in P-[lb mol]
51 //Element balance of C
52 P_CO2 = (CO2 + CO + CH4) * G - 1 * P_CO; //CO2 in
    P-[lb mol]
53 // Element balance of 2H
54 P_H2O = (H2 + 2 * CH4) * G ;// H2 in P-[lb mol]
55 // Element balance of 2O
56 P_O2 = (CO2 + O2 + 0.5 * CO) * G + total_O2 -
    P_CO2-0.5 * (P_H2O + P_CO); // O2 in P-[lb mol]
57 P = P_CO + P_N2 + P_CO2 + P_H2O + P_O2 ;//
    Product-[lb mol]
58 Tf3 = 400 ;// Temperature of product-[degree F]
59 T_prod = Tf3 + 460 ;//Temperature of product-[
    degree Rankine]
60 P_prod = 35 ;// Pressure of product -[in.Hg]

```

```

61 V_gas = (G * m_vol * T_gas * Ps)/(Ts * P_gas);
62 V_air = (air * m_vol * T_air * Ps)/(Ts * P_air);
63 V_prod = (P * m_vol * T_prod * Ps)/(Ts * P_prod);
64 air_ft3 = V_air/V_gas ;//Air supplied per ft^3 of
    gas entered-[cubic feet]
65 P_ft3 = V_prod/V_gas ;//Product gas produced per ft
    ^3 of gas entered-[cubic feet]
66
67 printf(' Air supplied per ft^3 of gas entered %.2f
    cubic feet.\n ',air_ft3);
68 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

```

1 clear ;
2 clc;
3 //Page No. 419
4 // Example 13.8
5 printf('Example 13.8\n\n');
6 // Solution fig E13.8
7
8 T1c = 15 ;// Temperature of F & P -[degree C]
9 T1 = 273 + T1c ;// Temperature of F & P -[K]
10 P1 = 105 ;// Pressure of F & P -[kPa]
11 // F analysis
12 F_CO2 = 1.2/100 ;// Volume fraction
13 F_odr = 98.8/100 ;// Volume fraction
14
15 // P analysis
16 P_CO2 = 3.4/100 ;// Volume fraction
17 P_odr = 96.6/100 ;// Volume fraction
18
19 Tc_CO2 = 7 ;//Temperature CO2 -[degree C]
20 T_CO2 = 273 + Tc_CO2 ;// Temperature CO2 -[K]

```

```

21 P_CO2 = 131 ;// Pressure of CO2 -[kPa]
22 C02 = 0.0917 ;// Volume flow rate of CO2-[cubic
    metre/min]
23 // Convert given volume flow rate of CO2 at
    temperature of F & P
24 nw_CO2 = (C02 * T1 * P_CO2)/(T_CO2 * P1) ;//
    volume flow rate of CO2 at temperature of F & P-[
    cubic metre]
25 // Solve P & F by following eqns. obtained by
    component balance of CO2 and total balance
26 // F(F_odr) = P(P_odr) - others balance
27 // F + nw_CO2 = P - Total balance
28 // Solving by matrix method
29 a = [F_odr -P_odr;1 -1];// Matrix formed by
    coefficients of unknown
30 b = [0;-nw_CO2] ;// Matrix formed by constants
31 x = a\b ;// matrix of solution , x(1) = F;x(2) = P
32 F = x(1) ;//Volume flow rate of entering gas-[cubic
    metre/min]
33 P = x(2) ;//Volume flow rate of product [cubic metre
    /min]
34 printf('Volume flow rate of entering gas is %.2f
    cubic metre/min ',F);

```

Chapter 14

Real Gases Compressibility

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

```
1 clear ;
2 clc;
3 // Example 14.1
4 printf('Example 14.1\n\n');
5 //Page No. 442
6 // Solution
7
8 T1f = 125 ;// Temperature of NH3 -[degree F]
9 T1 = 460 +T1f ;// Temperature NH3 -[degree Rankine]
10 Pg = 292 ;// Pressure of NH3 -[psig]
11 Pa = Pg+14.7 ;// Pressure of NH3 -[psia]
12 R = 10.73 ;// Universal gas constant -[(psia*cubic
    feet)/(lb mol*R)]
13 mw_NH3 = 17 ;// Molecular wt. 1 lb mol NH3-[lb]
14 n = 1/17 ;//[mol]
15 V_tank = 120 ;// Volume of tank-[cubic feet]
16 // Ideal V
17 V_id = (n*R*T1)/Pa ;// Specific volume of NH3
    treating it ideal gas-[cubic feet/lb]
18
```

```

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

```

Scilab code Exa 14.2 Use of Compressibility Factor to Calculate Pressure

```

1 clear ;
2 clc;
3 // Example 14.2
4 printf('Example 14.2\n\n');
5 //Page No. 444
6 // Solution
7
8 //From appendix D
9 Tc = 154.4 ;//[K]
10 Pc1 = 49.7 ;// [atm]
11 Pc = 101.3 * Pc1;// [kPa]
12
13 T_021 = -25 ;// Temperature-[degree C]
14 T_02 = 273+T_021;//Temperature -[K]
15 R = 8.134 ;// gas constant -[(cubic metre * kPa)/(kg
    mol * K)]
16 V_tank = 0.0284 ;// Volume of tank-[cubic metre]

```



```

17 mol_O2 = 32 ;// Kmole wt. of O2-[kg]
18 m_O2 = 3.5 ;// Mass of liquid O2-[kg]
19 V_sp = V_tank * mol_O2/m_O2 ;// Specific molar
    volume-[m^3/kg]
20 Vc = R * Tc/Pc ;// [cubic metre/kg mol]
21 Vr = V_sp/Vc;
22 Tr = T_O2/Tc;
23
24 // Now use Vr and Tr to get Pr from Nelson and Obert
    chart ,Pr = 1.43
25 Pr = 1.43 ;// [kPa]
26 P_O2 = Pr * Pc ;// The pressure in the tank -[kPa]
27 printf('The pressure in the tank is %.2f kPa.',P_O2
    );

```

Scilab code Exa 14.3 Calculation of Properties of Real Gas Mixture

```

1 clear ;
2 clc;
3 // Example 14.3
4 printf('Example 14.3\n\n');
5 //Page No. 448
6 // Solution
7
8 // Given
9 Tc = 100 ;// Temperature -[degree C]
10 T = 273 +Tc ;// Temperature -[K]
11 P = 90 ;// Pressure [atm]
12 R = 82.06 ;// gas constant -[(cubic centimetre * atm)
    /(g mol * K)]
13 Y_CH4 = 20/100 ;// [mole fraction]
14 Y_C2H4 = 30/100 ;// [mole fraction]
15 Y_N2 = 50/100 ;//[mole fraction]
16
17 //Additional information from appendix D

```

```

18 Tc_CH4 = 191 ;// [K]
19 Pc_CH4 = 45.8 ;// [atm]
20 Tc_C2H4 = 283 ;// [K]
21 Pc_C2H4 = 50.5 ;// [atm]
22 Tc_N2 = 126 ;// [K]
23 Pc_N2 = 33.5 ;// [atm]
24
25 //(a)-Ideal gas law
26 V_sp1 = R * T/P ;// Molar volume-[cubic centimetre/g
    mol]
27 printf('(a) The volume per mole of mixture by ideal
    gas law is %.1f cubic centimetre/g mol.\n',V_sp1)
    ;
28
29 //(b)
30 Pc_mix = Pc_CH4 * Y_CH4+Pc_C2H4 * Y_C2H4+Pc_N2 *
    Y_N2;// [atm]
31 Tc_mix = Tc_CH4 * Y_CH4+Tc_C2H4 * Y_C2H4+Tc_N2 *
    Y_N2 ;// [K]
32 Pr_mix = P/Pc_mix;
33 Tr_mix = T/Tc_mix;
34 // With 2 parameters(Pr_mix and Tr_mix) , you can
    find from figure 14.4b that z * Tr_mix = 1.91
35 z = 1.91/Tr_mix;
36 V_sp2 = z * R * T/P ;// Molar volume-[cubic
    centimetre/g mol]
37 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 Waals Equation to Calculate a Temperature

```
1 clear ;
2 clc;
3 // Example 15.1
4 printf('Example 15.1\n\n');
5 //Page No. 464
6 // Solution
7
8 // Given
9 R = 82.06 ;// gas constant -[(cm^3 *atm)/(g mol *K)]
10 a = 9.24 *10^(6) ;//(atm) *(cm^3/g mol)^2
11 b = 90.7 ;// (cm^3)/(g mol)
12 m_C3H8 = 22.7 ;// Mass of propane -[kg]
13 mw_C3H8 = 44 ;// Mol. wt. of 1kmol propane -[kg]
14 V = 0.15 *10^(6) ;// Volume of cylinder -[cm^3]
15 pg = 4790 ;// Gauge pressure -[kPa]
16 P = (pg +101.3)/101.3 ;// Pressure absolute -[atm abs
    ]
17 n = (m_C3H8/mw_C3H8) *10^3 ;// Moles of propane
18 // Get T using Van der Waal's eqn.
19 T = ((P +((n^(2) *a/(V^(2)))))*(V-n *b))/(R *n) ;//
```

```

    Temperature of propane – [K]
20 printf('\nTemperature of propane is %.0f K.',T);

```

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

```

1 clear ;
2 clc ;
3 // Example 15.2
4 printf('Example 15.2\n\n');
5 //Page No. 465
6 // Solution
7
8 // Given
9 R = 10.73 ; // gas constant – [(cubic feet *psia)/(lb
    mol *R)]
10 a = 3.49 * 10^4 ; // (psia) *(cubic feet/lb mol)^2
11 b = 1.45 ; // (cubic feet)/(lb mol)
12 p = 679.7 ; // Pressure – [ psia]
13 n = 1.136 ; // Amount of mole – [lb mol]
14 T = 683 ; // Temperature – [degree R]
15
16 // Get V using Van der Waal's eqn.
17 def(' [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');
18 V=fsolve(b,g) ; // Volume of final solution (volume
    of vessel) [cubic feet]
19
20 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

```
1 clear ;
2 clc;
3 // Example 16.1
4 printf('Example 16.1\n\n');
5 //Page no. 486
6 // Solution Fig E16.1
7
8 // Given
9 Tc = 972 ;//[degree C]
10 T = 273+Tc ;//[K]
11 A = 8.799;
12 B = 1.615 * 10^4;
13 C = 0;
14 mw = 26.98;
15 // Use Antoine eqn. to get vapour pressure at 972
    degree C
16 vP = exp(A-(B/(C+T))) ;// vapour pressure at 972
    degree C-[mm Hg]
17 P = vP * 101.325/760 ;//[kPa]
```

```

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

```

Scilab code Exa 16.2 Interpolating in Steam Tables

```

1 clear ;
2 clc;
3 // Example 16.2
4 printf('Example 16.2\n\n');
5 //Page no. 491
6 // Solution
7
8
9 //(a)
10 // Given
11 // get essential data from steam table
12 Ta = [310,315] ;//Temperature data from steam table-
    [K]
13 pa = [6.230,8.143] ;// Pressure data from steam
    table - [kPa]
14 pfa = interp1n([Ta;pa],312) ;// Pressure at 312 K -
    [kPa]
15 printf('(a) Saturation pressure of water at 312 K is
    %.1f kPa.\n',pfa );
16
17 //(b)
18
19 //For initial condition get specific volume
20 // Double interpolation
21 //first interpolation
22 // at 600 degree F ,data from steam table
23 pb_600 = [90,95] ;// Pressure data - [psia]

```

```

24     vb_600= [6.916,6.547] ;// specific volume data
        - [(cubic feet/lb)]
25     v_600 = interpln([pb_600;vb_600],92);//
        specific volume at 92 psia and 600 degree F
        - [(cubic feet/lb)]
26
27     //at 700 degree F,data from steam table
28     pb_700 = [90,95] ;// Pressure data - [psia]
29     vb_700 = [7.599,7.195];// specific volume data
        - [(cubic feet/lb)]
30     v_700 = interpln([pb_700;vb_700],92) ;//
        specific volume at 92 psia and 700 degree F
        - [(cubic feet/lb)]
31
32     // second interpolation 92 psia ,data from steam
        table
33     Tb_92 = [600,700];// Temperature data from
        steam table - [degree F]
34     vb_92 = [v_600,v_700];// specific volume data
        - [(cubic feet/lb)]
35     v_640_92 = interpln([ Tb_92;vb_92],640);//
        specific volume at 92 psia and 640 degree F
        - [(cubic feet/lb)]
36
37
38
39     //For final condition get specific volume
40     // Double interpolation
41     //first interpolation
42     // at 450 degree F,data from steam table
43     pc_450 = [50,55] ;// Pressure data - [
        psia]
44     vc_450 = [10.69,9.703];// specific volume
        data - [(cubic feet/lb)]
45     v_450 = interpln([pc_450;vc_450],52);//
        specific volume at 52 psia and 450
        degree F - [(cubic feet/lb)]
46

```

```

47 //at 500 degree F,data from steam table
48 pc_500 = [50,55] ; // Pressure data - [psia]
49 vc_500 = [11.30,10.26]; // specific volume data
    - [(cubic feet/lb)]
50 v_500 = interpln([pc_500;vc_500],52); //
    specific volume at 52 psia and 500 degree F
    - [(cubic feet/lb)]
51
52 // second interpolation 52 psia ,data from steam
    table
53 Tc_52 = [450,500]; // Temperature data from
    steam table - [degree F]
54 vc_52 = [v_450,v_500]; // specific volume data
    - [(cubic feet/lb)]
55 v_480_52 = interpln([ Tc_52;vc_52],480) ; //
    specific volume at 52 psia and 480 degree F
    - [(cubic feet/lb)]
56
57 del_v = v_480_52 - v_640_92 ; // Change in specific
    volume - [(cubic feet/lb)]
58 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

```

1 clear ;
2 clc;
3 // Example 16.3
4 printf('Example 16.3\n\n');
5 //Page no. 494
6 // Solution
7
8 //Given
9 T1 = 110 ; // Temperature of chlorobenzene - [degree

```



```

    C]
10 T1F = (9*T1)/(5) + 32 ;// Temperature of
    chlorobenzene - [degree F]
11 P1 = 400 ;//Pressure of chlorobenzene - [mm of Hg]
12 P1_psia = P1*14.7/760 ;//Pressure of chlorobenzene -
    [psia]
13 T2 = 205 ;// Temperature of chlorobenzene - [degree
    C]
14 T2F = (9*T2)/(5) + 32 ;// Temperature of
    chlorobenzene - [degree F]
15 P2 = 5 ;//Pressure of chlorobenzene - [atm]
16 P2_psia = P2*14.7 ;//Pressure of chlorobenzene - [
    psia]
17
18 // Data from steam table
19 x1 = [.9487,3.72,11.525,29.8,67,247,680,1543,3094];
20 y1 = [100,150,200,250,300,400,500,600,700];
21
22 x2 = [P1_psia,P2_psia];
23 y2 = [T1F,T2F];
24
25 // Cox chart using given and steam table data
26 plot2d("ln",x1,y1,5);
27 //plot2d("ln",x1,y1)
28 xgrid(3);
29 //plot2d("ln",x2,y2)
30 plot2d("ln",x2,y2,2);
31 xgrid(3);
32 legend("Water", "Chlorobenzene");
33 plot2d("ln",x1,y1,-9);
34 plot2d("ln",x2,y2,-9);
35 title('Figure E16.3 Cox chart for the problem');
36 xlabel('Vapour Pressure , psia (log(10) scale)');
37 ylabel('Temperature , degree F(special scale)');
38
39 // Estimate vapour pressure of chlorobenzene from
    cox chart prepared, it is
40 vp1 = 150 ;// vapour pressure of chlorobenzene from

```

```

    cox chart prepared at 245 degree C
41 vp2 = 700 ;// vapour pressure of chlorobenzene from
    cox chart prepared at 359 degree C
42
43 printf('Temperature           Estimated vapour
    pressure of chlorobenzene from cox chart\n');
44 printf('\n 245 degree C           %i psia\n',vp1);
45 printf(' 359 degree C           %i psia ',vp2);

```

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

```

1 clear ;
2 clc;
3 // Example 16.4
4 printf('Example 16.4\n\n');
5 //Page no. 495
6 // Solution
7
8 // Given
9 OP_Et = 400 ;//OSHA PEL of ethyl acetate -[ppm by
    volume]
10 OP_Mek = 200 ;//OSHA PEL of Methyl ethyl ketone [
    ppm by volume]
11 OP_Nba = 1.3 ;//OSHA PEL of n-butyl acetate [ppm by
    volume]
12
13 vp_Et = 96.9 ;// Vapour pressure of ethyl acetate
    obtained from CD-[mm of Hg]
14 vp_Mek = 94.8 ;// Vapour pressure of Methyl ethyl
    ketone obtained from CD-[mm of Hg]
15 vp_Nba = 20 ;// Vapour pressure of n-butyl acetate
    obtained from Perry-[mm of Hg]
16
17 // Combined hazard criterion

```

```
18 Chz_Et = vp_Et/OP_Et ;// Combined hazard criterion
    of ethyl acetate
19 Chz_Mek = vp_Mek/OP_Mek ;// Combined hazard
    criterion of Methyl ethyl ketone
20 Chz_Nba = vp_Nba/OP_Nba ;// Combined hazard
    criterion of n-butyl acetate
21
22 printf('\nCombined hazard criterion of solvents in
    increasing order are :\n');
23 printf('\nEthyl acetate :                %.2 f ',Chz_Et)
    ;
24 printf('\nMethyl ethyl ketone :         %.2 f ',Chz_Mek
    );
25 printf('\nn-butyl acetate :             %.2 f ',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

```
1 clear ;
2 clc;
3 // Example 17.1
4 printf('Example 17.1\n');
5 // Page no. 511
6 // Solution
7
8 // Basis : F = 1 mol
9 F = 1 ; //H2C2O4- [mol]
10 ex_O2 = 248 ; //Excess air- [%]
11 f_C = 65/100 ; // Fraction of Carbon which convert to
    CO2
12 P = 101.3 ; // Atmospheric pressure -[kPa]
13
14 // H2C2O4 + 0.5*O2-->2*CO2 + H2O
15 // H2C2O4 -->2*CO + H2O + 0.5*O2
```

```

16 O2_req = F*0.5 ;// O2 required by the above reaction
    -[mol]
17 O2_in = (1 + ex_O2*F/100)*0.5 ;// Mol. of O2
    entering
18
19 // Use Elemental balance moles of species in output
20 n_CO2 = f_C*2 ;// [mol]
21 n_H2O = (2*F)/2 ;// From 2H balance-[mol]
22 n_N2 = ((O2_in*0.79)/(0.21)) ;// From 2N balance-[
    mol]
23 n_CO = 2-n_CO2 ;// From C balance-[mol]
24 n_O2 = ((4 + O2_in*2)-(n_H2O + n_CO + 2*n_CO2))/2 ;
    // From O2 balance-[mol]
25 total_mol = n_CO2 + n_H2O + n_N2 + n_CO + n_O2 ;//
    Total moles in output stream-[mol]
26 y_H2O = n_H2O/total_mol ;// Mole fraction of H2O
27 pp_H2O = y_H2O*P ;// Partial pressure of H2O-[kPa]
28
29 printf('\nPartial pressure of H2O %.2f kPa.',pp_H2O)
    ;
30 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

```

1 clear ;
2 clc ;
3 // Example 17.2
4 printf('Example 17.2\n');
5 //Page no. 517
6 // Solution Fig E17.2b
7
8 gas = 1 ;// Entering gas-[g mol]

```

```

9 T = 26 ;// Temperature (for isothermal process)-[
    degree C]
10 // From fig. its clear that at 26 C saturation
    pressure is at point A
11 // Get vapour pressure of benzene from Perry
    handbook or CD,it is
12 vp = 99.7 ;// vapour pressure of benzene at 26 C-[
    mm of Hg]
13
14 // Analysis of entering gas
15 f_C6H6 = 0.018 ;// Mol fraction of benzene
16 f_air = 0.982 ;// Mol fraction of air
17 mol_C6H6 = 0.018*gas ;// Moles of benzene-[g mol]
18 mol_air = 0.982*gas ;// Moles of air-[g mol]
19
20 // Analysis of exit gas
21 C6H6_rec = 95/100 ;// Fraction of benzene recovered
22 C6H6_out = 1-C6H6_rec ;//Fraction of benzene in
    exit stream
23 C6H6_out = mol_C6H6*C6H6_out ;//Moles of benzene in
    exit stream-[g mol]
24 air_out = mol_air ;//Moles of air in exit stream-[g
    mol]
25 total_mol = C6H6_out+air_out ;// Total moles in exit
    stream
26 y_C6H6_out = C6H6_out/total_mol ;// Mole fraction of
    benzene in exit
27 P = vp/y_C6H6_out ;// Pressure total of exit
28
29 printf('\n Pressure total at exit of compressor %.2e
    mm of Hg. ',P);

```

Scilab code Exa 17.3 Smokestack Emissions and Pollution

```
1 clear ;
```

```

2  clc;
3  // Example 17.3
4  printf('Example 17.3\n');
5  // Page no. 519
6  // Solution Fig E17.3b
7
8  // Given
9  // coal analysis from handbook
10 ex_air = .4 ;// Fraction of excess air required
11 w_C = 12 ;// Mol. wt. of C-[g]
12 mol_C = 71/w_C ;//[kg mol]
13 w_H2 = 2.016 ;// Mol. wt. of H2 - [g]
14 mol_H2 = 5.6/w_H2;
15 air_O2 = 0.21;// Fraction of O2 in air
16 air_N2 = 0.79;// Fraction of N2 in air
17
18 // Natural Gas
19 // Basis = 1 kg mol C
20 // CH4 + 2O2 --> CO2 + 2H2O .... Eqn. (a)
21 CO2_1 = 1 ;// By Eqn. (a) CO2 produced -[kg mol]
22 H2O_1 = 2 ;// By Eqn. (a) H2O produced -[kg mol]
23 Req_O2_1 = 2 ;// By Eqn. (a) -[kg mol]
24 ex_O2_1 = Req_O2_1*ex_air ;// Excess O2 required -[
    kg mol]
25 O2_1 = Req_O2_1 + ex_O2_1 ;// Total O2 required - [
    kg mol]
26 N2_1 = O2_1*(air_N2/air_O2) ;//Total N2 required - [
    kg mol]
27 Total_1 = CO2_1 + H2O_1 + N2_1 + ex_O2_1 ;// Total
    gas produced- [kg mol]
28
29 // Coal
30 // C + O2 --> CO2 ..eqn (b)
31 // H2 + 1/2(O2) --> H2O.... eqn (c)
32 CO2_2 = 1 ;// By Eqn. (a) CO2 produced -[kg mol]
33 H2O_2 = mol_H2/mol_C ;// By Eqn. (a) H2O produced -[
    kg mol]
34 Req_O2_2 = 1 + (mol_H2/mol_C)*(1/2) ;// By Eqn. (b)

```

```

    and (c) -[kg mol]
35 ex_O2_2 = Req_O2_2*ex_air ;// Excess O2 required -[
    kg mol]
36 O2_2 = Req_O2_2 + ex_O2_2; // Total O2 required - [
    kg mol]
37 N2_2 = O2_2*(air_N2/air_O2); //Total N2 required - [
    kg mol]
38 Total_2 = CO2_2 + H2O_2 + N2_2 + ex_O2_2 ;// Total
    gas produced- [kg mol]
39
40 // Let P (total pressure) = 100 kPa
41 P = 100 ;// Total pressure -[kPa]
42 p1 = P*(H2O_1/Total_1) ;// Partial pressure of water
    vapour in natural gas - [kPa]
43 Eq_T1 = 52.5 ;// Equivalent temperature -[degree C]
44 p2 = P*(H2O_2/Total_2) ;// Partial pressure of water
    vapour in coal - [kPa]
45 Eq_T2 = 35 ;// Equivalent temperature -[degree C]
46 printf('
                                Natural gas
                                Coal\n')
47 printf('
    _____
    _____\n')
48 printf('Partial pressure:
                                %.1f kPa
                                %.1f kPa\n',p1,p2 ) ;
49 printf('Equivalent temperature:
                                %.1f C
                                %.1f C\n',Eq_T1,
    Eq_T2 );

```

Scilab code Exa 17.4 Material Balance involving Condensation

```

1 clear ;
2 clc;
3 // Example 17.4
4 printf('Example 17.4\n\n');

```



```

5 //Page no. 522
6 // Solution Fig E17.4
7
8 F = 30 ;// Volume of initial gas-[m^3]
9 P_F = 98.6 ;// Pressure of gas-[kPa]
10 T_F = 273+100 ;// Temperature of gas-[K]
11 P_p = 109 ;//[kPa]
12 T_p = 14+273 ;// Temperature of gas-[K]
13 R = 8.314 ;// [(kPa*m^3)/(k mol*K)]
14 // Additional condition
15 vpW_30 = 4.24 ;//Vapour pressure-[kPa]
16 vpW_14 = 1.60 ;//Vapour pressure-[kPa]
17 n_F = (P_F*F)/(R*T_F) ;// Number of moles in F
18
19 // Material balance to calculate P & W
20 P = (n_F*((P_F-vpW_30)/P_F))/((P_p-vpW_14)/P_p) ;//
    P from mat. bal. of air-[kg mol]
21 W = (n_F*(vpW_30/P_F))- P*(vpW_14/P_p); // W from
    mat. bal. of water-[kg mol]
22 iW = n_F*(vpW_30/P_F) ;// Initial amount of water -[
    kg mol]
23 fr_con = W/iW ;//Fraction of water condensed
24
25 printf('\n Fraction of water condensed %.3f.',
    fr_con);

```

Scilab code Exa 17.5 Vaporization to Saturate Dry air

```

1 clear;
2 clc;
3 // Example 17.5
4 printf('Example 17.5\n');
5 //Page no. 527
6 // Solution Fig E17.5
7

```

```

8 P = 100 ;// Pressure of air -[kPa]
9 T = 20 + 273 ;// Temperature of air -[K]
10 R = 8.314 ;// [(kPa*m^3)/(k mol*K)]
11 EOH = 6 ;// Amount of ethyl alcohol to evaporate -[
    kg]
12 mw_EOH = 46.07 ;// Mol.wt. of 1 k mol ethyl alcohol
    -[kg]
13 // Additional data needed
14 vp_EOH = 5.93 ;// Partial pressure of alcohol at 20
    C-[kPa]
15 vp_air = P-vp_EOH ;// Partial pressure of air at 20
    C-[kPa]
16 n_EOH = EOH/mw_EOH ;//Moles of ethyl alcohol -[kg
    mol]
17 n_air = (n_EOH*vp_air)/vp_EOH ;// Moles of air -[kg
    mol]
18 V_air = n_air*R*T/P ;// Volume of air required
19
20 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

```

1 clear ;
2 clc;
3 // Example 17.6
4 printf('Example 17.6\n\n');
5 //Page no. 529
6 // Solution
7
8 P = 760 ;// Pressure -[ mm of Hg]
9 // Get vapour pressure of n-heptane from Perry , 40
    mm of Hg

```

```
10 vp = 40 ;// vapour pressure of n-heptane-[mm of Hg]
11
12 // Use the 2nd relation given in problem to find K
13 K = 10^((log10(vp/P)-0.16)/1.25) ;
14
15 // Get t using the 1st relation in the question
16 // For t_half
17 x = 0.5 ;// mole fraction after t_half
18 x0 = 1 ;// initial mole fraction
19 t_half = (log(x/x0))/(-K);// Time required to reduce
    the concentration to one-half-[min]
20 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

```
1 clear ;
2 clc;
3 // Example 18.1
4 printf('Example 18.1\n\n');
5 //Page no.539
6 // Solution
7
8 V = 1 ;// Volume of water vapour-[cubic metre]
9 rel_h = 43 ;// relative humidity -[%]
10 vp_H2O = 1.61 ;// vapour pressure of water at 94 F-[
    in. of Hg]
11 P_H2O = vp_H2O*(rel_h/100) ;// Pressure of water
    vapour in air-[in. of Hg]
12 P = 29.92 ;// [in of Hg]
13 T = 94+460 ;// Temperature -[Rankine]
14 Ts = 492 ;//Temperature std. -[Rankine]
```

```

15 mw_H2O = 18 ;// molecular mass of water -[lb]
16 H2O = (5280^3*Ts*P_H2O*mw_H2O)/(T*P*359) ;//mass of
    H2O-[lb]
17 // The dew point is temperature at which water
    vapour in air first condense ,i.e at realative
    humidity 100 %, therefore
18 psat_H2O = P_H2O ;// Saturation pressure of H2O -[in
    . of Hg]
19
20 printf('\nSaturation pressure of H2O %.3f in. of Hg\
    n',psat_H2O);
21 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

```

1 clear ;
2 clc;
3 // Example 18.2
4 printf('Example 18.2\n\n');
5 //Page no. 541
6 // Solution
7
8 // Data from steam table
9 psat_H2O = 31.8 ;// Saturation pressure -[mm of Hg]
10
11 //(c)
12 H = .0055 ;// Humidity
13 mw_H2O = 18 ;// Molecular wt. of water-[lb]
14 mw_air = 29 ;// Molecular wt. of air -[lb]
15 P = 750 ;// Pressure total -[mm of Hg]
16 p_H2O = ((H*mw_air*P)/mw_H2O)/(1+(H*mw_air/mw_H2O))
    ;// Partial pressure of water vapour in air-[mm
    of Hg]

```

```

17
18 //(a)
19 rel_H = (p_H20/psat_H20)*100 ;// relative humidity
    -[%]
20
21 //(b)
22 mol_H = (p_H20)/(P-p_H20) ;// Molal humidity
23
24 printf('\n(a)Relative humidity is %.0f%%%\n',rel_H)
    ;
25 printf('\n(b)Molal humidity is %.1e\n',mol_H);
26 printf('\n(c)Partial pressure of water vapour in air
    is %.1f mm of Hg.\n',p_H20);

```

Scilab code Exa 18.3 Dehydration of Moist air

```

1 clear ;
2 clc;
3 // Example 18.3
4 printf('Example 18.3\n\n');
5 //Page No. 544
6 // Solution fig.E18.3
7
8 V_BDA = 1000 ;// Volume of bone dry air(BDA) at 20
    C & 108.0 kPa
9 mol_V = 22.4 ;// Molar volume of gas at standard
    condition -[m^3]
10 T = 20+273 ;// Temperature of BDA-[K]
11 P = 108.0 ;//Pressure of BDA-[kPa]
12 Ts = 273 ;// Standard temperature-[K]
13 Ps = 101.3 ;//Standard pressure-[kPa]
14 W = 0.93 ;// [kg]
15 mw_W = 18 ;// mol. wt. of 1kmol water -[kg]
16 mol_W = W/mw_W ;// amount of water vapour(W)-[kg mol
    ]

```

```

17 mol_BDA = (V_BDA*Ts*P)/(T*Ps*mol_V) ;// amount of
    BDA-[kg mol]
18 p_H2O = (mol_W/(mol_W+mol_BDA))*P ;// Partial
    pressure of H2O-[kPa]
19
20 // Get vapour pressure for water at 15 C , namely
    1.70 kPa
21 psat_H2O = 1.70 ;//vapour pressure for water at 15
    C-[kPa]
22 rel_H = (p_H2O/psat_H2O) ;//Fractional relative
    humidity -[]
23 printf('\n(a) Fractional relative humidity of
    original air was %.3f .\n',rel_H);

```

Scilab code Exa 18.4 Humidification of Air

```

1 clear ;
2 clc;
3 // Example 18.4
4 printf('Example 18.4\n\n');
5 //Page no.545
6 // Solution fig.E18.4
7
8 F = 1000 ;// Volume of entering moist air at 22 C &
    101.0 kPa
9 mol_V = 22.4 ;// Molar volume of gas at standard
    condition -[m^3]
10 T_in = 22+273 ;// Temperature of entering moist air
    -[K]
11 P_in = 101.0 ;//Pressure of entering moist air -[kPa
    ]
12 dp_in = 11+273 ;// Dew point of entering air -[K]
13 Ts = 273 ;// Standard temperature -[K]
14 Ps = 101.3 ;//Standard pressure -[kPa]
15 T_out = 58+273 ;// Temperature of exiting moist air

```

```

    -[K]
16 P_out = 98 ;//Pressure of exiting moist air -[kPa]
17
18 // Additional vapour pressure data
19 psat_in = 1.31 ;//Vapour pressure of entering moist
    air -[kPa]
20 psat_out = 18.14 ;// Vapour pressure of exiting
    moist air -[kPa]
21 pBDA_in = P_in-psat_in ;// Pressure of entering dry
    air - [kPa]
22 pBDA_out = P_out - psat_out ;// Pressure of
    exiting dry air - [kPa]
23
24 mol_F = (F*P_in*Ts)/(Ps*T_in*mol_V) ;// Moles of
    moist air entering -[kg mol]
25
26 //Material Balances to get W
27 mol_P = (mol_F*(pBDA_in/P_in))/(pBDA_out/P_out); //
    BDA balance - [kg mol]
28 mol_W = mol_P-mol_F ;// Total balance -[kg mol]
29
30 // To calculate kg of wet air entering
31 mw_BDA = 29 ;// Mol. wt. of dry air
32 mw_H2O = 18 ;// Mol. wt. of water vapour
33 m_BDA = (mol_F*pBDA_in/P_in)*mw_BDA ;// Mass of dry
    air entering -[kg]
34 m_H2O = (mol_F*psat_in/P_in)*mw_H2O ;// Mass of
    water vapour entering -[kg]
35 wa_in = m_BDA+m_H2O ;//Total wet air entering -[kg]
36 H2O_ad = mol_W*mw_H2O/wa_in ;//Water added to each
    kg of wet air entering the process -[kg]
37
38 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

```
1 clear ;
2 clc;
3 // Example 18.5
4 printf('Example 18.5\n');
5 //Page No.547
6 // Solution fig.E18.5
7
8 // Given data
9 //Basis: F = 29.76 lb mol
10 F = 29.76 ;// amount of entering moist air -[lb mol
    ]
11 F_rh = 90/100 ;// Relative humidity
12 T_in = 100 + 460 ;// Temperature of entering moist
    air -[Rankine]
13 P_in = 29.76 ;//Pressure of entering moist air -[in.
    of Hg]
14 psat_in = 1.93 ;// Saturation pressure from steam
    table -[in. of Hg]
15 T_out = 120 + 460 ;// Temperature of exiting dry air
    -[Rankine]
16 P_out = 131.7 ;//Pressure of exiting dry air -[in.
    of Hg]
17 psat_out = 3.45 ;// Saturation pressure from steam
    table -[in. of Hg]
18 mol_V = 22.4 ;// Molar volume of gas at standard
    condition -[m^3]
19 mw_H2O = 18.02 ;// Mol. wt. of water -[lb]
20 mw_air = 29 ;// Mol. wt. of air -[lb]
21 p_H2O_in = F_rh*psat_in ;// Partial pressure of
    water vapour at inlet --[in. of Hg]
22 p_air_in = P_in-p_H2O_in ;// Partial pressure of air
    at inlet --[in. of Hg]
23
24 // Assume condensation takes place , therefore
    output gas P is saturated ,
25 P_rh = 1; // Relative humidity of output gas
```

```

26 p_H2O_out = P_rh*psat_out ;// Partial pressure of
    water vapour at outlet--[in. of Hg]
27 p_air_out = P_out-p_H2O_out ;// Partial pressure of
    air at outlet--[in. of Hg]
28
29 // Get W and P from balance of air and water
30 P = (p_air_in*F/P_in)/(p_air_out/P_out) ;// From air
    balance-[ lb mol]
31 W = (p_H2O_in*F/P_in)-(P*p_H2O_out/P_out);// From
    water balance -[lb mol]
32 W_ton = (W*mw_H2O*2000)/(p_air_in*mw_air) ;// Moles
    of water condenses per ton dry air-[lb mol]
33 W_m = mw_H2O*W_ton ;// Mass of water condenses per
    ton dry air-[lb]
34 // Since W is positive our assumption(condensation
    takes place ) is right .
35 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);
36 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

```
1 clear ;
2 clc;
3 // Example 19.1
4 printf('Example 19.1\n\n');
5 //Page No. 563
6 // Solution
7
8 // Use phase rule to get degree of freedom(F) = 2-P
9 // (a)
10 N1 = 1;
11 P1 = 1 ;// Number of phases present
12 C1 = 1 ;//Number of components present
13 F1 = 2-P1+C1 ;//Number of degree of freedom
14 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);
```

```

15
16 // (b)
17 N2 = 1;
18 P2 = 2 ;// Number of phases present
19 C2 = 1 ;//Number of components present
20 F2 = 2-P2+C2 ;//Number of degree of freedom
21 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);
22
23 // (c)
24 N3 = 2;
25 P3 = 2 ;// Number of phases present
26 C3 = 2 ;//Number of components present
27 F3 = 2-P3+C3 ;//Number of degree of freedom
28 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);
29
30 // (d)
31 N4 = 2;
32 P4 = 2 ;// Number of phases present
33 C4 = 2 ;//Number of components present
34 F4 = 2-P4+C4 ;//Number of degree of freedom
35 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 can Occur

```

1 clear ;
2 clc;
3 // Example 19.2
4 printf('Example 19.2\n\n');
5 //Page No.564
6 // Solution
7
8 // Use phase rule to get degree of freedom(F) = 2-P
   +C
9 // (a)
10 N1 = 5;
11 P1 = 1; // Number of phases present ,here 1 gas
12 C1 = 3 ;//Number of independent components present ,
   here 3 because 3 elements(C,O and H)
13 F1 = 2-P1+C1 ;//Number of degree of freedom
14 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);
15
16 // (b)
17 N2 = 4;
18 P2 = 4 ;// Number of phases present ,here 3 different
   solid phases and 1 gas phase
19 C2 = 3 ;//Number of components present , here 3
   because 3 elements(Zn,O and C) ,you can also use
   method explained in Appendix L1
20 F2 = 2-P2+C2 ;//Number of degree of freedom
21 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

```

1  clear;
2  clc;
3  // Example 19.3
4  printf('Example 19.3\n\n');
5  //Page No.576
6  // Solution
7
8  P_atm = 1 ;//[atm]
9  P = 760 ;//[mm of Hg]
10 x_1 = 4/100 ;// Mole fraction of hexane in liquid
    phase
11 // Constant A,B and C for Antoine eqn. of n_hexane
12 A1 = 15.8366;
13 B1 = 2697.55 ;
14 C1 = -48.784;
15 // Constant A,B and C for Antoine eqn. of n_octane
16 A2 = 15.9798;
17 B2 = 3127.60 ;
18 C2 = -63.633;
19
20 // Solve for bubble point temperature by eqn.
    obtained by using Antoine equation
21 deff(' [y] = f(T) ', 'y = exp(A1-(B1/(C1+T)))*x_1 +exp(
    A2-(B2/(C2+T)))*(1-x_1) - P');
22 T = fsolve(390,f) ;// Bubble point temperature
23 funcprot(0);
24 printf('Bubble point temperature is %.1f K\n',T);
25
26 // Composition of first vapour
27 // Get vapour pressure of hexane and octane from
    Perry, it is
28 vp_1 = 3114 ;// vapour pressure of hexane-[mm of
    Hg]
29 vp_2 = 661 ;// vapour pressure of octane-[mm of Hg]
30 y_1 = vp_1*x_1/P ;// Mole fraction of hexane in
    vapour phase
31 y_2 = 1- y_1 ;//Mole fraction of octane in vapour
    phase

```

```

32 printf('\n Composition of first vapour.\n ');
33 printf('Component           Mole fraction.\n ');
34 printf('n_hexane             %.3f\n',y_1);
35 printf(' n_octane              %.3f\n',y_2);

```

Scilab code Exa 19.4 Flash Calculation for a Binary Liquid Mixture

```

1 clear ;
2 clc;
3 // Example 19.4
4 printf('Example 19.4\n\n');
5 //Page no. 577
6 // Solution
7
8 // Basis : 100 g solution
9 F = 100 ;// Amount of solution –[g]
10 P_atm = 1 ;//[atm]
11 P = 760 ;// Total pressure –[mm of Hg]
12 wf_hex = 68.6/100 ;//Weight fraction of hexane in
    mixture
13 wf_tol = 31.4/100 ;//Weight fraction of toluene in
    mixture
14 mw_hex = 86.17 ;// Mol.wt. of hexane –[g]
15 mw_tol = 92.13 ;// Mol.wt. of toluene –[g]
16 mol_hex = wf_hex *F/mw_hex ;// moles of hexane –[g
    mol]
17 mol_tol = wf_tol*F/mw_tol ;// moles of toluene –[g
    mol]
18 mol_total = mol_hex + mol_tol ;// Total moles in
    mixture –[g mol]
19 molf_hex = mol_hex/mol_total ;// Mole fraction of
    hexane
20 molf_tol = mol_tol/mol_total ;// Mole fraction of
    toluene
21

```

```

22 // Get vapour pressure of hexane and toluene at 80
    deg. C from Perry, it is
23 vp_hex = 1020 ;// vapour pressure of hexane –[mm of
    Hg]
24 vp_tol = 290 ;// vapour pressure of toluene –[mm of
    Hg]
25 K_hex = vp_hex/P ;// K-value of hexane
26 K_tol = vp_tol/P ;// K-value of toluene
27 rec_K_hex = 1/K_hex ;// Reciprocal of K-value of
    hexane
28 rec_K_tol = 1/K_tol ;// Reciprocal of K-value of
    toluene
29
30 // Let L/F = x, then use eqn. 19.11 to find x(L/F)
31 deff(' [y] = g(x)', 'y = (molf_hex)/(1-x*(1-rec_K_hex)
    ) + (molf_tol)/(1-x*(1-rec_K_tol))-1');
32 x = fsolve(1,g) ;// L/F value
33
34 printf('\n Fraction of liquid(L/F) that will remain
    at equilibrium after vaporization is %.3f.\n ',x)
    ;

```

Scilab code Exa 19.5 Separation of Virus from a Culture

```

1 clear ;
2 clc ;
3 // Example 19.5
4 printf('Example 19.5\n\n');
5 //Page no. 578
6 // Solution
7
8 Vo = 3.0 ;// Initial volume of the solution
    containing the culture and virus –[L]
9 Vp = 0.1 ;// Volume of the polymer solution added to
    the vessel –[L]

```



```
10 Kpc = 100 ;// Partition coefficient for virus(cp/cc)
    between two phases
11
12 //Use virus particle balance to get cp/co, where co
    is initial concentration of virus in solution of
    culture and virus
13 Vc = Vo ;// At equilibrium  $-[L]$ 
14 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
15 Fr_rec = cp_by_co*(Vp/Vo) ;// Fraction of the
    initial virus in the culture phase that is
    recovered in the polymer phase
16
17 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

```
1 clear ;
2 clc;
3 // Example 20.1
4 printf('Example 20.1\n\n');
5 // Page no. 594
6 // Solution Fig E20.1
7
8 // Given
9 p_CO2 = [0,25,50,100,200,400,760] ;// Values of
    partial pressure of CO2 - [mm Hg]
10 y =
    [0,6.69*10^-2,9.24*10^-2,0.108,0.114,0.127,0.137]
    ;// adsorption of CO2 -[g adsorbed / g sieves]
11
12 // R square is a perfect fit
13 plot(p_CO2,y);
14 title('Figure E20.1 The Freundlich and Langmuir
    isotherms coincide for the adsorption of CO2 on 5A
    molecular sieves');
```

```

15 xlabel('P- partial pressure of CO2');
16 ylabel('y');
17 xgrid(1);

```

Scilab code Exa 20.2 Separation of Biochemicals by Solvent Extraction

```

1 clear;
2 clc;
3 // Example 20.2
4 printf('Example 20.2\n\n');
5 //page no. 596
6 // Solution
7
8 //Given
9 G = 1000 ;// Volume of solution - [L]
10 S_ad = 1.56 ;// amount of Steptomycin adsorbed per
    gram resin -[g strep./g resin]
11 cn_S = 6 ;// Concentration of streptomycin solution
    -[g/L]
12 // Assume equilibrium occurs so that total(max)
    amount of streptomycin is adsorbed
13 max_S = cn_S*G ;// Maximum streptomycin adsorbed -[g]
14 //Use streptomycin balance to get amount of resin
    required
15 R = max_S/S_ad ;//Amount of resin required to adsorb
    required amount of streptomycin
16
17 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

```

1  clear ;
2  clc;
3  // Example 20.3
4  printf('Example 20.3\n\n');
5  //page no. 596
6  // Solution
7
8  //Given
9  G = 1000 ;// Volume of solution - [L]
10 x = [19.2,17.2,12.6,8.6,3.4,1.4] ;// concentration
    of solute- [g/L]
11 ac = [0,0.01,0.04,0.08,0.20,0.40] ;// Activated
    charcoal added-[g/1000g sol]
12 // Assume all concentration can be treated as g
    solute/1000 g sol.
13
14 y2 = (x(1)-x(2))/ac(2) ;// -[ g solute/g carbon]
15 y3 = (x(1)-x(3))/ac(3) ;// -[ g solute/g carbon]
16 y4 = (x(1)-x(4))/ac(4) ;// -[ g solute/g carbon]
17 y5 = (x(1)-x(5))/ac(5) ;// -[ g solute/g carbon]
18 y6 = (x(1)-x(6))/ac(6) ;// -[ g solute/g carbon]
19
20 // Use polymath to get Freundlich isotherm to be y=
    37.919*x^(0.583)
21 y = 37.919*x(6)^(0.583) ;//From Freundlich isotherm
22 A_by_G = (x(1)-x(6))/y ;//Minimum mss of activated
    carbon required- [g carbon/1000 g sol.]
23
24 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 How the Path affects the Value of the Work

```
1 clear ;
2 clc;
3 // Example 21.1
4 printf('Example 21.1\n\n');
5 //page no. 616
6 // Solution Fig. E21.1a and E21.1b
7
8 //Given
9 V1 = 0.1 ;// Volume of gas initially -[cubic metres]
10 V2 = 0.2 ;// Volume of gas finally -[cubic metres]
11 T1 = 300 ;// Temperature of gas initially -[K]
12 P1 = 200 ;// Pressure of gas finally -[kPa]
13 R = 8.314 ;// Universal gas constant
14 n = (P1*V1)/(T1*R) ;// Moles of gas taken -[kg mol]
15 //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  $W = \int P.dv$ 
```

```

16
17 //(a)
18 // Isobaric process see fig E21.1b to see the path
    followed
19 W= integrate( '-(P1)', 'V', V1, V2) ;// Work done by gas
    on piston -[kJ]
20 printf('\n (a)Work done by gas on piston for
    isobaric process is %.0f kJ .\n ',W);
21
22 //(b)
23 // Isobaric process see fig E21.1b to see the path
    followed
24 W= integrate( '-(T1*R*n/V)', 'V', V1, V2) ;// Work done
    by gas on piston -[kJ]
25 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 Fluid

```

1 clear ;
2 clc;
3 // Example 21.2
4 printf('Example 21.2\n\n');
5 //page no. 624
6 // Solution
7
8 //Given
9 id = 3 ;// Internal diameter of tube -[cm]
10 Vf = 0.001 ;// Volume flow rate of water in tube -[
    cubic meter/s]
11 rho = 1000 ;// Assumed density of water -[kg/cubic
    meter]
12
13 rad = id/2 ;// Radius of tube -[ cm]

```

```

14 a = 3.14*rad^2 ;// Area of flow of tube -[square
    centimeter]
15 v = Vf*(100)^2/a ;// Velocity of water in tube - [m/
    s]
16 KE = v^2/2 ;// Specific(mass=1kg) kinetic energy of
    water in tube -[J/kg]
17
18 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

```

1 clear ;
2 clc;
3 // Example 21.3
4 printf('Example 21.3\n\n');
5 //page no. 626
6 // Solution
7
8 //Given
9 // Let water level in first reservoir be the
    reference plane
10 h = 40 ;// Difference of water-[ft]
11 g = 32.2 ;// acceleration due to gravity -[ft/square
    second]
12 PE=g*h/(32.2*778.2) ;///// Specific(mass=1kg)
    potential energy of water -[Btu/lbm]
13
14 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

```

1 clear ;
2 clc;
3 // Example 21.4
4 printf('Example 21.4\n\n');
5 //page no. 629
6 // Solution
7
8 //Given
9 //Constant volume process
10 mol_air = 10 ;// Moles of air-[kg mol]
11 T1 = 60+273 ;// Initial temperature of air-[K]
12 T2 = 30+273 ;// final temperature of air-[K]
13 // Additional data needed
14 Cv = 2.1*10^4 ; // Specific heat capacity of air at
    constant volume-[J/(kg mol*C)]
15
16 // Use eqn. 21.6 for del_U
17 del_U = integrate('mol_air*Cv', 'T', T1, T2) ;//Change
    in internal energy-[J]
18 printf('\nChange in internal energy is %.1e J .\n ',
    del_U);

```

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

```

1 clear ;
2 clc;
3 // Example 21.5
4 printf('Example 21.5\n\n');
5 //page no. 629
6 // Solution
7
8 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

```
1 clear ;
2 clc;
3 // Example 21.6
4 printf('Example 21.6\n\n');
5 //page no. 632
6 // Solution
7
8 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

```
1 clear;
2 clc;
3 // Example 21.7
4 printf('Example 21.7\n\n');
5 //page no. 633
6 // Solution
7
8 //Given
9 //Constant pressure process
10 mol_air = 10 ;// Moles of air-[kg mol]
```

```
11 T1 = 60+273 ;// Initial temperature of air-[K]
12 T2 = 30+273 ;// final temperature of air-[K]
13 // Additional data needed
14 Cp = 2.9*10^4 ;// Specific heat capacity of air at
    constant pressure-[J/(kg mol*C)]
15
16 // Use eqn. 21.11 for del_H
17 del_H = integrate('mol_air*Cp', 'T', T1, T2) ;//Change
    in enthalpy-[J]
18 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

```
1 clear;
2 clc;
3 // Example 22.1
4 printf('Example 22.1\n\n');
5 //page no. 651
6 // Solution
7
8 //Assume that properties of water can be used to
   substitute properties of solution
9 // Given
10 V = 1.673 ;// Volume of closed vessel -[cubic metre]
11 m = 1 ;// mass of saturated liquid vaporized -[kg]
12 Pi = 1 ;// Initial pressure -[atm]
13 Ti = 10 ;// Initial temperature -[degree C]
14 Pf = 1 ;// final pressure -[atm]
15 Tf = 100 ;// final temperature -[degree C]
```

```

16
17 // Use steam table to obtain additional information
    at given condition
18 Ui = 35 ;// Initial enthalpy -[kJ/kg]
19 Uf = 2506.0 ;// Final enthalpy -[kJ/kg]
20
21 // Use eqn. 22.2 after modifying it using given
    conditions(W = 0,del_KE = 0 and del_PE = 0 )
22 Q = m*(Uf - Ui) ;// Heat transferred to the vessel
    - [kJ]
23
24 printf('\nHeat transferred to the vessel is %.1f kJ
    .\n ',Q);

```

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

```

1 clear ;
2 clc;
3 // Example 22.2
4 printf('Example 22.2\n\n');
5 //page no. 652
6 // Solution
7
8 // Given
9 T1 = 80 ;// Initial temperature -[degree F]
10 T2 = 40 ;// final temperature -[degree F]
11
12 // Additional data obtained from steam table at
    given temperatures and corresponding vapour
    pressures
13 p1 = 0.5067 ;// Initial saturation pressure -[psia]
14 p2 = 0.1217 ;// Final saturation pressure -[psia]
15 V1 = 0.01607 ;// Initial specific volume - [cubic
    feet/lb]

```

```

16 V2 = 0.01602 ;// Final specific volume - [cubic feet
    /lb]
17 H1 = 48.02 ;// Initial specific enthalpy -[Btu/lb]
18 H2 = 8.05 ;// Final specific enthalpy -[Btu/lb]
19
20 del_P = p2 - p1 ;// Change in pressure -[psia]
21 del_V = V2 - V1 ;// Change in specific volume -[
    cubic feet/lb]
22 del_H = H2 - H1 ;// Change in specific enthalpy -[
    Btu/lb]
23 del_pV = p2*144*V2/778 - p1*144*V1/778 ;// Change in
    pv-[Btu]
24 del_U = del_H - del_pV ;// Change in specific
    internal energy - [Btu/lb]
25 del_E = del_U ;// Change in specific total energy(
    since KE=0,PE=0 and W=0) -[Btu/lb]
26
27 printf('\nChange in pressure is %.3f psia .\n ',
    del_P);
28 printf('\nChange in specific volume is %.5f cubic
    feet/lb (negligible value) .\n ',del_V);
29 printf('\nChange in specific enthalpy is %.2f Btu/lb
    .\n ',del_H);
30 printf('\nChange in specific internal energy is %.2f
    Btu/lb .\n ',del_U);
31 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

```

1 clear ;
2 clc;
3 // Example 22.3
4 printf('Example 22.3\n\n');

```

```

5 //page no. 662
6 // Solution fig.E22.3a
7
8 //Lets take tank to be system
9 // Given
10 T = 600 ; // Temperature of steam -[K]
11 P = 1000 ;// Pressure of steam -[kPa]
12
13 // Additional data for steam obtained from CD
    database at T and P
14 U = 2837.73 ;// Specific internal energy -[kJ/kg]
15 H = 3109.44 ;// Specific enthalpy -[kJ/kg]
16 V = 0.271 ;// Specific volume -[cubic metre/kg]
17
18 // Use eqn. 22.6 to get change in specific internal
    energy ,by simplifying it with following assumption
    :
19 //1. Change in KE and PE of system = 0, therefore
    change in total energy = change in internal
    energy
20 //2. W = 0,work done by or on the system
21 //3. Q = 0 , system is well insulated
22 //4. Change in KE and PE of entering steam = 0
23 //5. H_out = 0, no stream exits the system
24 //6. Ut1 = 0, initially no mass exists in the system
25
26 // By the reduced equation
27 Ut2 = H ;// Internal energy at final temperature -[kJ
    /kg]
28
29 printf('\nThe specific internal energy at final
    temperature is %.2f kJ/kg. \nNow use two
    properties of the steam (P = %i kPa and Ut2 = %.2
    f 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

```
1 clear;
2 clc;
3 // Example 22.4
4 printf('Example 22.4\n\n');
5 //page no. 669
6 // Solution
7
8 // Take milk plus water in tank to be system
9 // Given
10 T1_water = 70 ;// Temperature of entering water -[
    degree C]
11 T2_water = 35 ;// Temperature of exiting water -[
    degree C]
12 T1_milk = 15 ;//Temperature of entering milk -[
    degree C]
13 T2_milk = 25 ;//Temperature of exiting milk -[
    degree C]
14
15 // Get additional data from steam table for water
    and milk, assuming milk to have same properties as
    that of water.
16 H_15 = 62.01 ;//Change in specific internal energy -[
    kJ/kg]
17 H_25 = 103.86 ;//Change in specific internal energy
    -[kJ/kg]
18 H_35 = 146.69 ;//Change in specific internal energy
    -[kJ/kg]
19 H_70 = 293.10 ;//Change in specific internal energy
    -[kJ/kg]
20
21 // Assumptions to simplify Equation 22.8 are:
```

```

22 printf('\nAssumptions to simplify Equation 22.8 are
    :\n');
23 printf('1. Change in KE and PE of system = 0.\n');
24 printf('2. Q = 0 ,because of way we picked the
    system, it is is well insulated.\n');
25 printf('3. W = 0,work done by or on the system.\n');
26
27 //Basis m_milk = 1 kg/min , to directly get the
    answer .
28 m_milk = 1 ;// Mass flow rate of milk-[kg/min]
29 // By applying above assumptions eqn. 22.8 reduces to
    del_H = 0 .Using it get m_water-
30 m_water = (m_milk*(H_15 - H_25))/(H_35 - H_70) ; //
    Mass flow rate of water-[kg/min]
31 m_ratio = m_water/m_milk ;// Mass flow rate of water
    per kg/min of milk-[kg/min]
32 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

```

1 clear ;
2 clc;
3 // Example 22.5
4 printf('Example 22.5\n\n');
5 //page no. 670
6 // Solution
7
8 // Take pipe between initial and final level of
    water
9 // Given
10 h_in = -20 ;// Depth of water below ground-[ft]
11 h_out = 5 ;// Height of water level above ground-[ft
    ]

```



```

12 h = h_out - h_in ;// Total height to which water is
    pumped-[ft]
13 V = 0.50 ;// Volume flow rate of water - [cubic feet
    /s]
14 ef = 100; // Efficiency of pump - [%]
15 g = 32.2; // Acceleration due to gravity -[ft/square
    second]
16 gc = 32.2 ;//[(ft*lbm)/(second square*lbf)]
17
18 M = V * 62.4 ;// mass flow rate - [lbm/s]
19 PE_in = 0 ;// Treating initial water level to be
    reference level
20 PE_out = (M*g*h*1.055)/(gc*778.2) ;// PE of
    discharged water -[lbm*(square feet/square second
    )]
21
22 // Assumptions to simplify Equation 22.8 are:
23 //1. Change in KE = 0.
24 //2. Q = 0 -By given assumption
25 //3. Let us assume that temperature of water is same
    in well and when it is discharged , therefore
    del_H = 0
26 // Reduced equation is W = del_PE , using this:
27 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]
28
29 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

```
1 clear ;
2 clc;
3 // Example 23.1
4 printf('Example 23.1\n\n');
5 // Page no. 686
6 // Solution Fig E23.1
7
8 // Given
9 x_Tl = [90,92,97,100] ;// Temperature of saturated
    liquid- [degree C]
10 x_Tg = [100,102,107,110] ;// Temperature of
    saturated vapour- [degree C]
11 y_Hl = [376.9,385.3,406.3,418.6] ;// Enthalpy change
    of saturated liquid -[kJ/kg]
12 y_Hg = [2256.44,2251.2,2237.9,2229.86] ;// Enthalpy
    change of saturated vapour -[kJ/kg]
13 plot(x_Tl,y_Hl,x_Tg,y_Hg);
14 title('Figure E23.1 Change in enthalpy Vs
    Temperature ');
```

```

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

```

Scilab code Exa 23.2 Comparison of Various Sources to Estimate the Heat of Vapourization

```

1 clear ;
2 clc;
3 // Example 23.2
4 printf('Example 23.2\n\n');
5 //page no. 687
6 // Solution
7
8 // Basis : 1 g mol
9 R = 8.314 * 10^-3 ;// Ideal gas constant -[kJ/(g mol
    * K)]
10 Hv = 30.20 ;// Experimental value of heat of
    vaporization of acetone -[kJ/g]
11
12 // additional needed data for acetone from Appendix
    D
13 T = 329.2 ;// Normal boiling point of acetone - [K]
14 Tc = 508.0 ;// Critical temperature of acetone - [K
    ]
15 Pc = 47.0 ;// Critical presure of acetone -[atm]
16
17 Tbc = T/Tc ;// variable required in etimation
    equations
18 lnPc = log(Pc) ;// variable required in etimation
    equations
19
20 //(a)
21 //Using the Clayperon and Antoine's equation(from
    eqn. 23.2), we get

```

```

22 // del_Hv=(RBT^2)/(C+T)^2 -- eqn. c:
23 //From Appendix G
24 B = 2940.46 ;
25 C = -35.93 ;
26 // using eqn. c
27 del_Hv1 = (R*B*T^2)/((C+T)^2) ;//Heat of
    vapourization -[kJ/g]
28 d1 = (abs(Hv - del_Hv1)*100)/Hv ;// differece of
    experimental and calculated value -[%]
29 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);
30
31 //(b)
32 // Using Chen's equation (from eqn. 23.5)
33 del_Hv2 = R*T*((3.978*Tbc - 3.938 +1.555*lnPc)/(1.07
    - Tbc)) ;//Heat of vapourization -[kJ/g]
34 d2 = (abs(Hv - del_Hv2)*100)/Hv ;// differece of
    experimental and calculated value -[%]
35 printf(' (b) Heat of vapourization of acetone is %.2
    f kJ/g mol. And differece of experimental and
    calculated value is %.1f %% . \n', del_Hv2,d2);
36
37 //(c)
38 // Using Riedel's Equation , from equation 23.6
39 del_Hv3 = 1.093*R*Tc*((Tbc*(lnPc-1))/(0.93-Tbc)) ;//
    Heat of vapourization -[kJ/g]
40 d3 = (abs(Hv - del_Hv3)*100)/Hv ;// differece of
    experimental and calculated value -[%]
41 printf(' (c) Heat of vapourization of acetone is %.2
    f 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

```

1 clear ;
2 clc;
3 // Example 23.3
4 printf('Example 23.3\n\n');
5 // Page no. 693
6 // Solution
7
8 // Given
9 // Heat capacity = 2.675*10^4 + (42.27)Tk -
    1.425*10^-2Tk^2 J/(kg mol K)
10 // First convert heat capacity to Btu/(lb mol*F) to
    get c + dT - eT^2, where
11 c = (2.675*10^4*.4536)/(1055*1.8) ;
12 d = (42.27*.4536)/(1055*1.8) ;
13 e = (1.425*10^-2*.4536)/(1055*1.8) ;
14
15 //Now convert Tk (Temperature in K) to TF (
    temperature in F) to get answer of form x + yT -
    zT^2, where
16 x = c + d*460/1.8 - e*((460/1.8)^2) ;
17 y = d/1.8;
18 z = e/(1.8*1.8) ;
19
20 printf('The required answer is %.2e + (%.2e)T - (%.3
    e) T^2 Btu/(lb mol*F) , where T is in degree F .
    \n',x,y,z);
21
22 // Note answer in textbook seems wrong by order of
    10^-3

```

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

```

1 clear ;
2 clc;

```

```

3 // Example 23.4
4 printf('Example 23.4\n\n');
5 //page no. 694
6 // Solution
7
8
9 //Given
10 // Cp = a + bT +cT^2
11 // we will use the least square procedure defined in
    Appendix M
12 // step 1 : find expression for sum of square of
    residuals: Sr = sum(Cpi - a - bTi - cTi^2)^2
13
14 // step 2 : Now differentiate Sr wrt to each
    coefficient to get 3 equation in 3 unknown
    coefficient , the equations are:
15
16     //n*a + sum(Ti)*b +sum(Ti^2)*c = sum(Cpi)
        ... Eqn.( a)
17     //sum(Ti)*a + sum(Ti^2)*b +sum(Ti^3)*c = sum(
        Cpi*Ti)      ... Eqn.( b)
18     //sum(Ti^2)*a + sum(Ti^3)*b +sum(Ti^4)*c = sum(
        Cpi*Ti^2)   ... Eqn.( c)
19
20 // Take all 18 experimenta data in an array Cp
21 Cpi =
    [39.87,39.85,39.90;45.16,45.23,45.17;50.72,51.03,50.90;56.85,56.8
    ;// Array of Cpi(Heat capacity) values
22 // Take corresponding temperatures in array T
23 Ti =
    [300,300,300;400,400,400;500,500,500;600,600,600;700,700,700;800,8
    ;// array of Ti
24 Ti_sqr =
    [300^2,300^2,300^2;400^2,400^2,400^2;500^2,500^2,500^2;600^2,600^2
    ;// array of Ti^2
25 Ti_cub =
    [300^3,300^3,300^3;400^3,400^3,400^3;500^3,500^3,500^3;600^3,600^3
    // array of Ti^3

```

```

26 Ti_qd =
    [300^4,300^4,300^4;400^4,400^4,400^4;500^4,500^4,500^4;600^4,600^4,600^4]
    // array of Ti^4
27 Cpi_Ti =
    [39.87*300,39.85*300,39.90*300;45.16*400,45.23*400,45.17*400;50.7*500,50.7*500,50.7*500]
    // Array of Cpi(Heat capacity)*Ti values
28 Cpi_Ti_sqr =
    [39.87*300^2,39.85*300^2,39.90*300^2;45.16*400^2,45.23*400^2,45.17*400^2;50.7*500^2,50.7*500^2,50.7*500^2]
    // Array of Cpi(Heat capacity)*Ti^2 values
29
30 n = 18 ;// Number of data
31
32 // Solve equations (a),(b) & (c) simultaneously
    using matrix
33 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
34 b = [sum(Cpi);sum(Cpi_Ti);sum(Cpi_Ti_sqr)] ;//
    Matrix of constants
35 x = (a)^-1 * b ;// Matrix of solutions a = x(1), b =
    x(2) , c = x(3)
36
37 printf('The solution is Cp = %.2f + %.3e T + %.2e T
    ^2 .\nTherefore coefficients are as follows :',x
    (1),x(2),x(3));
38 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 using Heat Capacity Equations for Each Component

```

1 clear ;
2 clc;
3 // Example 23.5
4 printf('Example 23.5\n\n');

```

```

5 //page no. 695
6 // Solution
7
8 // Basis : 1 g mol of gas
9 //Given
10 T1 = 550 ;// Initial temperature - [degree F]
11 T2 = 200 ;// Final temperature - [degree F]
12 CO2 = 9.2/100 ;// Mole fraction
13 CO = 1.5/100 ;// Mole fraction
14 O2 = 7.3/100 ;// Mole fraction
15 N2 = 82.0/100 ;//Mole fraction
16
17 // Additional data needed :
18 // Coefficients in the heat capacity equations
19 a_N2 = 6.895; // constant
20 b_N2 = 0.7624*10^-3; // coefficient of T
21 c_N2 = -0.7009*10^-7; // coefficient of square T
22 a_O2 = 7.104 ; // constant
23 b_O2 = (0.7851*10^-3); // coefficient of T
24 c_O2 = (-0.5528*10^-7); // coefficient of square T
25 a_CO2 = 8.448; // constant
26 b_CO2 = 5.757*10^-3; // coefficient of T
27 c_CO2 = -21.59*10^-7; // coefficient of square T
28 d_CO2 = 3.059*10^-10; // coefficient of cubic T
29 a_CO = 6.865 ; // constant
30 b_CO = 0.8024*10^-3; // coefficient of T
31 c_CO = -0.7367*10^-7; // coefficient of square T
32
33 // New coefficients after multiplying mole fraction
    of each component
34 a1_N2 = 6.895*N2 ; // constant
35 b1_N2 = N2*0.7624*10^-3; // coefficient of T
36 c1_N2 = (-0.7009*10^-7)*N2; // coefficient of square
    T
37 a1_O2 = 7.104*O2 ; // constant
38 b1_O2 = (0.7851*10^-3)*O2; // coefficient of T
39 c1_O2 = (-0.5528*10^-7)*O2; // coefficient of square
    T

```



```

40 a1_C02 = 8.448*C02; // constant
41 b1_C02 = (5.757*10^-3)*C02; // coefficient of T
42 c1_C02 = (-21.59*10^-7)*C02; // coefficient of
    square T
43 d1_C02 = (3.059*10^-10)*C02; // coefficient of cubic
    T
44 a1_C0 = 6.865*C0; // constant
45 b1_C0 = (0.8024*10^-3)*C0; // coefficient of T
46 c1_C0 = (-0.7367*10^-7)*C0; // coefficient of square
    T
47
48 // Get net coefficients of T , square T and cubic T
    by adding them
49 a_net = a1_N2+a1_C02+a1_C0+a1_02; //Net constant
50 b_net = b1_N2+b1_C02+b1_C0+b1_02; //Net coefficient
    of T
51 c_net = c1_N2+c1_C02+c1_C0+c1_02 ; //Net coefficient
    of square T
52 d_net = d1_C02; //Net coefficient of cubic T
53
54 //Cp_net = a_net + b_net*T + c_net*T^2 + d_net*T^3
55 // Integrate Cp_net*dT over given temperature range
    to get change in enthalpy
56 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]
57
58 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 Tabulated Enthalpy Values

```

1 clear;
2 clc;

```

```

3 // Example 23.6
4 printf('Example 23.6\n\n');
5 //page no. 700
6 // Solution
7
8 //Given
9 N2 = 1 ;// Moles of N2 - [kg mol]
10 P = 100 ;// Pressure of gas - [kPa]
11 T1 = 18 ;// Initial temperature - [degree C]
12 T2 = 1100 ;// Final temperature - [degree C]
13
14 // In the book it is mentioned to use tables in
    Appendix D6 to calculate enthalpy change, we get
15 H_T1 = 0.524; // Initial enthalpy -[kJ/kg mol]
16 H_T2 = 34.715 ;// Final enthalpy - [kJ/kg mol]
17 del_H = H_T2 - H_T1 ;// Change in enthalpy - [kJ/kg
    ]
18
19 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

```

1 clear;
2 clc;
3 // Example 23.7
4 printf('Example 23.7\n\n');
5 //page no. 701
6 // Solution Fig.E23.7
7
8 //Given
9 T1 = 640 ;// Initial temperature -[degree F]
10 T2 = 480 ;// Final temperature -[degree F]
11 P1 = 92 ;// Initial pressure -[psia]

```

```

12 P2 = 52 ;// Final pressure - [psia]
13
14 // We need to use steam table to get H value at
    initial and final condition by interpolation
15 //From steam table
16 //At 90 psia
17 H1_600 = 1328.7 ;//H at 90 psia and 600 F-[Btu/lb]
18 H1_700 = 1378.1 ;//H at 90 psia and 700 F-[Btu/lb]
19 //At 95 psia
20 H2_600 = 1328.4 ;//H at 95 psia and 600 F-[Btu/lb]
21 H2_700 = 1377.8 ;//H at 95 psia and 700 F-[Btu/lb]
22 //H at 92 psia and 600 F
23 H3_600 = H1_600+ ((H2_600-H1_600)/(95-90))*(92-90) ;
    //H at 92 psia and 600 F-[Btu/lb]
24 //H at 92 psia and 700 F
25 H3_700 = H1_700+ ((H2_700-H1_700)/(95-90))*(92-90) ;
    //H at 92 psia and 700 F-[Btu/lb]
26 // Now we need to get V at 92 psia and 640 F
27 H3_640 = H3_600+((H3_700-H3_600)/(700-600))
    *(640-600); //H at 92 psia and 640 F-[Btu/lb]
28
29 //At 50 psia
30 H1_450 = 1258.7 ;//H at 50 psia and 450 F-[Btu/lb]
31 H1_500 = 1282.6 ;//H at 50 psia and 500 F-[Btu/lb]
32 //At 55 psia
33 H2_450 = 1258.2 ;//H at 55 psia and 450 F-[Btu/lb]
34 H2_500 = 1282.2 ;//H at 55 psia and 500 F-[Btu/lb]
35 //V.P at 52 psia and 450 F
36 H3_450 = H1_450+ ((H2_450-H1_450)/(55-50))*(52-50) ;
    //H at 52 psia and 450 F-[Btu/lb]
37 //V at 52 psia and 500 F
38 H3_500 = H1_500+ ((H2_500-H1_500)/(55-50))*(52-50) ;
    //H at 52 psia and 500 F-[Btu/lb]
39 // Now we need to get H at 52 psia and 480 F
40 H3_480 = H3_450+((H3_500-H3_450)/(500-450))
    *(480-450); // H at 52 psia and 480 F-[Btu/lb]
41
42 del_H = H3_480 - H3_640; // Change in enthalpy - [

```

```

    Btu/lb]
43 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 Calculate the final State of Water

```

1 clear ;
2 clc;
3 // Example 23.8
4 printf('Example 23.8\n\n');
5 //page no. 702
6 // Solution
7
8 //Given
9 W = 4 ;// Mass of water -[kg]
10 Ti= 27+273 ;// Initial temperature -[K]
11 Pi = 200 ;// Initial pressure -[kPa]
12 // Neglect the effect of pressure on vloume of
    liquid , therefore
13 Pf = Pi ;// Final pressure -[kPa]
14
15 // From steam table
16 V1 = 0.001004 ;// Specific volume at Ti -[cubic
    metre/kg]
17 V2 = 1000 * V1 ;// Specific volume at final
    temperature(Tf) from given condition in problem -
    [cubic metre/kg]
18
19 // 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
20 va = 0.9024 ;// Specific volume -[cubic metre/kg]
21 Ta = 400 ;// [K]
22 vb = 1.025 ;// Specific volume -[cubic metre/kg]

```

```

23 Tb = 450 ;// [K]
24 vf = V2 ;// Final specific volume -[cubic metre/kg]
25
26 // Linear interpolation
27 m=(Tb - Ta)/(vb - va);// slope
28 Tf=Ta + m*(vf - va) ;// Final temperature - [K]
29
30 printf('\n Final temperature is %.0f K.\n',Tf);

```

Scilab code Exa 23.9 Calculate Enthalpy Difference between two States by Pressure Enthalpy Chart for Butane

```

1 clear ;
2 clc;
3 // Example 23.9
4 printf('Example 23.9\n\n');
5 //page no. 704
6 // Solution
7
8 //Given
9 mv = 1 ;// Mass of saturated vapour - [lb]
10 P1 = 2 ;// Initial pressure -[atm]
11 P2 = 20 ;// Final pressure -[atm]
12
13 // Additional data is obtained from figure 23.6 of
14 // the book on page no. 703
14 H_2 = 179 ;// Specific enthalpy at 2 atm - [Btu/lb]
15 H_20 = 233 ;// Specific enthalpy at 20 atm - [Btu/
16 // lb]
16 V_2 = 3.00 ;// Specific volume at 2 atm - [cubic
17 // feet/lb]
17 V_20 = 0.30 ;// Specific volume at 20 atm - [cubic
18 // feet/lb]
18 T_2 = 72 ;// Temperature at 2 atm -[degree F]
19 T_20 = 239 ;// Temperature at 20 atm -[degree F]

```

```
20 del_H = H_20 - H_2 ;// Change in specific enthalpy
    -[Btu/lb]
21 del_V = V_20 - V_2 ;// Change in specific volume -[
    cubic feet/lb]
22 del_T = T_20 - T_2 ;// Change in temperature -[
    degree F]
23 printf('(a) Change in specific enthalpy is %.0f Btu/
    lb.\n',del_H);
24 printf('(b) Change in specific volume is %.2f cubic
    feet/lb.\n',del_V);
25 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

```
1 clear ;
2 clc;
3 // Example 24.1
4 printf('Example 24.1\n\n');
5 //page no. 720
6 // Solution Fig. E24.1
7
8 // Assumptions to be made in eqn. 24.1 in following
   segment
9 printf('Assumptions to be made in eqn. 24.1 in
   following segments are:\n');
10 //(a)- 1 to 5
11 printf('\n(a)- 1 to 5.\n');
12 printf(' 1. Change in potential energy(del_PE) = 0(
   no change in level) .\n');
13 printf(' 2. Probably change in kinetic energy(
   del_KE)=0 .\n');
```

```

14 printf(' 3. Change in energy = 0 (process appears
    to be steady).\n');
15 printf(' Result : Q + W = del_H.\n');
16
17 //(b) 4 to 5
18 printf('\n\n(b) 4 to 5.\n');
19 printf(' 1. Q = W = 0 \n');
20 printf(' 2. Probably change in kinetic energy(
    del_KE)=0.\n');
21 printf(' 3. Change in energy = 0 (process appears
    to be steady).\n');
22 printf(' Result : del_H = -del_PE . \n');
23
24 //(c) 3 to 4
25 printf('\n\n(c) 3 to 4.\n');
26 printf(' 1. Q = W = 0 \n');
27 printf(' 2. Probably change in kinetic energy(
    del_KE)=0.\n');
28 printf(' 3. Change in energy = 0 (process appears
    to be steady).\n');
29 printf(' Result : del_H = -del_PE . \n');
30
31 //(d) 3 to 5
32 printf('\n\n(d) 3 to 5.\n');
33 printf(' 1. Q = W = 0 \n');
34 printf(' 2. Probably change in kinetic energy(
    del_KE)=0.\n');
35 printf(' 3. Change in energy = 0 (process appears
    to be steady).\n');
36 printf(' 4. Change in potential energy(del_PE) = 0(
    no change in level) .\n');
37 printf(' Result : del_H = 0 . \n');
38
39 //(e)- 1 to 3
40 printf('\n\n(e) 1 to 3.\n');
41 printf(' 1. Change in potential energy(del_PE) = 0(
    no change in level) .\n');
42 printf(' 2. Probably change in kinetic energy(

```



```

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

```

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

```

1 clear ;
2 clc;
3 // Example 24.2
4 printf('Example 24.2\n\n');
5 //page no. 725
6 // Solution
7
8 printf('Table to carry out degree of freedom
    analysis:\n');
9 // Number of variables involved
10 printf('\nI. Number of variables involved.\n');
11 printf('\n For materials:\n');
12 printf('      Hot gas : 4 component flows , T, and
    p          6 \n');
13 printf('      Cool gas : 4 component flows , T, and
    p          6\n');
14 printf('      Water in : 1 component flow , T, and
    p          3\n');
15 printf('      Water out : 1 component flow , T, and
    p          3\n');
16 printf('\n Energy:\n');
17 printf('      Q and W
    2 \n');
18 printf('      H,KE and PE associated with each
    stream flow          12 \n');
19 printf('\n

```

```

-----
\n');
20 printf(' Total

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

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

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

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

```

$$(32 - 32) = 0 \text{ \n' });$$

Scilab code Exa 24.3 Application of Energy Balance

```
1 clear ;
2 clc;
3 // Example 24.3
4 printf('Example 24.3\n\n');
5 //page no. 728
6 // Solution Fig. E24.3
7
8 // Given
9 m_CO2 = 10 ;// mass of CO2 - [lb]
10 Ti_CO2 = 80 ;// Initial temperature of CO2 - [
    degree F]
11 Vi = 4.0 ;// Initial volume of CO2-[cubic feet]
12 f_CO2 = 40/100 ;// Fraction of CO2 that convert to
    liquid finally
13 s_Vi = Vi /m_CO2 ;// Initial specific volume of CO2
    - [cubic feet/lb]
14 s_Vf = s_Vi ;// Constant volume -[cubic feet/lb]
15 // Use the CO2 chart in Appendix J to necessary
    data, according to book it is
16 // CO2 is gas at start of process and reference
    state for the CO2 chart is -40 degree F ,
    saturated liquid
17 // From chart
18 Pi = 300 ;// Intial pressure - [psia]
19 del_Hi = 160 ;// Intial change in specific enthalpy
    - [Btu/lb]
20 // Now again use chart to get fnal condition fixed
    by constant volume line and quality 0.6 ,
    according to book it is
21 del_Hf = 81 ;// Final change in specific enthalpy -
    [Btu/lb]
```

```

22 Pf = 140 ;//Final pressure - [psia]
23 // Use conditions given in problem ( W= 0 ,since
    volume is constant ,therefore del_PE and del_KE
    =0 ),simplifying the energy balance equation we
    get Q = del_H - del_(PV)
24 // Analysing the given conditions dof of system = 0
    , with 1 eqn. and 1 unknown Q
25 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]
26
27 printf(' Heat removed from the extinguisher is %i
    Btu .\n',Q);

```

Scilab code Exa 24.4 Applications of Energy Balance to Plasma Etching

```

1 clear ;
2 clc;
3 // Example 24.4
4 printf('Example 24.4\n\n');
5 //page no. 730
6 // Solution
7
8 // Pick the system as gas plus heater
9 // Given
10 Pi = 1.5 ;// Intial pressure - [Pa]
11 Vi = 2*10^-3 ;// Initial volume of gas - [cubic
    metre]
12 Ti = 300 ;// Initial temperature - [K]
13 W = 480 ;// Work done by heater on system
14 t = 5 ;// Time for which current is supplied -[ min]
15 m_ht = 12 ;// Mass of the heater - [g]
16 C_ht = 0.35 ;// Heat capacity of heater - [ J/gK]
17 R = 8.314 ;// Ideal gas constant - [(Pa*cubic metre)
    /(g mol* K)]

```

```

18
19 // It is assumed that heat transfer across system
    boundary for this short time is negligible ,
    therefore Q = 0
20 // Using the above assumption the equation reduces
    to del_U = W, therefore
21 del_U = W ;// Change in nternal energy - [J]
22
23 // Gas is assumed to be ideal , therefore get n by
    using pv = nRT
24 n = (Pi*Vi)/(R*Ti) ;// Number of moles of argon gas
    -[g mol]
25 Cp = (5/2)* R ;// Specific heat capacity of argon
    gas at constant pressure - [ J/gK]
26 Cv = Cp - R ;// Specific heat capacity of argon gas
    at constant volume - [ J/gK]
27 // del_Ug = n*Cv*(Tf - Ti) - change in internal
    energy of gas
28 // del_Uh = m_ht*C_ht*(Tf - Ti) - change in internal
    energy of gas
29 // get total change in internal energy = del_Ug +
    del_Uh , and solve it for Tf ( final temperature
    )
30 def f (' [y]=f(Tf) ', 'y=m_ht*C_ht*(Tf - Ti) + n*Cv*(Tf -
    Ti) - del_U ');
31 Tf=fsolve(400,f) ;// Final temperature -[K]
32 funcprot(0);
33 printf(' Final temperature of gas is %.0f K .\n',Tf
    );
34
35 Pf = (Tf/Ti)*Pi ;// Final pressure - [Pa]
36 printf(' Final pressure in chamber is %.2f Pa .\n'
    ,Pf);

```

Scilab code Exa 24.5 Energy Balance applied to a Batch Process

```

1  clear ;
2  clc;
3  // Example 24.5
4  printf('Example 24.5\n\n');
5  //page no. 732
6  // Solution Fig. E24.5
7
8  // Pick the system as shown in above figure of book
9  // Given
10 m_water = 10 ;// Mass of water - [lb]
11 T_water = 35 ;// Temperature of water - [degree F]
12 m_ice = 4 ;// Mass of ice - [lb]
13 T_ice = 32 ;// Temperature of ice - [degree F]
14 m_stm = 6 ;// Initial mass of steam -[lb]
15 T_stm = 250 ;// Temperature of stm - [degree F]
16 p = 20 ;// Pressure of system -[psia]
17
18 m_total = m_water + m_ice + m_stm ;// Mass of H2O in
    three phases initially -[lb]
19 // By following conditions of book,  $Q = 0$ ,  $W = 0$  ,
    change in PE and change in KE = 0, the energy
    balance reduces to  $\text{del}_U = 0$ 
20
21 // According to book additional information is
    obtained from the steam table and CD at given
    conditions ,it is as follows
22 U_ice = -143.6 ;// Specific internal energy of ice
    -[Btu/lb]
23 U_water = 3.025 ;// Specific internal energy of
    water -[Btu/lb]
24 U_stm = 1092.25 ;// Specific internal energy of
    steam -[Btu/lb]
25 V_water = 0.0162 ;// Specific volume of water -[
    cubic feet/lb]
26 V_stm = 20.80 ;// Specific volume of steam -[cubic
    feet/lb]
27 V_total = m_stm*V_stm ;//Total volume of container
    ignoring volume of water and ice as they are

```

```

negligible
28
29 V_sys = V_total/m_total ;// Specific volume of
    system -[cubic feet/lb]
30 U_sys =(m_water*U_water + m_ice*U_ice + m_stm*U_stm)
    /m_total ;// Final specific internal energy of
    system -[Btu/lb]
31
32 // Trial and error method
33 // 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
34 // Obtain necessary data from steam table at
    corresponding temperatures
35
36 T1 = 190 ;// assumed temperature
37 U1 = [157.17 1071.83] ;//specific internal energy of
    liquid and vapour respetively -[Btu/lb]
38 V1 = [0.0165 41.01] ;// Specific volume of liquid
    and vapour respetively -[cubic feet/lb]
39 x1 = V_sys/V1(2) ;// Quality of vapour
40 U1_sys = (1-x1)*U1(1) + x1*U1(2); // Specific
    internal energy of system at T1-[Btu/lb]
41
42 T2 = 200 ;// assumed temperature
43 U2 = [168.11 1073.96];// specific internal energy
    of liquid and vapour respetively -[Btu/lb]
44 V2 = [0.017 33.601] ;// Specific volume of liquid
    and vapour respetively -[cubic feet/lb]
45 x2 = V_sys/V2(2) ; // Quality of vapour
46 U2_sys = (1-x2)*U2(1) + x2*U2(2) ;// Specific
    internal energy of system at T2-[Btu/lb]
47
48 // Check whether assumption is right
49 if (U_sys > U1_sys )
50     if ( U_sys <U2_sys)
51         printf('Assumption is right , now find exact
            temperature by interpolation between 2

```

```

        assumed temperatures.\n');
52     else
53         printf('Assumption is wrong, assume
        different T2.\n');
54     end
55     else
56     printf('Assumption is wrong, assume different T1.\n'
        );
57     end
58
59 // Interpolation , to get final temperature
        corresponding to U_sys
60 T_sys = T1 + ((T2 - T1)*(U_sys - U1_sys))/(U2_sys -
        U1_sys);
61
62     printf(' The final temperature obtained by
        interpolation between 2 assumed temperatures is
        %.2f degree F.\n',T_sys);
63
64 // 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
65 V_vap = 39.35 ;//specific volume of vapour data at
        final temperature -[cubic feet/lb]
66 x = V_sys /V_vap ;// Quality of gas at final
        temperature
67
68 //Final state
69 Vap = m_total*x ;// Mass of vapour at final state -
        [lb]
70 stm_con = m_stm - Vap ;// Mass of steam condenses -
        [lb]
71
72     printf('\nTherefore , mass of steam condenses is %.2f
        lb.\n',stm_con);

```

Scilab code Exa 24.6 Applications of Energy Balance to Pumping Water

```
1 clear ;
2 clc;
3 // Example 24.6
4 printf('Example 24.6\n\n');
5 //page no. 736
6 // Solution Fig. E24.6
7
8 // Pick the system as shown in above figure of book
9 // Given
10 h1 = -15 ;// Initial level of water from ground
    level -[ft]
11 h2 = 165 ;//Final level of water from ground level
    -[ft]
12 V_rate = 200 ;// Volume flow rate of water - [gal/hr
    ]
13 Q1 = 30000 ;// Heat input by heater - [Btu/hr]
14 Q2 = 25000 ;// Heat lost by system -[Btu/hr]
15 T1 = 35 ;// Initial temperature of water - [degree F
    ]
16 g = 32.2 ;// Acceleration due to gravity - [ft/
    square second]
17 p_pump = 2 ;// Power of pump -[hp]
18 f_w = 55/100 ;// Fraction of rated horsepower that i
    used in pumping water
19 Cp = 1 ;// Specific heat capacity of water - [Btu/lb
    *F]
20
21 // Use following conditions to simplify the energy
    balance
22 // 1. Proces is in steady state , so change in
    energy = 0
23 // 2. m1 = m2 = m
```

```

24 // 3. change in KE = 0 , because we will assume that
    v1 = v2 = 0
25 // The energy balance reduce to Q + W = del_(H*m +
    PE*m)
26
27 m = V_rate * 8.33 ;// Total mass of water pumped -[
    lb]
28 del_PE = (m* g *(h2 - h1))/(32.2*778) ;// Change in
    PE - [Btu/hr]
29 Q = Q1 - Q2 ;// Net heat exchange -[Btu/hr]
30 W = 2* f_w * 60 * 33000/778 ;// Work on system - [
    Btu/hr]
31 del_H = Q + W - del_PE ;// By using reduced energy
    balance - [Btu/hr]
32 // Also del_H = m* Cp * (T2 - T1), all is known
    except T2 , solve for T2
33 deff(' [y] = f(T2) ', 'y = m*Cp*(T2-T1) - del_H ');
34 T2 = fsolve(40,f) ;// Boiling point temperature
35 funcprot(0);
36
37 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

```

1 clear ;
2 clc;
3 // Example 24.7
4 printf('Example 24.7\n\n');
5 //page no. 738
6 // Solution Fig. E24.7
7
8 // Pick the system as shown in above figure of book
9 // Given
10 T_stm = 250 + 273 ;// Temperature of entering steam

```

```

    - [K ]
11 Q_loss = -1.5 ;// Rate of heat loss from system - [
    kJ/s ]
12 T_mi = 20 + 273 ;//Temperature of entering material
    -[K ]
13 T_mf = 100 + 273 ;// Temperature of material after
    heating - [K]
14 m_m = 150 ;// Mass of charged material - [kg]
15 Cp_m = 3.26 ;// Average heat capacity of material -
    [ J/(g*K) ]
16
17 // Use following conditions to simplify the energy
    balance
18 // 1. Proces is not in steady state , so change in
    energy not equals 0.
19 // 2. Assume del_KE and del_PE = 0.
20 // 3. Assume del_KE and del_PE = 0, for entering and
    exiting material .
21 // 4. W = 0.
22 // 5. Assume m1 = m2 = m_stm
23 // The energy balance reduce to      del_E = del_U =
    Q - del_(H*m) .... eqn. (b)
24
25 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]
26 Q_loss_total = Q_loss * 3600; // Total heat loss by
    system n 1 hour - [kJ]
27 // 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
28 del_H_steam = -1701 ;// Specific change in enthalpy
    value of saturated steam -[kJ/kg]
29 // Get mass of steam per kg charge from reduced
    energy balance(eqn. (b))
30 m_stm_total = (del_U - Q_loss_total)/(-del_H_steam)
    ;// Total mass of stea used - [kg]

```

```

31 m_stm = m_stm_total/m_m ;// Mass of steam used per
    kg of charge - [kg]
32
33 printf(' Mass of steam used per kg of charge is %.3
    f kg .\n',m_stm);

```

Scilab code Exa 24.8 Sterilization of a Fermentation Medium

```

1 clear ;
2 clc;
3 // Example 24.8
4 printf('Example 24.8\n\n');
5 //page no. 741
6 // Solution Fig. E24.8
7
8 // Pick the system of whole process as shown in
    above figure of book
9 // Given
10 Q = 1.63 ;// Heat loss from the process - [ kW ]
11 m_bm = 150 ;// Mass flow rate of biological media
    into the sterlizer -[kg/min]
12 T_bm = 50 +273 ;// Temperature of biological media
    into the sterlizer - [K]
13 T_sm = 75 + 273 ;// Temperature of sterlize media
    out of the sterlizer - [K]
14 P_ss = 300 ;// Pressure of saturated steam entering
    the steam heater - [kPa]
15 P_sc = 300 ;// Pressure of saturated condensate
    exiting the steam heater - [kPa]
16
17 // Additional data of change in enthalpy is obtained
    from the steam table , according to book the data
    are
18 H_w1 = 207.5 ;// Change in specific enthalpy of
    water at 50 degree C - [kJ/kg]

```

```

19 H_w2 = 310.3 ;// Change in specific enthalpy of
    water at 75 degree C - [kJ/kg]
20 H_ss = 2724.9 ;//Change in specific enthalpy of
    saturated steam entering the steam heater at 300
    kPa - [kJ/kg]
21 H_sc = 561.2 ;//Change in specific enthalpy of
    saturated condensate exiting the steam heater at
    300 kPa - [kJ/kg]
22
23 // Use following conditions to simplify the energy
    balance
24 // 1. Proces is in steady state , so change in
    energy = 0.
25 // 2. Assume del_KE and del_PE = 0.
26 // 3. W = 0.
27 // 4. Assume m1 = m2 = m_stm
28 // The energy balance reduce to  $Q = H_{out} - H_{in}$ 
    , using it
29 m_sm = m_bm ;// By material balance -[kg/min]
30 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]
31
32 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 Distillation Problem

```

1 clear ;
2 clc;
3 // Example 24.9
4 printf('Example 24.9\n\n');
5 //page no. 742
6 // Solution Fig. E24.9a and Fig. E24.9b

```

```

7
8 // Given
9
10 // For material balance
11 F = 20000 ;// Feed rate of saturated liquid - [kg/h]
12 F_Bz = 0.5 ;// Fraction of benzene in feed
13 F_Tol = 0.5 ;// Fraction of toluene in feed
14 D_Bz = 0.98 ;// Fraction of benzene in distillate
15 D_Tol = 0.02 ;// Fraction of toluene in distillate
16 B_Bz = 0.04 ;// Fraction of benzene in bottoms
17 B_Tol = 0.96 ;// Fraction of toluene in bottoms
18 R_by_D = 4.0 ;// Recycle ratio
19 // Analysing the condition for material balance ,
    degree of freedom is 0.
20 // Solve equations obtained by material balances ,
    simultaneously to get B and D
21 a = [1 1;B_Bz D_Bz] ;// Matrix formed by
    coefficients of unknown
22 b = [ F ; F_Bz*F ] ;// Matrix formed by constants
23 x = a\b ;// Matrix of solutions
24 B = x(1) ;// Bottoms - [kg/h]
25 D = x(2) ;// Distillate - [kg/h]
26 R = D * R_by_D ;// Recycle - [kg/h]
27 V = R + D ;// Overhead vapour - [kg/h]
28
29 // For energy balance
30 // According to book additional data obtained from
    the fig.E24.9b are
31 H_F = 165 ;// Change in enthalpy of F - [kJ/kg]
32 H_B = 205 ;// Change in enthalpy of B - [kJ/kg]
33 H_D = 100 ;// Change in enthalpy of D - [kJ/kg]
34 H_R = 100 ;// Change in enthalpy of R - [kJ/kg]
35 H_V = 540 ;// Change in enthalpy of V - [kJ/kg]
36
37 Qc = R*H_R + D*H_D - V*H_V ;// The heat duty in the
    condenser - [kJ]
38 Qr = D*H_D + B*H_B - F*H_F - Qc ;// The heat duty to
    the reboiler - [kJ]

```

```
39
40 printf('  Ditillate (D)                %.2e kg/h.\n',D);
41 printf('  Bottoms (B)                  %.2e kg/h.\n',B)
    ;
42 printf('  The heat duty in the condenser (Qc)
    %.2e kJ/h.\n',Qc);
43 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 Measurements

```
1 clear ;
2 clc ;
3 // Example 25.1
4 printf('Example 25.1\n\n');
5 //page no. 766
6 // Solution Fig. E25.1
7
8 // Given
9 // C(s) + O2(g) --> CO2(g)
10 // CO(g) + (1/2)(O2)(g) --> CO2 (g)
11 // (A)
12 // (B)
11 Qa = -393.51 ;// Heat of reaction of reaction (a) -
    [kJ/g mol C]
12 Qb = -282.99 ;// Heat of reaction of reaction (b) -
    [kJ/g mol CO]
13 del_Ha = Qa ;// Change in enthalpy of reaction A - [
    kJ/g mol C]
```



```

14 del_Hb = Qb ;// Change in enthalpy of reaction B - [
    kJ/g mol CO]
15
16 // 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
17 del_Hfc = del_Ha - del_Hb ;// Standard heat of
    formation of CO - [kJ/g mol C]
18 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

```

1 clear ;
2 clc;
3 // Example 25.2
4 printf('Example 25.2\n\n');
5 //page no. 767
6 // Solution
7
8 // Given
9 // The main reaction is (1/2)*H2(g) +(1/2)* Cl2(g)
    --> HCl(g) (A)
10 // Look in Appendix F for heat of formation of H2 ,
    Cl2 and HCl
11 H_H2 = 0 ;// Standard heat of formation of H2 -[kJ/
    g mol H2]
12 H_Cl2 = 0 ;// Standard heat of formation of Cl2 -[kJ
    / g mol Cl2]
13 H_HCl = -92.311 ;// Standard heat of formation of
    HCl -[kJ/ g mol HCl]
14
15 H_f = 1*H_HCl - (1/2)*(H_H2 + H_Cl2) ; // Standard
    heat of formation of HCl by reaction - [kJ/ g mol

```

```

HCl]
16 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 Standard Heats of Formation

```

1 clear ;
2 clc;
3 // Example 25.3
4 printf('Example 25.3\n\n');
5 //page no. 771
6 // Solution
7
8 // Given
9 // The main reaction is 4*NH3(g) + 5*O2(g) --> 4*NO(
g) + 6*H2O (A)
10 H_fNH3 = -46.191 ;// Standard heat of formation of
NH3 -[kJ/ g mol]
11 H_fO2 = 0 ;//Standard heat of formation of O2 -[kJ/
g mol]
12 H_fNO = 90.374 ;// Standard heat of formation of NO
-[kJ/ g mol]
13 H_fH2O = -241.826 ;// Standard heat of formation of
H2O -[kJ/ g mol]
14
15 // Heat of above reaction is calculated by eqn. 25.1
16 H_rxn = ((4*H_fNO + 6*H_fH2O) - (4*H_fNH3 + 5*H_fO2)
)/4 ;// Heat of above reaction -[kJ/ g mol NH3]
17
18 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 different from the Standard Conditions

```
1 clear ;
2 clc;
3 // Example 25.4
4 printf('Example 25.4\n\n');
5 //page no. 773
6 // Solution
7
8 // Given
9 P1 = 1 ;// Initial pressure - [atm]
10 P2 = 1 ;// Final pressure - [atm]
11 T1 = 500 ;// Initial temperature -[degree C]
12 T2 = 500 ;// Final temperature -[degree C]
13
14 // The main reaction is  $\text{CO}_2(\text{g}) + 4\text{H}_2(\text{g}) + \text{---} \rightarrow 2\text{H}_2\text{O}(\text{g}) + \text{CH}_4(\text{g})$  (A)
15 // Data obtained from above reaction
16 m_CO2 = 1 ;// Moles of CO2 - [ g mol]
17 m_H2 = 4 ;// Moles of H2 - [ g mol]
18 m_H2O = 2 ;// Moles of H2O - [ g mol]
19 m_CH4 = 1 ;// Moles of CH4 - [ g mol]
20
21 // Additional required data are obtained from CD,
    according to book it is a follows-
22 H_fCO2 = -393.250 ;// Heat of formation of CO2 - [kJ
    /g mol]
23 H_fH2 = 0 ;// Heat of formation of H2 - [kJ/g mol]
24 H_fH2O = -241.835 ;// Heat of formation of H2O - [kJ
    /g mol]
25 H_fCH4 = -74.848 ;// Heat of formation of CH4 - [kJ/
    g mol]
26
27 H_CO2 = 21.425 ;// Change in enthalpy during
    temperature change from 25 to 500 degree C of
    CO2 - [kJ/g mol]
28 H_H2 = 13.834 ;// Change in enthalpy during
```

```

    temperature change from 25 to 500 degree C of H2
    - [kJ/g mol]
29 H_H2O = 17.010 ;// Change in enthalpy during
    temperature change from 25 to 500 degree C of H2O
    - [kJ/g mol]
30 H_CH4 = 23.126 ;// Change in enthalpy during
    temperature change from 25 to 500 degree C of CH4
    - [kJ/g mol]
31
32 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
33 sum_H_rct = m_CO2*H_CO2 + m_H2*H_H2 ;// sum of heat
    of formation of reactant - [kJ]
34 sum_H_pdt = m_CH4*H_CH4 + m_H2O*H_H2O ;//sum of heat
    of formation of product - [kJ]
35 // Heat of above reaction is calculated by eqn. 25.4
36 H_rxn_500 = sum_H_pdt - sum_H_rct + H_rxn_25 ;//
    Heat of reaction at 500 C
37
38 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 Process in which Reactants enter and Products leave at Different Temperatures

```

1 clear ;
2 clc;
3 // Example 25.5
4 printf('Example 25.5\n\n');
5 //page no. 775
6 // Solution
7
8 // Given
9 // The main reaction is CO2(g) + 4H2(g) + --> 2H2O(g)

```

```

    ) + CH4(g) (A)
10 // Data obtained from above reaction
11 m_CO2 = 1 ;// Moles of CO2 - [ g mol]
12 m_H2 = 4 ;// Moles of H2 - [ g mol]
13 m_H2O = 2 ;// Moles of H2O - [ g mol]
14 m_CH4 = 1 ;// Moles of CH4 - [ g mol]
15 P1 = 1 ;// Initial pressure - [atm]
16 P2 = 1 ;// Final pressure - [atm]
17
18 T1_CO2 = 800 ;// Initial temperature of entering CO2
    -[K]
19 T1_H2 = 298 ;// Initial temperature of entering H2
    -[K]
20 T2 = 1000 ;// Temperature of exiting product - [K]
21
22 // The main reaction is CO2(g) + 4H2(g) + --> 2H2O(g
    ) + CH4(g) (A)
23 // Data obtained from above reaction
24 m1_CO2 = 1 ;// Moles of entering CO2 - [ g mol]
25 m1_H2 = 4 ;// Moles of entering H2 - [ g mol]
26 f_con = 70/100 ;// Fractional conversion of CO2
27 m2_H2O = 2*f_con ;// Moles of H2O in product - [ g
    mol]
28 m2_CH4 = 1*f_con ;// Moles of CH4 in product - [ g
    mol]
29 m2_CO2 = m1_CO2*(1-f_con) ;// Moles of CO2 in
    product - [ g mol]
30 m2_H2 = m1_H2*(1-f_con) ;// Moles of CO2 in product
    - [ g mol]
31
32 // Additional required data are obtained from CD,
    according to book it is a follows-
33 H_fCO2 = -393.250 ;// Heat of formation of CO2 - [kJ
    /g mol]
34 H_fH2 = 0 ;// Heat of formation of H2 - [kJ/g mol]
35 H_fH2O = -241.835 ;// Heat of formation of H2O - [kJ
    /g mol]
36 H_fCH4 = -74.848 ;// Heat of formation of CH4 - [kJ/

```

```

    g mol]
37
38
39 H1_CO2 = 22.798 ;// Change in enthalpy during
    temperature change from 298K to 800 K of CO2 - [
    kJ/g mol]
40 H1_H2 = 0 ;// Change in enthalpy during temperature
    change from 298K to 298 K of H2 - [kJ/g mol]
41 H2_H2O = 25.986 ;// Change in enthalpy during
    temperature change from 298K to 1000 K of H2O -
    [kJ/g mol]
42 H2_CH4 = 38.325 ;// Change in enthalpy during
    temperature change from 298K to 1000 K of CH4 -
    [kJ/g mol]
43 H2_CO2 = 33.396; // Change in enthalpy during
    temperature change from 298K to 1000 K of CO2 -
    [kJ/g mol]
44 H2_H2 = 20.620; // Change in enthalpy during
    temperature change from 298K to 1000 K of H2 - [
    kJ/g mol]
45
46 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]
47 H_rxn_ac = f_con*H_rxn_25 ;// Heat of reaction
    actual - [kJ]
48 sum_H_rct = m1_CO2*H1_CO2 + m1_H2*H1_H2 ;// sum of
    heat of formation of reactant - [kJ]
49 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]
50 // Heat of above reaction is calculated by eqn. 25.4
51 H_rxn = sum_H_pdt - sum_H_rct + H_rxn_ac ;// Heat
    of reaction -[kJ/ g mol CO2]
52
53 // By above conditions the energy balance reduces to
    Q = del_H , therefore
54 Q = H_rxn ;// Heat transfer to/from the reactor - [

```

```

    kJ]
55 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

```

1 clear ;
2 clc;
3 // Example 25.6
4 printf('Example 25.6\n\n');
5 //page no. 776
6 // Solution
7
8 // Given
9 H_EtOH = -1330.51 ; // Change in enthalpy of ethanol
    -[kJ/g mol]
10 H_Ac = -887.01 ; // Change in enthalpy of acetate -[
    kJ/g mol]
11 H_Fr = -221.75 ; // Change in enthalpy of formate -[
    kJ/g mol]
12 H_Lc = -1330.51 ; // Change in enthalpy of lactate -[
    kJ/g mol]
13 H_Mn = -2882.78 ; // Change in enthalpy of mannitol
    -[kJ/g mol]
14 mol_EtOH = 1.29 ; //ethanol produced / g mol
    mannitol -[g mol]
15 mol_Ac = 0.22 ; //acetate produced / g mol
    mannitol -[g mol]
16 mol_Fr = 1.6 ; //formate produced / g mol
    mannitol-[g mol]
17 mol_Lc = 0.4 ; //lactate produced / g mol mannitol
    -[g mol]
18 mol_Mn = 1.0 ; //mannitol produced / g mol

```

```

    mannitol-[g mol]
19 B_growth = 40.5    ;// Biomass growth -[g cells/g mol
    mannitol]
20
21 // (a)
22 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]
23 printf(' (a) Net enthalpy change for several
    products (metabolites) per g mol mannitol
    consumed is %.2f kJ.\n',del_H1) ;
24
25 //(b)
26 del_H2 = del_H1 / B_growth ;//Net enthalpy change
    for several products (metabolites) per g cells
    produced-[kJ]
27 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

```

1 clear ;
2 clc;
3 // Example 25.7
4 printf('Example 25.7\n\n');
5 //page no. 777
6 // Solution
7
8 // Given
9 //Bhopal Process
10 //CH3NH2 + COCl2 + --> C2H3NO +2HCl      (
    A)
11 //C2H3NO + C10H8O --> C12H11O2N

```



```

                                (B)
12 //Alternate process
13 //C10H8O + COCl2 --> C11H7O2Cl
                                (C)
14 //C11H7O2Cl + CH3NH2 --> C12H11O2N + HCl      (D)
15
16 H_Cb = -26 ;//Standard heat of formation of carbaryl
    (C12H11O2N) -[kJ/ g mol]
17 H_HCl = -92.311 ;//Standard heat of formation of HCl
    -[kJ/ g mol]
18 H_Ma = -20.0 ;//Standard heat of formation of methyl
    amine(CH3NH2) -[kJ/ g mol]
19 H_Mi = -9*10^4 ;//Standard heat of formation of
    methyl isocyanate(C2H3NO) -[kJ/ g mol]
20 H_Nc = -17.9 ;//Standard heat of formation of 1-
    Naphthalenyl chloroformate(C11H7O2Cl) -[kJ/ g mol]
21 H_N = 30.9 ;//Standard heat of formation of naphthol(
    C10H8O) -[kJ/ g mol]
22 H_P = -221.85 ;//Standard heat of formation of
    phosgene(COCl2) -[kJ/ g mol]
23
24 H_rxn_a = (2*H_HCl + 1*H_Mi) - (1*(H_Ma) + 1*H_P )
    ;// Heat of reaction (A)-[kJ]
25 H_rxn_b = (1*H_Cb ) - (1*(H_Mi) + 1*H_N ) ;// Heat
    of reaction (B)-[kJ]
26 H_rxn_c = (1*H_Nc) - (1*(H_N) + 1*H_P ) ;// Heat of
    reaction (C)-[kJ]
27 H_rxn_d = (1*H_Cb + 1*H_HCl) - (1*(H_Nc) + 1*H_Ma )
    ;// Heat of reaction (D)-[kJ]
28
29 //Bhopal Process
30 printf(' Bhopal process .\n') ;
31 printf(' (a) Heat of reaction (A) is %.1e kJ.\n',
    H_rxn_a) ;
32 printf(' (b) Heat of reaction (B) is %.1e kJ.\n',
    H_rxn_b) ;
33
34 //Alternate process

```

```

35 printf('\n Alternate process .\n') ;
36 printf(' (c) Heat of reaction (C) is %.2f kJ.\n',
    H_rxn_c) ;
37 printf(' (d) Heat of reaction (D) is %.2f kJ.\n',
    H_rxn_d) ;
38 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 different from the Standard Conditions with Heat of Formation Merged with Sensible Heat

```

1 clear ;
2 clc;
3 // Example 25.8
4 printf('Example 25.8\n\n');
5 //page no. 782
6 // Solution
7
8 // Given
9 P1 = 1 ;// Initial pressure - [atm]
10 P2 = 1 ;// Final pressure - [atm]
11 T1 = 500 ;// Initial temperature -[degree C]
12 T2 = 500 ;// Final temperature -[degree C]
13
14 // The main reaction is CO2(g) + 4H2(g) + --> 2H2O(g
    ) + CH4(g) (A)
15 // Data obtained from above reaction
16 m_CO2 = 1 ;// Moles of CO2 - [ g mol]
17 m_H2 = 4 ;// Moles of H2 - [ g mol]
18 m_H2O = 2 ;// Moles of H2O - [ g mol]
19 m_CH4 = 1 ;// Moles of CH4 - [ g mol]
20
21 // Additional required data are obtained from CD,

```

```

    according to book it is a follows-
22 H_fCO2 = -393.250; // Heat of formation of CO2 - [kJ
    /g mol]
23 H_fH2 = 0 ;// Heat of formation of H2 - [kJ/g mol]
24 H_fH2O = -241.835 ;// Heat of formation of H2O - [kJ
    /g mol]
25 H_fCH4 = -74.848 ;// Heat of formation of CH4 - [kJ/
    g mol]
26
27 H_C02 = 21.425 ;// Change in enthalpy during
    temperature change from 25 to 500 degree C of
    CO2 - [kJ/g mol]
28 H_H2 = 13.834 ;// Change in enthalpy during
    temperature change from 25 to 500 degree C of H2
    - [kJ/g mol]
29 H_H2O = 17.010 ;// Change in enthalpy during
    temperature change from 25 to 500 degree C of H2O
    - [kJ/g mol]
30 H_CH4 = 23.126 ;// Change in enthalpy during
    temperature change from 25 to 500 degree C of CH4
    - [kJ/g mol]
31
32 H_in = (H_fCO2 + H_C02)*m_CO2 + (H_fH2 + H_H2)*m_H2
    ;// Enthalpy change for inputs -[kJ]
33 H_out = (H_fH2O + H_H2O)*m_H2O + (H_fCH4 + H_CH4)*
    m_CH4 ; // Enthalpy change for outputs -[kJ]
34 del_H = H_out - H_in ;// Net enthalpy change of
    process -[kJ]
35
36 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 Products leave at Different Temperatures

```

1 clear ;
2 clc;
3 // Example 25.9
4 printf('Example 25.9\n\n');
5 //page no. 783
6 // Solution
7
8 // Given
9 // The main reaction is CO(g,1 atm,298 K) + (1/2)O2(
    g,1 atm,400K)  --> CO2(g,1at,300 K)
                                (A)
10 // Conditions of input and output gases are shown in
    above reaction
11 m_CO = 1 ;// Moles of CO input- [g mol]
12 m1_O2 = 1.5 ;// Moles of O2 input - [g mol]
13 m_CO2 = 1 ;// Moles of CO2 output - [g mol]
14 m2_O2 = 1 ;// Moles of O2 output - [g mol]
15 T_in_CO = 298 ;// Temperature of entering CO -[K]
16 T_in_O2 = 400 ;//Temperature of entering O2 -[K]
17 T_out_CO2 = 300 ;// Temperature of exiting CO2 -[K]
18 T_out_O2 = 300 ;// Temperature of exiting O2 -[K]
19
20 // Additional data are obtained fro Appendix D and E
    , according to book it is as follows
21 H_fCO = -110.520 ;// Heat of formation of CO - [kJ/g
    mol]
22 H_fO2 = 0 ;// Heat of formation of O2 - [kJ/g mol]
23 H_fCO2 = -393.250 ;// Heat of formation of CO2 - [kJ
    /g mol]
24
25 H_CO = 0 ;// Change in enthalpy during temperature
    change from 298K to 298 K of CO - [kJ/g mol]
26 H1_O2 = 11.619 ;// Change in enthalpy during
    temperature change from 298K to 400 K of input O2
    - [kJ/g mol]
27 H_CO2 = 11.644 ;// Change in enthalpy during
    temperature change from 298K to 300 K of CO2 - [
    kJ/g mol]

```

```

28 H2_O2 = 8.389 ;// Change in enthalpy during
    temperature change from 298K to 300 K of output
    O2 - [kJ/g mol]
29
30 H_in = (H_fCO + H_CO)*m_CO + (H_fO2 + H1_O2)*m1_O2
    ;// Enthalpy change for inputs -[kJ]
31 H_out = (H_fCO2 + H_CO2)*m_CO2 + (H_fO2 + H2_O2)*
    m2_O2 ;// Enthalpy change for inputs -[kJ]
32 del_H = H_out - H_in ;// Net enthalpy change of
    process -[kJ]
33
34 printf('Heat of above reaction is %.1f kJ.',del_H)
    ;

```

Scilab code Exa 25.10 Heating Value of Coal

```

1 clear ;
2 clc;
3 // Example 25.10
4 printf('Example 25.10\n\n');
5 //page no. 788
6 // Solution
7
8 // Given
9 Ex_hv = 29770.0 ;// Experimental heating value of
    given coal - [kJ/kg]
10 // Coal analysis
11
12 C = 71.0/100 ;//Fraction of C in coal
13 H2 = 5.6/100 ;// Fraction of H2 in coal
14 N2 = 1.6/100 ;// Fraction of N2 in coal
15 S = 2.7/100 ;// Fraction of S in coal
16 ash = 6.1/100 ;// Fraction of ash in coal
17 O2 = 13.0/100 ;//Fraction of O2 in coal
18

```

```

19 //Higher heating value (HHV) by Dulong formula
20 HHV = 14544*C + 62028*(H2 - O2/8) + 4050*S ;//
    Higher heating value (HHV) by Dulong formula -[
    Btu/lb]
21 HHV_SI = HHV *1.055/0.454 ;// HHV in SI unt - [kJ/kg
    ]
22
23 printf('The experimental heating value -
    %.0f kJ.\n',Ex_hv) ;
24 printf(' Higher heating value (HHV) by Dulong
    formula - %.0f kJ.\n',HHV_SI) ;
25 printf('\n The two values are quite close.' ) ;

```

Scilab code Exa 25.11 Selecting a Fuel to reduce SO2 emissions

```

1 clear ;
2 clc ;
3 // Example 25.11
4 printf('Example 25.11\n\n');
5 //page no. 789
6 // Solution
7
8 // Given
9 H_req = 10^6 ;// Heat requirement - [Btu]
10
11 d_N6 = 60.2 ;// Density of fuel no. 6-[lb/ft^3]
12 d_N2 = 58.7 ;// Density of fuel no. 2-[lb/ft^3]
13 S_N6 = 0.72/100 ;// Sulphur content in fuel no. 6
14 S_N2 = 0.62/100; //Sulphur content in fuel no. 2
15 lhv_N6 = 155000 ;//Lower heating value of No.6 -[
    Btu/gal]
16 lhv_N2 = 120000 ;//Lower heating value of No.2 -[
    Btu/gal]
17
18 S1 = H_req*d_N6*S_N6/lhv_N6 ;// Sulphur emmited when

```

```
    we use fuel NO. 6-[lb]
19 S2 = H_req*d_N2*S_N2/lhv_N2 ;// Sulphur emmited when
    we use fuel NO. 2-[lb]
20
21 printf('\n Sulphur emmited when we use fuel NO. 6 is
    %.2f lb.',S1 ) ;
22 printf('\n Sulphur emmited when we use fuel NO. 2 is
    %.2f lb.\n',S2 ) ;
23 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

```
1 clear ;
2 clc;
3 // Example 26.1
4 printf('Example 26.1\n\n');
5 //page no. 804
6 // Solution
7
8 printf('Table to carry out degree of freedom
      analysis:\n');
9 // Number of variables involved
10 printf('\nI. Number of variables involved.\n');
11 printf('    Species in F1
      1 \n');
12 printf('    Species in F2
```



```

    2\n');
13 printf('    Specie in F3

    5\n');
14 printf('    Total stream flows

    3\n');
15 printf('    Stream temperatures

    3\n');
16 printf('    Stream pressures

    3 \n');
17 printf('    Q

    1 \n');
18 printf('    Extent of reactions

    2\n');
19 printf('\
-----\n');
20 printf('    Total

    20\n');
21 printf('\n\nII. Number of equations and
    specifications.\n');
22 printf('    Independent species material balances
                                                6\n');
    ;
23 printf('    Sum of species in each of the two
    streams
                                                2
    \n');
24 printf('    Energy balance

    1\n');
25 printf('    Total stream flows

```

```

26     2\n');
printf('    Species values(CO)

27     1\n');
printf('    Pressures

28     3 \n');
printf('    Temperatures

29     2 \n');
printf('    O2 to N2 ratio specified in F2

30     1 \n');
printf('    Complete reaction , hence the extent of
    reaction is implied to both reactions      2\n');
31 printf('\
-----
\n');
32 printf('    Total

33 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

```

1 clear ;
2 clc;
3 // Example 26.2
4 printf('Example 26.2\n\n');
5 //page no. 808
6 // Solution
7
8 // Given

```

```

9 // The main reaction is CO(g,1 atm,100 C) + (1/2)O2(
    g,1 atm,100 C)  --> CO2(g,1at,T K)
                                (A)
10 // Input compounds
11 m1_CO = 1; // Moles of CO input - [g mol]
12 m1_O2 = 1 ;// Moles of O2 input - [g mol]
13 m1_N2 = 3.76 ;// Moles of N2 input - [g mol]
14 //Output compounds
15 m2_CO2 = 1 ;// Moles of CO2 output - [g mol]
16 m2_O2 = .50 ;// Moles of O2 output - [g mol]
17 m2_N2 = 3.76 ;// Moles of N2 output - [g mol]
18
19 // Additional data is obtained from Appendix D,
    according to book it is as follows:
20 // Inputs
21 H1_fCO = -110520 ;// Heat of formation of CO - [J/g
    mol]
22 H1_fO2 = 0 ;// Heat of formation of O2 - [J/g mol]
23 H1_fN2 = 0 ;// Heat of formation of N2 - [J/g mol]
24 H1_CO = 2917 - 728 ;// Change in enthalpy during
    temperature change from 298K to 373 K of CO - [J/
    g mol]
25 H1_O2 = 2953 - 732 ;// Change in enthalpy during
    temperature change from 298K to 373 K of input O2
    - [J/g mol]
26 H1_N2 = 2914 - 728 ;// Change in enthalpy during
    temperature change from 298K to 373 K of input N2
    - [J/g mol]
27
28 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]
29
30 //Outputs - Assume it to be at 2000 K
31 H2_fCO2 = -393510 ;// Heat of formation of CO2 - [J/
    g mol]
32 H2_fO2 = 0 ;// Heat of formation of O2 - [J/g mol]
33 H2_fN2 = 0 ;// Heat of formation of N2 - [J/g mol]

```

```

34 H2_CO2 = 92466 - 912 ;// Change in enthalpy during
    temperature change from 298K to 2000 K of CO2 - [
    J/g mol]
35 H2_O2 = 59914-732 ;// Change in enthalpy during
    temperature change from 298K to 2000 K of output
    O2 - [J/g mol]
36 H2_N2 = 56902 - 728 ;// Change in enthalpy during
    temperature change from 298K to 2000 K of output
    O2 - [J/g mol]
37
38 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]
39
40 del_H1 = H1_out - H_in ;// Net enthalpy change of
    process -[J]
41
42 //Output- Assume it to be at 1750 K
43 H2_fCO2 = -393510 ;// Heat of formation of CO2 - [J/
    g mol]
44 H3_CO2 = 77455 - 912 ;// Change in enthalpy during
    temperature change from 298K to 1750 K of CO2 - [
    J/g mol]
45 H3_O2 = 50555 -732 ;// Change in enthalpy during
    temperature change from 298K to 1750 K of output
    O2 - [J/g mol]
46 H3_N2 = 47940 - 728 ;// Change in enthalpy during
    temperature change from 298K to 1750 K of output
    O2 - [J/g mol]
47
48 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]
49
50 del_H2 = H2_out - H_in ;// Net enthalpy change of
    process -[J]
51
52 printf('Heat of above reaction when output is

```

```

    assumed to be at 2000 K is %.0f J.\n',del_H1) ;
53 printf(' Heat of above reaction when output is
    assumed to be at 1750 K is %.0f J.\n',del_H2) ;
54 // Energy balance here reduce to del_H = 0
55 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') ;
56 // Use interpolation to get the theoretical flame
    temperature
57 del_H = 0 ;// Required condition
58 Ft = 1750 + ((del_H - del_H2)/(del_H1 - del_H2))
    *(2000 - 1750) ;// Interpolation to get Flame
    temperature(Ft)-[K]
59 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 More than one Reaction Occurs

```

1 clear ;
2 clc;
3 // Example 26.3
4 printf('Example 26.3\n\n');
5 //page no. 811
6 // Solution Fig E26.3b
7
8 // Given
9 v_CH4 = 1000 ;// Volume of CH4 taken - [ cubic feet]
10 CH4 = 1 ;// assumed for convenience- [ g mol]
11 ex_air = .5 ;// Fraction of excess O2 required
12 hp_CaCO3 = 0.130 ;// Heat capacity of CaCO3 -[kJ/g
    mol]
13 hp_CaO = 0.062 ;// Heat capacity of CaO -[kJ/g mol]
14 w_CaCO3 = 100.09 ;// Mol. wt. of CaCO3 -[g]

```

```

15 w_CaO = 56.08 ;// Mol. wt. ofCaO - [g]
16
17 // The main reaction are -
18 // (a) CaCO3(s,25 C) --> CaO(s,900 C) + CO2(g,500 C
19 // (b) CH4(g,25 C) + 2O2(g,25 C) --> CO2 (g,500 C)
20 // + 2H2O (g,500 C)
21 req_O2 = 2 ;// By eqn. (b), O2 required by CH4 - [g
22 // mol]
23 ex_O2 = ex_air*req_O2 ;// Excess O2 required - [ g
24 // mol]
25 O2 = req_O2 + ex_O2 ;// Total O2 entering - [ g mol]
26 N2 = O2 *(.79/.21) ;// Total N2 entering - [ g mol]
27
28 // By analysis DOF is zero.
29
30 // Carry out elemental balance to get the unknowns
31 nG_N2 = N2 ;// N2 balance - [ g mol]
32 nG_H2O = 4*CH4/2 ;// H2O balance - [ g mol]
33 nG_O2 = ex_O2 ;// [g mol]
34 // L = P , from Ca balance ...eqn. (A)
35 // 1 + L = nG_CO2 , from C balance ...eqn. (B)
36 // 3L + 2*O2 = 2*nG_CO2 + 2*nG_O2 + nG_H2O + P ,
37 // from O balance ... eqn. (C)
38
39 // For energy balance, get required data from
40 // software in the CD of book and sensible heat data
41 // from Appendix F
42 // given data of outputs is taken in array in order
43 // CO2(g), O2(g),N2(g),H2O(g) and then CaO(s)
44 del_Hi_out = [ -393.250,0,0,-241.835,-635.6] ;// //
45 // Heat of formation - [kJ/g mol]
46 del_Hf_out = [21.425,15.043,14.241,17.010,54.25] ;//
47 // Change in enthalpy during temperature change -[kJ
48 // /g mol]
49 del_H_out =del_Hi_out + del_Hf_out ;// Change in
50 // enthalpy final - [kJ/g mol]

```

```

41
42 // given data of inputs is taken in array in order
    CH4(g), CaCO3(s), O2(g) and N2(g)
43 del_Hi_in = [ -49.963, -1206.9, 0, 0] ;// // Heat of
    formation - [kJ/g mol]
44 del_Hf_in = [0, 0, 0, 0] ;// Change in enthalpy during
    temperature change -[kJ/g mol]
45 del_H_in = del_Hi_in + del_Hf_in ;// Change in
    enthalpy final - [kJ/g mol]
46 // Now do energy balance , assume Q = 0 ,
47 // 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)
48 // Solve eqn. (A), (B), (C), and (D) to get L ,P ,
    nG_CO2
49 a = [1 -1; (del_H_in(2)-del_H_out(5)) -del_H_out(1)]
    ;// Matrix of coefficients
50 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
51 x = a\b ;// Matrix of solutions , L = x(1), nG_CO2 =
    x(2)
52 g_CaCO3 = x(1)*w_CaCO3 ;//CaCO3 processed for each g
    mol of CH4 - [g]
53 printf(' CaCO3 processed for each g mol of CH4 is %
    .0 f g.\n', g_CaCO3) ;
54 m_CaCO3 = (v_CH4*g_CaCO3)/359.05 ;
55 printf(' Therefore , CaCO3 processed per 1000 ft^3 of
    CH4 is %.0 f lb.\n', m_CaCO3) ;

```

Scilab code Exa 26.4 Application of general Energy Balance in a Process Composed of Multiple Units

```
1 clear;
```

```

2  clc;
3  // Example 26.4
4  printf('Example 26.4\n\n');
5  //page no. 815
6  // Solution Fig E26.4b
7
8  // Given
9  SO2_in = 2200 ;// Amount of SO2 entering reactor 2-[
    lb mol/hr]
10 // Basis : 1 lb mol CO entering reactor 1, therefore
11 R1_CO_in = 1 ;//CO entering reactor 1-[lb mol]
12 air = .80 ;// Fraction of air used in burning
13
14 // System- reactor 2
15 // Given
16 R2_fSO2_in = 0.667 ;// Fraction of SO2 entering
    reactor 2
17 R2_fO2_in = 0.333 ;// Fraction of O2 entering
    reactor 2
18 R2_fSO3_out = 0.586 ;// Fraction of SO3 exiting
    reactor 2
19 R2_fSO2_out = 0.276 ;// Fraction of SO2 exiting
    reactor 2
20 R2_fO2_out = 0.138 ;// Fraction of O2 exiting
    reactor 2
21 // Main Reaction: CO , (1/2)*O2 ----> CO2
22 R1_O2_in = (1/2)*air ;// O2 entering reactor 1-[g
    mol]
23 R1_N2_in = R1_O2_in*(79/21) ;// N2 entering reactor
    1-[g mol]
24
25 //Output of reactor 1
26 R1_CO_out = R1_CO_in*(1 - air) ;// [g mol]
27 R1_CO2_out = 1*( air) ;// [g mol]
28 R1_N2_out = R1_N2_in ;//[g mol]
29
30 // By analysis DOF is zero.
31 // Get eqn. to solve by species balance

```



```

32 //Unknowns - P- exit stream of reactor 2 , F - entry
    stream of reactor 2 , ex - extent of reaction
33 // P*(R2_fSO2_out) - F*0 = 1*ex ... eqn.(a)- By SO3
    balance
34 // P*(R2_fSO2_out) - F*(R2_fSO2_in) = -1*ex ... eqn.(
    b) - By SO2 balance
35 // By O2 balance we will get eqn. equivalent to eqn.
    (b), so we need one more eqn.
36
37 // Energy balance
38 // For energy balance, get required data from
    software in the CD of book and sensible heat data
    from Appendix F
39 // given data of outputs is taken in array in order
    CO(g),CO2(g), N2(g),SO2(g),SO3(g) and then O2(g)
40 del_Hi_out = [
    -109.054,-393.250,0,-296.855,-395.263,0] ; //
    Heat of formation - [kJ/g mol]
41 del_Hf_out =
    [35.332,35.178,22.540,20.845,34.302,16.313] ;//
    Change in enthalpy during temperature change -[kJ
    /g mol]
42 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]
43
44 // given data of inputs is taken in array in order
    CO(g),CO2(g), N2(g),SO2(g) and then O2(g)
45 del_Hi_in = [ -109.054,-393.250,0,-296.855,0] ;//
    // Heat of formation - [kJ/g mol]
46 del_Hf_in = [17.177,17.753,11.981,0,0] ;//Change in
    enthalpy during temperature change -[kJ/g mol]
47 del_H_in = del_Hi_in+ del_Hf_in ;// Change in
    enthalpy final - [kJ/g mol]
48 // Now do energy balance , assume Q = 0 ,
49 // 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)

```

```

50
51 // Solve eqn. (a), (b) and (c) to get F ,P , ex
52 a = [(R2_fS03_out) 0 -1;(R2_fS02_out) -(R2_fS02_in)
      1;(del_H_out(4)*R2_fS02_out + del_H_out(5)*
      R2_fS03_out + del_Hi_out(6)*R2_fO2_out ) -(
      del_H_in(4)*R2_fS02_in) 0] ;// Matrix of
      coefficients
53 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
54 x = a\b ;// Matrix of solutions , P = x(1) , F = x(2)
      ,ex = x(3)
55 F = x(2) ;//exit stream of reactor 2 - [lb mol]
56 R2_S02_in = R2_fS02_in*F ;// Moles of SO2 required
      per lb mol of CO - [lb mol]
57 CO = (R1_CO_in*S02_in)/R2_S02_in ;//Mole of CO
      burned in reactor 1 - [lb mol]
58
59 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

```

1 clear ;
2 clc;
3 // Example 26.5
4 printf('Example 26.5\n\n');
5 //page no. 819
6 // Solution
7
8 // Given
9 CA = 10000 ;// Produced citric acid - [kg]
10 f_glucose = .30 ;// Fraction of glucose in solution
11 con_glucose = .60 ;// Fraction of glucose consumed

```

```

12 w_glucose = 180.16 ;// Mol. wt. of d,alpha glucose
    -[g]
13 H_glucose = -1266 ;// Specific enthalpy change of
    glucose - [kJ/g mol]
14 w_CA = 192.12; // Mol. wt. of citric acid -[g]
15 H_CA = -1544.8 ;// Specific enthalpy change of
    citric acid - [kJ/g mol]
16 w_BM = 28.6 ;// Mol. wt. of biomass -[g]
17 H_BM = -91.4 ;// Specific enthalpy change of
    biomass - [kJ/g mol]
18 H_CO2 = -393.51 ;// Specific enthalpy change of
    CO2 - [kJ/g mol]
19
20 // Main reaction is :
21 // 3 * glucose + 7.8*O2 ----> 5.35*BM + 2.22*CA +
    4.50*CO2 ..reaction (a)
22
23 // Material Balance
24 mol_CA = CA/w_CA ;// Mole of citric acid produced -
    [kg mol]
25 g_soln = (mol_CA*(3/2.22)*w_glucose*1)/(con_glucose*
    f_glucose) ;//Mass of 30 % glucose solution
    introduced -[kg]
26 i_glucose = g_soln* f_glucose / w_glucose ;//
    Initial moles of glucose - [kg mol]
27 f_glucose = (1 - con_glucose)*i_glucose ;// Final
    moles of glucose - [kg mol]
28 f_CA = mol_CA ;// Final moles of citric acid - [kg
    mol]
29 f_BM = f_CA*(5.35/2.22) ;// Using the reaction (a)-
    Final moles of biomass - [kg mol]
30 i_O2 = i_glucose*(7.8/3) ;// Using the reaction (a)
    - Initial moles of O2 - [kg mol]
31 f_CO2 = i_glucose*(4.5/3)*con_glucose ;// Using
    the reaction (a) - Final moles of CO2 - [kg mol]
32
33 // Energy balance
34 // For closed system - del_U = Q + W

```

```

35 power = 100 ;// Power of aerator -[hp]
36 time = 220 ;// Time taken for reaction - [ hr ]
37 W = (power*745.7*time*3600)/1000 ;// Work done by
    aerator - [kJ]
38
39 // Assume del_U = del_H , pv work is equal to zero ,
    hence
40 // Q = del_H - W
41
42 Hi_glucose = i_glucose*H_glucose*1000 ;// Enthalpy
    change of glucose input - [kJ]
43 Hi_O2 = i_O2*0*1000 ;// Enthalpy change of O2 input
    - [kJ]
44 H_in = Hi_glucose + Hi_O2 ;// Enthalpy change of
    input - [kJ]
45
46 Hf_glucose = f_glucose*H_glucose*1000 ;// Enthalpy
    change of glucose output - [kJ]
47 Hf_BM = f_BM * H_BM*1000 ;//Enthalpy change of
    biomass output - [kJ]
48 Hf_CA = f_CA *H_CA*1000 ;//Enthalpy change of citric
    acid output - [kJ]
49 Hf_CO2 = f_CO2 *H_CO2*1000 ;//Enthalpy change of CO2
    output - [kJ]
50 H_out = Hf_glucose + Hf_BM +Hf_CA + Hf_CO2 ;//
    Enthalpy change of output - [kJ]
51 del_H = H_out - H_in ;// Total enthalpy change in
    process - [kJ]
52 Q = del_H - W ;// Heat removed - [kJ]
53
54 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

```
1 clear ;
2 clc ;
3 // Example 27.1
4 printf('Example 27.1\n\n');
5 //page no. 838
6 // Solution E27.1
7
8 // Given
9 V_w = 1 ;// Volume of given water -[L]
10 P_atm = 100 ;// Atmospheric pressure - [kPa]
11
12 //W = -p*del_V
13 V_H2O = 0.001043 ;// Specific volume of water from
    steam table according to book- [cubic metre]
14 V_vap = 1.694 ;// Specific volume of vapour from
    steam table according to book- [cubic metre]
15 V1 = 0 ;// Initial volume of H2O in bag-[cubic metre
    ]
```

```

16 V2 = (V_w*V_vap)/(1000*V_H2O) ;// Final volume of
    water vapour -[cubic metre]
17 W = -P_atm*(V2 -V1)* 1000 ;// Work done by saturated
    liquid water -[J]
18
19 printf(' Work done by saturated liquid water is %.3e
    J.\n',W) ;

```

Scilab code Exa 27.2 Calculation of Work in a Batch Process

```

1 clear ;
2 clc ;
3 // Example 27.2
4 printf('Example 27.2\n\n');
5 //page no. 840
6 // Solution E27.2
7
8 // Given
9 m_N2 = 1 ;// Moles of N2 taken -[kg mol]
10 p = 1000; // Pressure of cylinder -[kPa]
11 T = 20 + 273 ;// Temperature of cylinder -[K]
12 a_pis = 6 ;// Area of piston - [square centimetre]
13 m_pis = 2 ;// Mass of piston - [kg]
14 R = 8.31 ;// Ideal gas constant - [(kPa*cubic metre)
    /(K * kmol)]
15
16 V = (R*T)/p ;// Specific volume of gas at initial
    stage -[cubic metre/kg mol]
17 V1 = V * m_N2 ;// Initial volume of gas - [cubic
    metre]
18 V2 = 2*V1 ;// Final volume of gas according to given
    condition -[cubic metre]
19
20 // Assumed surrounding pressure constant = 1 atm
21 p_atm = 101.3 ;// Atmospheric pressure -[kPa]

```

```

22 del_Vsys = V2 -V1 ;// Change in volume of system -[
    cubic metre]
23 del_Vsurr = - del_Vsys ;// Change in volume of
    surrounding -[cubic metre]
24 W_surr = -p_atm*del_Vsurr ;// Work done by
    surrounding - [kJ]
25 W_sys = -W_surr ;// Work done by system - [kJ]
26
27 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

```

1 clear ;
2 clc;
3 // Example 27.3
4 printf('Example 27.3\n\n');
5 //page no. 845
6 // Solution
7
8 // Given
9 p_plant = 20 ;// Power generated by plant -[MW]
10 h = 25 ;// Height of water level - [m]
11 V = 100 ;// Flow rate of water -[cubic metre/s]
12 d_water = 1000 ;// Density of water - [ 1000 kg /
    cubic metre]
13 g = 9.807 ;// Acceleration due to gravity -[m/square
    second]
14
15 M_flow = V*d_water ;// Mass flow rate of water -[kg/
    s]
16 del_PE = M_flow*g*h ;// Potential energy change of
    water per second -[W]
17 eff = (p_plant*10^6) /(del_PE) ;// Efficiency of

```

```

    plant
18
19 printf(' Efficiency of plant is %.2f .\n',eff) ;

```

Scilab code Exa 27.4 Calculation of Plant Efficiency

```

1 clear ;
2 clc;
3 // Example 27.4
4 printf('Example 27.4\n\n');
5 //page no. 845
6 // Solution Fig.E27.4
7
8 // Given
9 LHV = 36654 ;// LHV value of fuel - [kJ/ cubic metre
    ]
10 Q1 = 16 ;//- [kJ/ cubic metre]
11 Q2 = 0 ;//- [kJ/ cubic metre]
12 Q3 = 2432 ;//- [kJ/ cubic metre]
13 Q4 = 32114 ;//- [kJ/ cubic metre]
14 Q41 = 6988 ;//- [kJ/ cubic metre]
15 Q8 = 1948 ;//- [kJ/ cubic metre]
16 Q9 = 2643 ;//- [kJ/ cubic metre]
17 Q81 = 2352 - Q8 ;// - [kJ/ cubic metre]
18 Q567 = 9092 ;// Sum of Q5, Q6 and Q7- [kJ/ cubic
    metre]
19
20 //(a)
21 G_ef = (LHV+ Q1 +Q2 + Q3 - Q9)/(LHV) ;// Gross
    efficiency
22 printf('(a) Gross efficiency is %.3f .\n',G_ef) ;
23
24 //(b)
25 T_ef = (Q567+Q8)/(LHV+ Q1 +Q2 + Q3) ;//Thermal
    efficiency

```



```

26 printf(' (b) Thermal efficiency is %.3f .\n',T_ef) ;
27
28 //(c)
29 C_ef = Q4/(Q4 + Q41) ;// Combustion efficiency
30 printf(' (c) Combustion efficiency is %.3f .\n',C_ef
    ) ;

```

Scilab code Exa 27.5 Comparison of the Reversible Work for a Batch Process with that of a Flow Operating under Same Conditions

```

1 clear ;
2 clc ;
3 // Example 27.5
4 printf('Example 27.5\n\n');
5 //page no. 850
6 // Solution
7
8 // Given
9 V1 = 5 ;// Volume of gas initially - [cubic feet]
10 P1 = 1 ;// Initial pressure - [atm]
11 P2 = 10 ;// Final pressure - [atm]
12 T1 = 100 + 460 ;// initial temperature - [degree
    Rankine]
13 R = 0.7302 ;// Ideal gas constant -[(cubic feet*atm)
    /(lb mol)*(R)]
14 //Equation of state  $pV^{1.4} = \text{constant}$ 
15
16 //(a)
17 //Energy balance reduces to  $\text{del}_E = \text{del}_U = \text{del}_W$ 
18 V2 = V1*(P1/P2)^(1/1.4) ;// Final volume - [cubic
    feet]
19 W1_rev = integrate('-(P1)*(V1/V)^(1.4)', 'V', V1, V2) ;
    // Reversible work done in compression in a
    horizontal cylinder with piston -[cubic feet *atm
    ]

```

```

20 W1 = W1_rev *1.987/.7302 ;// Conversion to Btu -[Btu
    ]
21
22 printf('\n (a)Reversible work done in compression in
    a horizontal cylinder with piston is %.1f Btu .\
    n ',W1);
23
24 //(b)
25 n1 = (P1*V1)/(R*T1) ;// Number of moles of gas
26 W2_rev = integrate('(V1)*(P1/P)^(1/1.4)', 'P',P1,P2)
    ;// Reversible work done in compresion in a
    rotary compressor -[cubic feet *atm]
27 W2 = W2_rev *1.987/.7302 ;// Conversion to Btu -[Btu
    ]
28
29 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 of Water

```

1 clear ;
2 clc;
3 // Example 27.6
4 printf('Example 27.6\n\n');
5 //page no. 853
6 // Solution
7
8 // Given
9 m_water = 1 ;// Mass flow rate of water -[lb/min]
10 P1 = 100 ;// Initial pressure - [psia]
11 P2 = 1000 ;// Final pressure - [psia]
12 T1 = 80 + 460 ;// initial temperature - [degree
    Rankine]
13 T2 = 100 + 460 ;// final temperature - [degree

```

```

Rankine]
14 h = 10 ;// Difference in water level between entry
    and exit of stream-[ft]
15 g = 32.2 ;// Accleration due to gravity - [ft/
    square second]
16 gc = 32.2 ;//[(ft*lbm)/(lbf*quare second)]
17
18 // The mechanical energy balance reduces to W =
    PV_work + del_PE ....(A)
19 // From steam table , specific volume of liquid
    water at 80 and 100 degree F is noted , according
    to book it is as follows-
20 v1 = .01607 ;// specific volume of liquid water at
    80 degree F -[cubic feet/lbm]
21 v2 = .01613 ;// specific volume of liquid water at
    100 degree F -[cubic feet/lbm]
22 // But for pratical purposes wwater is taken to be
    incompressible and specific volume can be taken
    as v, ith following value
23 v= 0.0161 ;// -[cubic feet/lbm]
24
25 del_PE = (h*g)/(gc*778) ;// Change in potential
    energy - [Btu/lbm]
26 PV_work = integrate('(v)*(12^2/778)', 'P', P1, P2) ;//
    PV work done -[Btu/lbm]
27 //From eqn. (A)
28 W = PV_work + del_PE ;// Work per minute required to
    pump 1 lb water per minute - [Btu/lbm]
29
30 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

```
1 clear ;
2 clc ;
3 // Example 28.1
4 printf('Example 28.1\n\n');
5 //page no. 869
6 // Solution
7
8 // Given
9 Ref_T = 77 ;//Reference temperature -[degree F]
10
11 //(a)
12 mol_NH3 = 1 ;// Moles of NH3 - [lb mol]
13 mw_NH3 = 17 ;//Molecular t. of NH3 -[lb]
14 mw_H2O = 18 ;//Molecular t. of H2O -[lb]
15 f1_NH3 = 3/100 ;// Fraction of NH3 in solution
16 m_H2O = (mw_NH3/f1_NH3) - mw_NH3 ;// Mass of water
    in solution -[lb]
17 mol_H2O = m_H2O/mw_H2O ;// Moles of H2O in solution
    -[lb mol]
18
19 printf('(a) Moles of H2O in solution is %.1f lb
```

```

    mol .\n ',mol_H2O);
20 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 ');
21 H_soln = -14800 ;// From table given in question in
    book -[Btu/lb mol NH3]
22 printf('      The amount of cooling needed is , %.0f
    Btu heat removed.\n ',abs(H_soln));
23
24 //(b)
25 V = 100 ;// Volume of solution produced -[gal]
26 f2_NH3 = 32/100 ;// Fraction of NH3 in solution
27 // From Lange's Handbook of chemistry additional
    data is obtained , according to book it is as
    follows -
28 sg_NH3 = .889 ;// Specific gravity of NH3
29 sg_H2O = 1.003 ;// Specific gravity of H2O
30 d_soln = sg_NH3*62.4*sg_H2O*100/7.48 ;// Density of
    solution - [lb / 100 gal]
31 NH3 = d_soln*f2_NH3/mw_NH3 ;// Mass of NH3 - [ lb
    mol/ 100 gal]
32 m1_H2O = (mw_NH3/f2_NH3) - mw_NH3 ;// Mass of water
    in solution -[lb]
33 mol1_H2O = m1_H2O/mw_H2O ;// Moles of H2O in
    solution -[lb mol]
34
35 printf('\n (b) Moles of H2O in solution is %.1f lb
    mol .\n ',mol1_H2O);
36 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 ');
37 H_soln = -13700 ;// From table given in question in
    book -[Btu/lb mol NH3]
38 total_H = abs(NH3*H_soln) ;// Total heat removed
    from solution -[Btu]
39 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

```
1 clear ;
2 clc;
3 // Example 28.2
4 printf('Example 28.2\n\n');
5 //page no. 872
6 // Solution
7
8 // Given
9 p = 100 ;// Mass of product - [kg]
10 f_HCl = 25/100 ;//Fraction of HCl in product
11 //Product analysis
12 HCl = f_HCl*p ;// Mass of HCl in product - [kg]
13 H2O = (1-f_HCl)*p ;// Mass of H2O in product -[kg]
14 mw_HCl = 36.37 ;// Molecular weight of HCl -[kg]
15 mw_H2O = 18.02 ;// Molecular weight of H2O -[kg]
16 mol_HCl = HCl /mw_HCl ;// Moles of HCl - [kg mol]
17 mol_H2O = H2O /mw_H2O; // Moles of H2O - [kg mol]
18 total_mol = mol_HCl + mol_H2O ;// Total no. of moles
    -[kg mol]
19 mf_HCl = mol_HCl / total_mol ;// mole fraction of
    HCl
20 mf_H2O = mol_H2O / total_mol ; // mole fraction of
    H2O
21 mr = mol_H2O/mol_HCl ;// Mole ratio of H2O to HCl
22 MW = mf_HCl*mw_HCl + mf_H2O*mw_H2O ;// Molecular t.
    of solution -[kg]
23 Ref_T = 25 ;//Reference temperature -[degree C]
24
25 // Energy balance reduces to  $Q = \Delta H$ 
26 // Additional data is obtained from Table E.1 ,
    according to book it is as follows -
27 mol1_HCl = total_mol ;// Moles of HCl // Moles of
```

```

    HCl output -[g mol]
28 Hf1_HCl = -157753 ;// Heat of formation of HCl
    output-[J/ g mol HCl ]
29 Hf_HCl = -92311 ;// Heat of formation of HCl input-[
    J/ g mol HCl ]
30 Hf_H2O = 0 ;// Heat of formation of H2O input-[J/ g
    mol HCl ]
31 H1_HCl = 556 ;// Change in enthalpy during
    temperature change from 25 C to 35 C of HCl - [J/
    g mol]
32 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]
33
34 H_H2O = 0 ;// Change in enthalpy during temperature
    change from 25 C to 25 C of H2O - [J/g mol]
35
36 H_in = (Hf_HCl + H_HCl)*mol_HCl + (Hf_H2O + H_H2O)*
    mol_H2O ;// Enthalpy change of input -[J]
37 H_out = Hf1_HCl*mol_HCl +H1_HCl*mol1_HCl ;//
    Enthalpy change of output -[J]
38
39 del_H = H_out - H_in ;// Net enthalpy change n
    process - [J]
40 Q = del_H; // By energy balance - [J]
41
42 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

```

1 clear ;
2 clc;
3 // Example 28.3

```

```

4 printf('Example 28.3\n\n');
5 //page no. 875
6 // Solution fig. 28.3
7
8 // Given
9 //Input analysis
10 soln1 = 600 ; // Mass flow rate of entering solution
    1 -[lb/hr]
11 c1_NaOH = 10/100 ;// Fraction of NaOH in entering
    solution 1
12 T1 = 200 ;// Temperature at entry
13 soln2 = 400 ;// Mass flow rate of another solution 2
    entering -[lb/hr]
14 c2_NaOH = 50/100 ;// Fraction of NaOH in another
    entering solution 2
15
16 // 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)
17 F = soln1 + soln2; // Mass flow rate of final
    solution - [lb/hr]
18
19 // Material balance to get composition of final
    solution
20 F_NaOH = c1_NaOH * soln1 + c2_NaOH * soln2 ;// Mass
    of NaOH in final solution -[lb]
21 F_H2O = F - F_NaOH ;// Mass of H2O in final solution
    -[lb]
22
23 // Enthalpy data from H-x chart , according to book
    it is as follows
24 H_soln1 = 152 ;// Specific enthalpy change for
    solution 1-[Btu/lb]
25 H_soln2 = 290 ;// Specific enthalpy change for
    solution 2-[Btu/lb]
26
27 // Energy balance

```



```

28 H_F = (soln1*H_soln1 + soln2*H_soln2)/F ;// Specific
    enthalpy change for final solution -[Btu/lb]
29
30 //(a)
31 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');
32
33 //(b)
34 cF = F_NaOH*100/F; // Concentration of final
    solution -[wt % NaOH ]
35 printf(' (b) The concentration of final solution is
    %.0f wt.%% NaOH . \n',cF);
36
37 //(c)
38 // 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
39 x = (F*H_F - F*175)/(1158 - 175) ;// H2O evaporated
    per hour -[lb]
40
41 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

```
1 clear ;
2 clc;
3 // Example 29.1
4 printf('Example 29.1\n\n');
5 //page no. 895
6 // Solution fig. E29.1
7
8 // Given
9 DBT = 90 ;// Dry bulb temperature - [degree F]
10 WBT = 70 ;// Wet bulb temperature - [degree F]
11
12 //Get point A using DBT & WBT. Following information
    is obtained from humidity chart , fig. E29.1
13
14 printf('(a) The Dew point is located at point B or
    about 60 degree F, using constant humidity line.\n
    n');
15 printf(' (b) By interpolation between 40%% and 30%%
```

```

    RH , you can find point A is at 37%% relative
    humidity .\n');
16 printf(' (c) You can read humidity from the
    righthand ordinate as 0.0112 lb H2O/lb dry air .\n
    n');
17 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');
18 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

```

1 clear ;
2 clc ;
3 // Example 29.2
4 printf('Example 29.2\n\n');
5 //page no. 897
6 // Solution fig. E29.2
7
8 // Given
9 DBT1 = 38 ;// Initial dry bulb temperature - [degree
    C]
10 DBT2 = 86 ;// Final dry bulb temperature - [degree C
    ]
11 RH1 = 49 ;// Relative humidity - [%]
12
13 //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
14
15 printf('The Dew point is unchanged in the process
    because humidity is unchanged, and it is located

```

```

    at 24.8 degree C.\n');
16
17 // Additional data is obtained from humidity chart ,
    according to book data is as follows
18 A_Hsat = 90.0 ;// Enthalpy of saturation at point A-
    [kJ/kg]
19 A_dH = -0.5 ;//Enthalpy deviation -[kJ/kg]
20 A_Hact = A_Hsat + A_dH ;// Actual enthalpy at point
    A -[kJ/kg]
21 B_Hsat = 143.3 ;// Enthalpy of saturation at point B
    - [kJ/kg]
22 B_dH = -3.3 ;//Enthalpy deviation -[kJ/kg]
23 B_Hact = B_Hsat + B_dH ;// Actual enthalpy at point
    B -[kJ/kg]
24
25 // Energy balance reduces to Q = del_H
26 del_H = B_Hact - A_Hact ;// Total change in enthalpy
    - [kJ/kg]
27 v = 0.91 ;// Specific volume of moist air at point A
    -[cubic metre / kg]
28 Q = del_H/v ;// Heat added per cubic metre of inital
    moist air -[kJ]
29 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

```

1 clear ;
2 clc;
3 // Example 29.3
4 printf('Example 29.3\n\n');
5 //page no. 898
6 // Solution fig. E29.3b
7
8 // Given

```

```

9 DBT1 = 40 ;// Initial dry bulb temperature - [degree
  C]
10 DBT2 = 27 ;// Final dry bulb temperature - [degree C
  ]
11
12 // Process is assumed to be adiabatic , therefore
  wet bulb temperature is constant
13 WBT1 = 22 ;// Initial wet bulb temperature - [degree
  C]
14 WBT2 = WBT1 ;// Final wet bulb temperature - [degree
  C]
15
16 //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
17 H_B = 0.0145 ;// Humidity at point B -[kg H2O/kg dry
  air]
18 H_A = 0.0093 ;// Humidity at point A -[kg H2O/kg dry
  air]
19 Diff = H_B - H_A ;// Moisture added in kg per
  kilogram of dry air going through humidifier -[kg
  H2O/kg dry air]
20
21 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

```

1 clear ;
2 clc;
3 // Example 29.4
4 printf('Example 29.4\n\n');
5 //page no. 900

```

```

6 // Solution fig. E29.4
7
8 // Given
9 c_b1 = 8.30 * 10^6 ;// Capacity of blower - [cubic
    feet/hr]
10 DBT_A = 80 ;// Initial dry bulb temperature of moist
    air - [degree F]
11 DBT_B = 95 ;// Final dry bulb temperature of exit
    air - [degree F]
12 WBT_A = 65 ;// Initial wet bulb temperature of moist
    air - [degree F]
13 WBT_B = 90 ;// Final wet bulb temperature of exit
    air - [degree F]
14 T1_H2O = 120 ;// Initial temperature of water - [
    degree F]
15 T2_H2O = 90 ;// Final temperature of water - [degree
    F]
16
17 //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
18 H_A = 0.0098; // Humidity of air at A - [lb H2O / lb
    dry air]
19 H1_A = 69 ;// Humidity of air at A - [grains H2O /
    lb dry air]
20 delH_A = 30.05 - 0.12; // Enthalpy of entering air
    -[Btu/lb dry air]
21 v_A = 13.82 ;// Specific volume of entering air -[
    cubic feet/lb dry air]
22 H_B = 0.0297; // Humidity of air at B - [lb H2O / lb
    dry air]
23 H1_B = 208 ;// Humidity of air at B - [grains H2O /
    lb dry air]
24 delH_B = 55.93 - 0.10 ;// Enthalpy of exit air -[
    Btu/lb dry air]
25 v_B = 14.65 ;// Specific volume of exit air -[cubic
    feet/lb dry air]

```

```

26 Eq_A = c_b1 /v_A ;// Entering dry air equivalent of
    capacity of blower -[lb dry air]
27
28 // Reference temperature for water stream is 32
    degree F
29 del_H1_H2O = 1*(T1_H2O - 32) ;//Enthalpy of entering
    water -[Btu/lb H2O]
30 del_H2_H2O = 1*(T2_H2O - 32) ;//Enthalpy of exit
    water -[Btu/lb H2O]
31 tr_H2O = H_B - H_A ;// Transfer of water to air -[lb
    H2O / lb dry air]
32
33 // Energy balance around the entire process yields W
    -
34 W = (delH_B - del_H2_H2O*tr_H2O - delH_A)/(
    del_H1_H2O - del_H2_H2O) ;// Water entering tower
    - [lb H2O/lb dry air]
35 W1 = W - tr_H2O ;// Water leaving tower -[lb H2O/lb
    dry air]
36 Total_W1 = W1* Eq_A ;// Total water leaving tower -[
    lb/hr]
37
38 printf('Amount of water cooled per hour is %.2e lb/
    hr .\n',Total_W1);

```

Scilab code Exa 29.5 Drying of Chlorella

```

1 clear ;
2 clc;
3 // Example 29.5
4 printf('Example 29.5\n\n');
5 //page no. 902
6 // Solution fig. E29.5
7
8 // Given

```

```

 9 W = 100 ;// Amount of entering water -[lb/hr]
10 H1 = .020 ;// Humidity of entering air -[lb H2O / lb
    dry air]
11 T1 = 155 ;//Temperature of entering air -[degree F]
12 DTB = 110 ;// Dry bulb temperature of exit air -[
    degree F]
13 WTB = 100 ;// Wet bulb temperature of exit air -[
    degree F]
14
15 // Additional data is obtained from humidity chart ,
    it is as follows
16 H2 = .0405 ;//Humidity of exit air -[lb H2O / lb dry
    air]
17
18 del_H = H2 - H1 ;// Change in humidity between two
    states -[lb H2O / lb dry air]
19 air_in = (W*1.02)/(del_H * 1) ;// Amount of wet
    air entering -[lb]
20
21 mol_air = 29 ;// Molecular wt. of air -[lb]
22 Ref_T = 32 + 460 ;// Reference temperature - [
    degree R]
23 gi_T = 90 + 460; // Given temperature on which
    calculation is based - [degree R]
24 air = (air_in *359*gi_T)/( mol_air*Ref_T) ;// Air
    consumption of dryer at 90 degree F and 1 atm -[
    cubic feet]
25
26 printf('Air consumption of dryer at 90 degree F and
    1 atm is %.2e cubic feet .\n',air);

```
