

Scilab Textbook Companion for
Introduction To Chemical Engineering
by S. K. Ghoshal, S. K. Sanyal And S. Datta¹

Created by
Himanshu Bhatia
Btech
Chemical Engineering
IIT Guwahati
College Teacher
Dr Prakash Kotecha
Cross-Checked by
Ganesh R

May 24, 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: Introduction To Chemical Engineering

Author: S. K. Ghoshal, S. K. Sanyal And S. Datta

Publisher: Tata McGraw Hill Education Pvt. Ltd., New Delhi

Edition: 1

Year: 2006

ISBN: 0-07-460140-7

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 Introduction	5
2 Physico Chemical Calculations	19
3 Material and Energy Balances	42
4 Flow Of Fluids	69
5 Heat Transfer	83
6 Mass Transfer	96
7 Chemical Kinetics	111
8 Measuring Devices	120
9 Computers and their application	129

List of Scilab Codes

Exa 1.1	Air composition	5
Exa 1.2	Volume calculation	6
Exa 1.3	Gas Composition	6
Exa 1.4	Volume calculation	8
Exa 1.5	Amount of CO2 released	8
Exa 1.6	Vapor pressure	9
Exa 1.7	Duhring Plot calculations	10
Exa 1.8	Vapor Pressure of Mixture	10
Exa 1.9	Vapor pressure	11
Exa 1.10	Flow relation	12
Exa 1.11	Average Velocity	12
Exa 1.12	Velocity determination	13
Exa 1.13	Velocity determination	14
Exa 1.14	Dimensional analysis	15
Exa 1.15	Dimensional analysis	15
Exa 1.16	Dynamic similarity	15
Exa 1.17	Dynamic similarity	16
Exa 1.18	Nomographic chart	17
Exa 1.19	Calculation using Nomograph	18
Exa 2.1	Ideal gas system	19
Exa 2.2	Mixture properties	19
Exa 2.3	Equivalent metal mass	20
Exa 2.4	Purity of Sodium Hydroxide	21
Exa 2.5	Carbon content formulation	22
Exa 2.6	Combustion of gas	22
Exa 2.7	Sulphuric acid preparation	23
Exa 2.8	Molarity Molality Normality Calculation	24
Exa 2.9	Normality calculation	24

Exa 2.10	Precipitation of KClO_3	25
Exa 2.11	Solubility of CO_2	25
Exa 2.12	Vapor pressure calculation	26
Exa 2.13	Boiling point calculation	26
Exa 2.14	Colligative properties	27
Exa 2.15	Huggins Equation	28
Exa 2.16	Molecular Formula	28
Exa 2.17	Molecular Formula	29
Exa 2.18	Molecular Formula	30
Exa 2.19	Molecular Formula	31
Exa 2.20	Metal deposition	31
Exa 2.21	EMF of cell	32
Exa 2.22	EMF of cell	33
Exa 2.23	EMF of cell	33
Exa 2.24	Silver deposition	34
Exa 2.25	Electroplating time	34
Exa 2.26	Water hardness	35
Exa 2.27	Water hardness	35
Exa 2.28	Water hardness	36
Exa 2.29	Mixture composition	37
Exa 2.30	Mixture composition	38
Exa 2.31	Mixture properties	39
Exa 2.32	Humidity	40
Exa 3.1	Coal consumption	42
Exa 3.2	Nitric acid preparation	43
Exa 3.3	HCl production	43
Exa 3.4	Acetylene consumption	44
Exa 3.5	Screen effectiveness	45
Exa 3.6	Absorption	46
Exa 3.7	Extraction	47
Exa 3.8	Distillation	48
Exa 3.9	Distillation	49
Exa 3.10	Crystallization	50
Exa 3.11	crystallization	51
Exa 3.12	Drying	51
Exa 3.13	Conditioning of air	52
Exa 3.14	Ammonia Synthesis	53
Exa 3.15	Enthalpy calculation	54

Exa 3.16	Enthalpy calculation	55
Exa 3.17	Enthalpy of formation	55
Exa 3.18	Combustion	56
Exa 3.19	Heat of reaction	57
Exa 3.20	Heat transfer	57
Exa 3.21	Calorific value	58
Exa 3.22	Coal combustion	58
Exa 3.23	Coal combustion	60
Exa 3.24	Petrol combustion	62
Exa 3.25	Air supply	63
Exa 3.26	CO ₂ cooling	64
Exa 3.27	Heating area	64
Exa 3.28	Distillation column	65
Exa 3.29	Crystallization	67
Exa 3.30	Combustion	68
Exa 4.1	Water compressibility	69
Exa 4.2	Isothermal Compressibility	69
Exa 4.3	Viscosity	70
Exa 4.4	Streamline flow	70
Exa 4.5	Frictional losses	70
Exa 4.6	Velocity profile	71
Exa 4.7	Velocity profile	71
Exa 4.8	Boundary layer	71
Exa 4.9	Pipe flow	72
Exa 4.10	Temperature rise	73
Exa 4.11	Bernoulli equation	74
Exa 4.12	Power requirements	74
Exa 4.13	Hagen Poiseulle equation	75
Exa 4.14	Pressure Head calculation	76
Exa 4.15	Level difference calculation	76
Exa 4.16	Energy cost calculation	77
Exa 4.17	Pressure loss	78
Exa 4.18	Pressure gradient	78
Exa 4.19	Flow rate	79
Exa 4.20	Pipe dimensions	80
Exa 4.21	Pressure gradient	80
Exa 4.22	Minimum fluidization velocity	81
Exa 4.23	Pumping of fluids	82

Exa 5.1	Heat conduction	83
Exa 5.2	Heat conduction	83
Exa 5.3	Heat conduction through sphere	84
Exa 5.4	Composite wall	85
Exa 5.5	Composite Pipeline	86
Exa 5.6	Parellel Resistance	87
Exa 5.7	Heat transfer coefficient	88
Exa 5.8	Heat transfer coefficient	89
Exa 5.9	Earth Temperature	90
Exa 5.10	Earth Temperature	90
Exa 5.11	Equilibrium temperature	91
Exa 5.12	Equilibrium temperature	91
Exa 5.13	Temperature calculation	92
Exa 5.14	Solar constant	92
Exa 5.15	Evaporator	93
Exa 5.16	Evaporator	94
Exa 5.17	Evaporator	94
Exa 6.1	Diffusivity	96
Exa 6.2	Absorption	96
Exa 6.3	Equimolar counter diffusion	97
Exa 6.4	Resistane to diffusion	97
Exa 6.5	Vapor diffusion	98
Exa 6.6	Flux of HCl	98
Exa 6.7	Vaporization	99
Exa 6.8	Gas Absorption	100
Exa 6.9	Equilibrium Composition	101
Exa 6.10	Equilibrium Composition	101
Exa 6.11	Vapor Liquid Equilibrium	102
Exa 6.12	Distillation Column	102
Exa 6.13	Distillation	103
Exa 6.14	Steam Distillation	104
Exa 6.15	Mcabe Thiele Method	104
Exa 6.16	Liquid liquid extraction	105
Exa 6.17	Liquid liquid extraction	106
Exa 6.18	Humidity calculation	106
Exa 6.19	Drying operation	107
Exa 6.20	Crystallization	108
Exa 6.21	Crystallization	109

Exa 7.1	Constant volume reaction	111
Exa 7.2	Rate of reaction	111
Exa 7.3	Rate of reaction	111
Exa 7.4	Order of reaction	112
Exa 7.5	Rate Expression	112
Exa 7.6	Volume function	113
Exa 7.7	Pressure time relation	113
Exa 7.8	Entropy changes	113
Exa 7.9	Hydrocarbon cracking	114
Exa 7.10	Equilibrium conversion	114
Exa 7.11	Equilibrium conversion	115
Exa 7.12	Concentration calculation	115
Exa 7.13	Equilibrium conversion	116
Exa 7.14	Equilibrium shifts	117
Exa 7.15	Rate equation	117
Exa 7.16	Rate of reaction	119
Exa 8.1	Specific gravity	120
Exa 8.2	Specific gravity	120
Exa 8.3	Specific gravity	121
Exa 8.4	Mixture density	121
Exa 8.5	Viscosity calculation	122
Exa 8.6	Solution viscosity	122
Exa 8.7	Flow rate calculation	123
Exa 8.8	Venturi meter	124
Exa 8.9	Venturi meter	125
Exa 8.10	Pitot tube	126
Exa 8.11	Rotameter capacity	127
Exa 8.12	Flow rate calculation	127
Exa 9.1	Coiled tube pressure drop	129
Exa 9.2	Heat exchanger pressure drop	130
Exa 9.3	Heat exchanger area	131
Exa 9.4	Batch distillation	132
Exa 9.5	Gas mixture exit temperature	133
Exa 9.6	Friction factor calculation	133

Chapter 1

Introduction

Scilab code Exa 1.1 Air composition

```
1 clc
2 clear
3 printf("example 1.1 page number 19\n\n")
4 //to find composition of air by weight
5 y_oxygen = 0.21 //mole fraction of oxygen
6 y_nitrogen = 0.79 //mole fraction of nitrogen
7 molar_mass_oxygen = 32
8 molar_mass_nitrogen = 28
9
10 molar_mass_air = y_oxygen*molar_mass_oxygen+
    y_nitrogen*molar_mass_nitrogen;
11 mass_fraction_oxygen =y_oxygen*molar_mass_oxygen/
    molar_mass_air;
12 mass_fraction_nitrogen = y_nitrogen*
    molar_mass_nitrogen/molar_mass_air;
13
14 printf("mass fraction of oxygen = %f \n\n",
    mass_fraction_oxygen)
15 printf("mass fraction of nitrogen = %f \n\n",
    mass_fraction_nitrogen)
16
```

```

17 V1 = 22.4           //in liters
18 P1 = 760           //in mm Hg
19 P2= 735.56        //in mm Hg
20 T1= 273           //in K
21 T2 = 298          //in K
22
23 V2= (P1*T2*V1)/(P2*T1);
24 density = molar_mass_air/V2;
25
26 printf("density = %f gm/l",density)

```

Scilab code Exa 1.2 Volume calculation

```

1  clc
2  clear
3  printf("example 1.2 page number 20\n\n")
4  //find the volume occupied by propane
5
6  mass_propane=14.2    //in kg
7  molar_mass=44       //in kg
8  moles=(mass_propane*1000)/molar_mass;
9  volume=22.4*moles;  //in liters
10
11 printf("volume = %d liters\n\n",volume)

```

Scilab code Exa 1.3 Gas Composition

```

1  clc
2  clear
3  printf("example 1.3 page number 20\n\n")
4  //to find the average weight, weight composition,
   gas volume in absence of SO2
5  y_CO2 = 0.25;

```

```

6 y_CO = 0.002;
7 y_SO2 = 0.012;
8 y_N2 = 0.680;
9 y_O2 = 0.056;
10
11 Mm = y_CO2*44+y_CO*28+y_SO2*64+y_N2*28+y_O2*32;
12 printf ("\n molar mass = %d \n",Mm)
13
14 printf("\n finding weight composition \n")
15 w_CO2 = y_CO2*44*100/Mm;
16 printf ("\n weight_CO2 = %f \n\n",w_CO2)
17 w_CO = y_CO*28*100/Mm;
18 printf ("weight_CO = %f \n\n",w_CO)
19 w_SO2 = y_SO2*64*100/Mm;
20 printf ("weight_SO2 = %f \n\n", w_SO2)
21 w_N2 = y_N2*28*100/Mm;
22 printf ("weight_N2 = %f \n\n", w_N2)
23 w_O2 = y_O2*32*100/Mm;
24 printf ("weight_O2 = %f \n\n", w_O2)
25
26 printf("if SO2 is removed \n\n")
27 v_CO2 = 25;
28 v_CO = 0.2;
29 v_N2 = 68.0;
30 v_O2 = 5.6;
31 v = v_CO2+v_CO+v_N2+v_O2;
32 v1_CO2 = (v_CO2*100/98.8);
33
34 printf ("volume_CO2 = %f \n\n", v1_CO2)
35 v1_CO = (v_CO*100/98.8);
36 printf ("volume_CO = %f \n\n",v1_CO)
37 v1_N2 = (v_N2*100/98.8);
38 printf ("volume_N2 = %f \n\n",v1_N2)
39 v1_O2 = (v_O2*100/98.8);
40 printf ("volume_O2 = %f \n\n",v1_O2 )

```

Scilab code Exa 1.4 Volume calculation

```
1 clc
2 clear
3 printf("example 1.4 page number 24\n\n")
4 //to find volume of NH3 dissolvable in water
5
6 p=1 //atm
7 H=2.7 //atm
8 x=p/H;
9
10 mole_ratio = (x)/(1-x);
11 moles_of_water=(100*1000)/18;
12 moles_of_NH3=mole_ratio*moles_of_water;
13
14 printf("moles of NH3 dissolved = %f\n\n",
        moles_of_NH3)
15
16 volume_NH3=(moles_of_NH3*22.4*293)/273;
17 printf("volume of NH3 dissolved = %f liters",
        volume_NH3)
```

Scilab code Exa 1.5 Amount of CO2 released

```
1 clc
2 clear
3
4 printf("example 1.5 page number 24\n\n")
5
6 //to calculate amount of CO2 released by water
7 p=746 //in mm Hg
8 H=1.08*10^6 //in mm Hg, Henry's constant
```

```

9
10 x= p/H;    //mole fraction of CO2
11 X=x*(44/18);    //mass ratio of CO2 in water
12
13 initial_CO2 = 0.005;    //kg CO2/kg H2O
14 G=1000*(initial_CO2-X);
15
16 printf("CO2 given up by 1 cubic meter of water = %f
    kg CO2/cubic meter H2O",G)

```

Scilab code Exa 1.6 Vapor pressure

```

1  clc
2  clear
3
4  printf('example 1.6 page number 27 \n\n')
5  //to find vapor pressre of ethyl alchohal
6
7  pa1 = 23.6;    //VP of ethyl alchohal at 10 degree
    C
8  pa3=760    //VP of ethyl alchohal at 78.3 degree C
    in mm Hg
9  pb1 = 9.2    //VP of ethyl water at 10 degree C in
    mm Hg
10 pb3=332    //VP of ethyl water at 78.3 degree C in
    mm Hg
11
12 C=(log10(pa1/pa3)/(log10(pb1/pb3)));
13
14 pb2=149    //VP of water at 60 degree C in mm Hg
15
16 pas=(pb3/pb2);
17 pa=C*log10(pas);
18 pa2=pa3/(10^pa);
19

```

```
20 printf("vapor pressure of ethyl alcohol at 60
    degree C = %f mm Hg",pa2)
```

Scilab code Exa 1.7 Duhring Plot calculations

```
1 clc
2 clear
3
4 printf('example 1.7 page number 28 \n\n')
5
6 //to find vapor pressure using duhring plot
7
8 t1 = 41 //in degree C
9 t2=59 //in degree C
10 theta_1 =83 //in degree C
11 theta_2=100 //in degree C
12
13 K = (t1-t2)/(theta_1-theta_2);
14 t=59+(K*(104.2-100));
15
16 printf ("boiling point of SCl2 at 880 Torr = %f
    degree celcius",t)
```

Scilab code Exa 1.8 Vapor Pressure of Mixture

```
1 clc
2 clear
3 printf('example 1.8 page number 29\n\n')
4 //to find the amount of steam released
5
6 vp_C6H6 = 520 //in torr
7 vp_H2O = 225 //in torr
8 mass_water=18
```

```

9 mass_benzene=78
10
11 amount_of_steam = (vp_H20/vp_C6H6)/(mass_benzene/
    mass_water);
12
13 printf("amount of steam = %f", amount_of_steam)

```

Scilab code Exa 1.9 Vapor pressure

```

1 clc
2 clear
3 printf('example 1.9 page number 30\n\n')
4
5 //to find equilibrium vapor liquid composition
6 p0b = 385 //vapor pressue of benzene at 60
    degree C in torr
7 p0t=140 //vapor pressue of toluene at 60 degree
    C in torr
8 xb=0.4;
9 xt=0.6;
10
11 pb=p0b*xb;
12 pt=p0t*xt;
13 P=pb+pt;
14
15 printf("total pressure = %f torr\n\n",P)
16
17 yb=pb/P;
18 yt=pt/P;
19 printf("vapor composition of benzene = %f \n vapor
    composition of toluene = %f\n\n",yb,yt)
20
21 //for liquid boiling at 90 degree C and 760 torr ,
    liquid phase composition
22 //x=(760-408)/(1013-408);

```



```

23 (1013*x)+(408*(1-x))==760;
24 printf("mole fraction of benzene in liquid mixture =
      %f \n mole fraction of toluene in liquid mixture
      = %f",x,1-x)

```

Scilab code Exa 1.10 Flow relation

```

1  clc
2  clear
3
4  printf('example 1.10 page number 33\n')
5
6  //to find relation between friction factor and
      reynold's number
7
8  //log f=y, log Re=x, log a=c
9  sigma_x=23.393;
10 sigma_y=-12.437;
11 sigma_x2=91.456
12 sigma_xy=-48.554;
13 m=((6*sigma_xy)-(sigma_x*sigma_y))/(6*sigma_x2-(
      sigma_x)^2);
14 printf("m = %f \n",m)
15
16 c=((sigma_x2*sigma_y)-(sigma_xy*sigma_x))/(6*
      sigma_x2-(sigma_x)^2);
17 printf("c = %f \n",c)
18
19 printf("f=0.084*Re^-0.256")

```

Scilab code Exa 1.11 Average Velocity

```

1  clc

```

```

2 clear
3 printf("example 1.11 page number 35\n\n")
4
5 //to find the average velocity
6
7 u = [2;1.92;1.68;1.28;0.72;0];
8 r = [0;1;2;3;4;5];
9
10 z = u.*r;
11 plot(r,z)
12 title("variation of ur with r")
13 xlabel("r")
14 ylabel("ur")
15
16 //by graphical integration , we get
17 u_avg = (2/25)*12.4
18 printf("average velocity = %f cm/s\n",u_avg)

```

Scilab code Exa 1.12 Velocity determination

```

1 clc
2 clear
3
4 printf('example 1.12 page number 37\n')
5
6 //to find the average velocity
7 printf('using trapezoid rule\n')
8
9 n = 6;
10 h = (3 - 0)/n;
11 I = (h/2)*(0+2*0.97+2*1.78+2*2.25+2*2.22+2*1.52+0);
12 u_avg = (2/3^2)*I;
13
14 printf("average velocity = %f cm/s\n",u_avg)
15

```

```

16 disp('Simpsons rule')
17
18 n = 6;
19 h = 3/n;
20 I = (h/3)*(0+4*(0.97+2.25+1.52)+2*(1.78+2.22)+0);
21 u_avg = (2/3^2)*I;
22
23 printf("average velocity = %f cm/s\n",u_avg)

```

Scilab code Exa 1.13 Velocity determination

```

1  clc
2  clear
3
4  printf('example 1.13 page number 38\n\n')
5
6  //to find the settling velocity as a function of
   time
7  z0 = 30.84;
8  z1 = 29.89;
9  z2 = 29.10;
10 h = 4;
11
12 u1_t0 = (-3*z0+4*z1-z2)/(2*h);
13 u1_t4 = (-z0+z2)/(2*h);
14 u1_t8 = (z0-4*z1+3*z2)/(2*h);
15
16 //considering data set for t = 4,8,12 min
17 z0 = 29.89;
18 z1 = 29.10;
19 z2 = 28.30;
20 u2_t4 = (-3*z0+4*z1-z2)/(2*h);
21 u2_t8 = (-z0+z2)/(2*h);
22 u2_t12 = (z0-4*z1+3*z2)/(2*h);
23

```

```

24 //considering data set for t = 8,12,16 min
25 z0 = 29.10;
26 z1 = 28.30;
27 z2 = 27.50;
28 u3_t8 = (-3*z0+4*z1-z2)/(2*h);
29 u3_t12 = (-z0+z2)/(2*h);
30 u3_t16 = (z0-4*z1+3*z2)/(2*h);
31
32 //taking average
33 u_t4 = (u1_t4+u2_t4)/2;
34 u_t8 = (u1_t8+u2_t8+u3_t8)/3;
35 u_t12 = (u2_t12+u3_t12)/2;
36
37 printf("u_t0 = %f cm/min\n u_t4 = %f cm/min\n u_t8 =
        %f cm/min \n u_t12 = %f/n cm/min\n u_t16 =%f/n
        cm/min ",u1_t0,u_t4,u_t8,u_t12,u3_t16)

```

Scilab code Exa 1.14 Dimensional analysis

```

1 printf('example 1.14 page number 45')
2 disp ("this is a theoretical question , book shall be
        referred for solution")

```

Scilab code Exa 1.15 Dimensional analysis

```

1 printf('example 1.15 page number 46')
2 disp ("this is a theoretical question , book shall be
        referred for solution")

```

Scilab code Exa 1.16 Dynamic similarity

```

1  clc
2  clear
3  printf('example 1.16 page number 49\n')
4
5  //to find the flow rate and pressure drop
6  density_water=988      //in kg/m3
7  viscosity_water=55*10^-5 //in Ns/m2
8  density_air=1.21      //in kg/m3
9  viscosity_air=1.83*10^-5 //in Ns/m2
10 L=1      //length in m
11
12 L1=10*L   //length in m
13 Q=0.0133;
14
15 Q1=((Q*density_water*viscosity_air*L)/(L1*
      viscosity_water*density_air))
16
17 printf("flow rate = %f cubic meter/s\n",Q1)
18
19 //equating euler number
20
21 p=9.8067*10^4;      //pressure in pascal
22 p1=(p*density_water*Q^2*L^4)/(density_air*Q1^2*L1^4)
      ;
23
24 printf("pressure drop corresponding to 1kp/square cm
      = %f kP/square cm",p1/p)

```

Scilab code Exa 1.17 Dynamic similarity

```

1  clc
2  clear
3  printf('example 1.17 page number 50\n')
4
5  //to find the specific gravity of plastic

```

```

6
7 L=1      //length of prototype in m
8 L1=10*L  //length of model in m
9 density_prototype=2.65 //gm/cc
10 density_water=1      //gm/cc
11
12 density_model=(L^3*(density_prototype-density_water)
    )/(L1^3)+1;
13
14 printf("specific gravity of plastic = %f",
    density_model)

```

Scilab code Exa 1.18 Nomographic chart

```

1 clc
2 clear
3 printf('example 1.18 page number 53\n\n')
4
5 //to find error in actual data and nomographic chat
  value
6
7 //for my
8 ly = 8 //in cm
9 my = ly/((1/0.25) - (1/0.5));
10 lz = 10.15 //in cm
11 mz = lz/((1/2.85) - (1/6.76));
12 mx = (my*mz)/(my+mz);
13 printf("mx = %f cm\n",mx)
14 err = ((1-0.9945)/0.9945)*100;
15 printf("\nerror = %f \n",err)
16
17 x = 2
18 y = 0.5:0.5:2.5;
19
20 plot(x,y)

```

```
21 title("nomograph")
22 xlabel("x")
23 ylabel("y")
24
25 x = 3
26 y = 0.4:0.2:2;
27 plot(x,y)
```

Scilab code Exa 1.19 Calculation using Nomograph

```
1 clc
2 clear
3 printf('example 1.19 page number 54\n')
4
5 //to find the economic pipe diameter from nomograph
6 //from the nomograph,we get the values of w and
   density
7
8 w=450 //in kg/hr
9 density=1000 //in kg/m3
10 d=16 //in mm
11
12 u=(w/density)/(3.14*d^2/4);
13 Re=u*density*d/0.001;
14
15 if Re>2100 then printf("flow is turbulent and d= %f
   mm",d)
16 else disp ("flow is laminar and this nomograph is
   not valid")
17 end
```

Chapter 2

Physico Chemical Calculations

Scilab code Exa 2.1 Ideal gas system

```
1 clc
2 clear
3 printf("example 2.1 page number 71\n\n")
4
5 //to find the volume of oxygen that can be obtained
6
7 p1=15      //in bar
8 p2=1.013   //in bar
9 t1=283     //in K
10 t2=273    //in K
11 v1=10     //in l
12
13 v2=p1*v1*t2/(t1*p2);
14
15 printf("volume of oxygen = %f liters",v2)
```

Scilab code Exa 2.2 Mixture properties


```

1  clc
2  clear
3  printf("example 2.2 page number 71\n\n")
4
5  //to find volumetric composition ,partial pressue of
   each gas and total pressure of mixture
6
7  nCO2=2/44;      //moles of CO2
8  nO2=4/32;      //moles of O2
9  nCH4=1.5/16;   //moles of CH4
10
11 total_moles=nCO2+nO2+nCH4;
12 yCO2=nCO2/total_moles;
13 yO2=nO2/total_moles;
14 yCH4=nCH4/total_moles;
15
16 printf (" Composition of mixture = \nCH4 = %f \nO2 =
   %f \n CO2 = %f \n\n",yCH4,yO2,yCO2)
17
18 pCO2=nCO2*8.314*273/(6*10^-3);
19 pO2=nO2*8.314*273/(6*10^-3);
20 pCH4=nCH4*8.314*273/(6*10^-3);
21
22 printf ("pressure of CH4 = %f kPa \npressure of O2 =
   %f kPa\n pressure of CO2 =%f kPa\n\n",pCH4
   *10^-3,pO2*10^-3,pCO2*10^-3)
23
24 total_pressure=pCO2+pCH4+pO2;
25 printf ("total pressure = %f Kpa",total_pressure
   *10^-3)

```

Scilab code Exa 2.3 Equivalent metal mass

```

1  clc
2  clear

```

```

3 printf("example 2.3 page number 72\n\n")
4
5 //to find equivalent mass of metal
6
7 P=104.3 //total pressure in KPa
8 pH2O=2.3 //in KPa
9 pH2=P-pH2O; //in KPa
10
11 VH2=209*pH2*273/(293*101.3)
12
13 printf("volume of hydrogen obtained = %f ml\n\n",VH2
    )
14
15 //calculating amount of metal having 11.2l of
    hydrogen
16
17 m=350/196.08*11.2 //mass of metal in grams
18 printf("mass of metal equivalent to 11.2 litre/mol
    of hydrogen = %f gm",m)

```

Scilab code Exa 2.4 Purity of Sodium Hydroxide

```

1 clc
2 clear
3 printf("example 2.4 page number 72\n\n")
4
5 //to find NaCl content in NaOH solution
6
7 w=2 //in gm
8 m=0.287 //in gm
9
10 //precipitate from 58.5gm of NaCl=143.4gm
11
12 mNaCl=58.5/143.4*m;
13

```

```
14 printf("mass of NaCl = %f gm\n",mNaCl )
15
16 percentage_NaCl=mNaCl/w*100;
17 printf("amount of NaCl = %f",percentage_NaCl)
```

Scilab code Exa 2.5 Carbon content formulation

```
1 clc
2 clear
3 printf("example 2.5 page number 72\n\n")
4
5 //to find the carbon content in sample
6
7 w=4.73 //in gm5
8 VC02=5.30 //in liters
9
10 weight_C02=44/22.4*VC02;
11 carbon_content=12/44*weight_C02;
12
13 percentage_content=(carbon_content/w)*100;
14
15 printf("percentage amount of carbon in sample = %f",
    percentage_content)
```

Scilab code Exa 2.6 Combustion of gas

```
1 clc
2 clear
3 printf("example 2.6 page number 73\n\n")
4 //to find the volume of air
5
6 volume_H2=0.5 //in m3
7 volume_CH4=0.35 //in m3
```

```

8 volume_CO=0.08 //in m3
9 volume_C2H4=0.02 //in m3
10 volume_oxygen=0.21 //in m3 in air
11
12 //required oxygen for various gases
13 H2=0.5*volume_H2;
14 CH4=2*volume_CH4;
15 CO=0.5*volume_CO;
16 C2H4=3*volume_C2H4;
17
18 total_O2=H2+CH4+CO+C2H4;
19 oxygen_required=total_O2/volume_oxygen;
20
21 printf("amount of oxygen required = %f cubic meter",
        oxygen_required)

```

Scilab code Exa 2.7 Sulphuric acid preparation

```

1 clc
2 clear
3 printf("example 2.7 page number 73\n\n")
4
5 //to find the volume of sulphuric acid and mass of
  water consumed
6
7 density_H2SO4 = 1.10 //in g/ml
8 mass_1 = 100*density_H2SO4; //mass of 100ml of 15%
  solution
9 mass_H2SO4 = 0.15*mass_1;
10 density_std = 1.84 //density of 96% sulphuric acid
11 mass_std = 0.96*density_std; //mass of H2SO4 in 1
  ml 96% H2SO4
12
13 volume_std = mass_H2SO4/mass_std; //volume of 96
  %H2SO4

```

```

14 mass_water = mass_1 - mass_H2SO4;
15
16 printf("volume of 0.96 H2SO4 required = %f ml",
        volume_std)
17 printf("\nmass of water required = %f g",mass_water)

```

Scilab code Exa 2.8 Molarity Molality Normality Calculation

```

1  clc
2  clear
3  printf("example 2.8 page number 73\n\n")
4
5  //to find molarity ,molality and normality
6
7  w_H2SO4=0.15    //in gm/lgm solution
8  density=1.10   //in gm/ml
9  m=density*1000; //mass per liter
10 weight=m*w_H2SO4; //H2SO4 per liter solution
11 molar_mass=98;
12
13 Molarity=weight/molar_mass;
14 printf("Molarity = %f mol/l\n\n",Molarity)
15
16 equivalent_mass=49;
17 normality=weight/equivalent_mass;
18 printf("Normality = %f N\n\n",normality)
19
20 molality=176.5/molar_mass;
21 printf("Molality = %f",molality)

```

Scilab code Exa 2.9 Normality calculation

```

1  clc

```

```

2 clear
3 printf("example 2.9 page number 74\n\n")
4
5 molar_mass_BaCl2=208.3;           //in gm
6 equivalent_H2SO4=0.144;
7 normality=equivalent_H2SO4*1000/28.8;
8
9 printf("Normality = %f N",normality)

```

Scilab code Exa 2.10 Precipitation of KClO₃

```

1 clc
2 clear
3 printf("example 2.10 page number 74\n\n")
4
5 //to find amount of KClO3 precipitated
6
7 solubility_70=30.2           //in gm/100gm
8 w_solute=solubility_70*350/130.2; //in gm
9
10 w_water=350-w_solute;
11 solubility_30=10.1         //in gm/100gm
12 precipitate=(solubility_70-solubility_30)*w_water
13                          /100
14 printf("amount precipitated = %f gm",precipitate)

```

Scilab code Exa 2.11 Solubility of CO₂

```

1 clc
2 clear
3 printf("example 2.11 page number 74\n\n")
4

```

```

5 //to find the pressure for solubility of CO2
6
7 absorbtion_coefficient=1.71 //in liters
8 molar_mass=44;
9 solubility=absorbtion_coefficient*molar_mass/22.4;
//in gm
10 pressure=8/solubility*101.3;
11
12 printf("pressure required = %f kPa",pressure)

```

Scilab code Exa 2.12 Vapor pressure calculation

```

1 clc
2 clear
3 printf("example 2.12 page number 74\n\n")
4
5 //to find the vapor pressure of water
6
7 w_water=540 //in gm
8 w_glucose=36 //in gm
9 m_water=18; //molar mass of water
10 m_glucose=180; //molar mass of glucose
11
12 x=(w_water/m_water)/(w_water/m_water+w_glucose/
m_glucose);
13 p=8.2*x;
14 depression=8.2-p;
15
16 printf("depression in vapor pressure = %f Pa",
depression*1000)

```

Scilab code Exa 2.13 Boiling point calculation

```

1  clc
2  clear
3  printf("example 2.13 page number 75\n\n")
4
5  //to find the boiling point of solution
6
7  w_glucose=9      //in gm
8  w_water=100     //in gm
9  E=0.52;
10 m=90/180;      //moles/1000gm water
11
12 delta_t=E*m;
13 boiling_point=100+delta_t;
14
15 printf("boiling_point of water = %f degreeC",
        boiling_point)

```

Scilab code Exa 2.14 Colligative properties

```

1  clc
2  clear
3  printf("example 2.14 page number 75\n\n")
4
5  //to find the molar mass and osmotic pressure
6
7  K=1.86;
8  c=15    //concentration of alcohol
9  delta_t=10.26;
10
11 m=delta_t/K;    //molality
12 M=c/(m*85);    //molar mass
13 printf("molar mass = %f gm\n\n",M*1000)
14
15 density=0.97    //g/ml
16 cm=c*density/(M*100);

```



```

17 printf("molar concentration of alcohol = %f moles/l\n\n",cm)
18
19 p=cm*8.314*293 //osmotic pressure
20 printf("osmotic pressure = %f Mpa\n\n",p/1000)

```

Scilab code Exa 2.15 Huggins Equation

```

1 clc
2 clear
3 printf("example 2.15 page number 75\n\n")
4
5 //to find u_in , M_v, k'
6
7 u_in = 0.575 //from the graph
8 u_s = 0.295 //in mPa-s
9
10 M_v = (u_in/(5.80*10^-5))^(1/0.72);
11 u_red = 0.628; //in dl/g
12
13 c = 0.40 //in g/dl
14 k = (u_red-u_in)/((u_in^2)*c);
15
16 printf("k = %f \nMv = %f\nu_in = %f dl/gm",k,M_v,
u_in)

```

Scilab code Exa 2.16 Molecular Formula

```

1 clc
2 clear
3 printf("example 2.16 page number 76\n\n")
4
5 //to find the molecular formula

```

```

6
7 C=54.5      //% of carbon
8 H2=9.1      //% of hydrogen
9 O2=36.4     //% of oxygen
10 x=C/12;    //number of carbon molecules
11 y=O2/16;   //number of oxygen molecules
12 z=H2/2     //number of hydrogen molecules
13 molar_mass=88;
14 density=44;
15
16 ratio=molar_mass/density;
17 x=ratio*2;
18 y=ratio*1;
19 z=ratio*4;
20
21 printf("x = %f, y = %f, z = %f",x,y,z)
22 printf("\n\nformula of butyric acid is = C4H8O2")

```

Scilab code Exa 2.17 Molecular Formula

```

1 clc
2 clear
3 printf("example 2.17 page number 77\n\n")
4
5 //to find molecular foemula
6 C=93.75     //% of carbon
7 H2=6.25     //% of hydrogen
8 x=C/12      //number of carbon atoms
9 y=H2/2      //number of hydrogen atoms
10 molar_mass=64
11 density=4.41*29;
12
13 ratio=density/molar_mass;
14
15 x=ratio*5;

```

```

16 y=ratio*4;
17
18
19 printf("x = %f, y = %f",x,y)
20 printf("\n\nformula of butyric acid is = C10H8")

```

Scilab code Exa 2.18 Molecular Formula

```

1  clc
2  clear
3  printf("example 2.18 page number 77\n\n")
4
5  //to find molecular formula
6  C=50.69      //% of carbon
7  H2=4.23     //% of hydrogen
8  O2=45.08    //% of oxygen
9  a=C/12;     //number of carbon molecules
10 c=O2/16;    //number of oxygen molecules
11 b=H2/2;     //number of hydrogen molecules
12 molar_mass=71;
13
14 function M=f(m)
15     M=(2.09*1000)/(60*m);
16
17 endfunction
18
19 M=f((1.25/5.1));
20
21 printf("actual molecular mass = %f\n\n",M)
22
23 ratio=M/molar_mass;
24 a=ratio*3;
25 b=ratio*3;
26 c=ratio*2;
27

```

```
28
29 printf("a = %f, b = %f, c = %f",a,b,c)
30 printf("\n\nformula of butyric acid is = C6H6O4")
```

Scilab code Exa 2.19 Molecular Formula

```
1  clc
2  clear
3  printf("example 2.19 page number 78\n\n")
4
5  //to find the molecular formula
6  C=64.6      //% of carbon
7  H2=5.2      //% of hydrogen
8  O2=12.6     //% of oxygen
9  N2=8.8      //% of nitrogen
10 Fe=8.8      //% of iron
11
12 a=C/12;     //number of carbon molecules
13 c=8.8/14;   //number of nitrogen molecules
14 b=H2/2;     //number of hydrogen molecules
15 d=O2/16;    //number of oxygen molecules
16 e=Fe/56     //number of iron atoms
17
18 cm=243.4/(8.31*293) //concentration
19
20 molar_mass=63.3/cm;
21
22 printf("a = %f, b = %f, c = %f, d = %f, e = %f",a
23        *6.5,b*6.5,c*6.5,d*6.5,e*6.5)
24 printf("\n\nformula of butyric acid is =
25        C34H33N4O5Fe")
```

Scilab code Exa 2.20 Metal deposition

```

1  clc
2  clear
3  printf("example 2.20 page number 78\n\n")
4
5  //to find sequence of deposition
6  E1=-0.25;
7  E2=0.80;
8  E3=0.34;
9
10 a=[E1;E2;E3];
11 b=gsort(a);
12
13 printf("sorted potential in volts =")
14 disp (b)
15 disp ("E2>E3>E1")
16 disp ("silver >copper >nickel")

```

Scilab code Exa 2.21 EMF of cell

```

1  clc
2  clear
3  printf("example 2.21 page number 79\n\n")
4
5  //to find the emf of cell
6
7  E0_Zn=-0.76;
8  E0_Pb=-0.13;
9  c_Zn=0.1;
10 c_Pb=0.02;
11
12 E_Zn=E0_Zn+(0.059/2)*log10(c_Zn);
13 E_Pb=E0_Pb+(0.059/2)*log10(c_Pb);
14 E=E_Pb-E_Zn;
15
16 printf("emf of cell = %f V",E)

```

```
17 printf("\n\nSince potential of lead is greater than
    that of zinc thus reduction will occur at lead
    electrode and oxidation will occur at zinc
    electrode")
```

Scilab code Exa 2.22 EMF of cell

```
1 clc
2 clear
3 printf("example 2.22 page number 79\n\n")
4
5 //to find the emf of cell
6 E0_Ag=0.80;
7 E0_AgNO3=0.80;
8 c_Ag=0.001;
9 c_AgNO3=0.1;
10
11 E_Ag=E0_Ag+(0.059)*log10(c_Ag);
12 E_AgNO3=E0_AgNO3+(0.059)*log10(c_AgNO3);
13 E=E_AgNO3-E_Ag;
14
15 printf("emf of cell = %f V" ,E)
16 printf("\n\nsince E is positive , the left hand
    electrode will be anode and the electron will
    travel in the external circuit from the left hand
    to the right hand electrode")
```

Scilab code Exa 2.23 EMF of cell

```
1 clc
2 clear
3 printf("example 2.23 page number 79\n\n")
4
```

```

5 //to find emf of cell
6 pH=12; //pH of solution
7 E_H2=0;
8 E2=-0.059*pH;
9 E=E_H2-E2;
10 printf("EMF of cell = %f V",E)

```

Scilab code Exa 2.24 Silver deposition

```

1 clc
2 clear
3 printf("example 2.24 page number 80\n\n")
4
5 //to find amount of silver deposited
6 I=3 //in Ampere
7 t=900 //in s
8 m_eq=107.9 //in gm/mol
9 F=96500;
10
11 m=(I*t*m_eq)/F;
12 printf("mass = %f gm",m)

```

Scilab code Exa 2.25 Electroplating time

```

1 clc
2 clear
3 printf("example 2.25 page number 80\n\n")
4
5 //to find the time for electroplating
6 volume=10*10*0.005; //in cm3
7 mass=volume*8.9;
8 F=96500;
9 atomic_mass=58.7 //in amu

```

```

10 current=2.5      //in Ampere
11
12 charge=(8.9*F*2)/atomic_mass;
13 yield=0.95;
14 actual_charge=charge/(yield*3600);
15 t=actual_charge/current;
16
17 printf("time required = %f hours",t)

```

Scilab code Exa 2.26 Water hardness

```

1  clc
2  clear
3  printf("example 2.26 page number 80\n\n")
4
5  //to find hardness of water
6  m_MgSO4=90    //in ppm
7  MgSO4_parts=120;
8  CaCO3_parts=100;
9
10 hardness=(CaCO3_parts/MgSO4_parts)*m_MgSO4;
11
12 printf("hardness of water = %f mg/l",hardness)

```

Scilab code Exa 2.27 Water hardness

```

1  clc
2  clear
3  printf("example 2.26 page number 80\n\n")
4
5  m1 = 162    //mass of calcium bi carbonate in mg
6  m2 = 73    //mass of magnesium bi carbonate in mg
7  m3 = 136   // mass of calsium sulfate in mg

```



```

8 m4 = 95 // mass of magnesium chloride
9 m5 = 500 //mass of sodium chloride in mg
10 m6 = 50 // mass of potassium chloride in mg
11
12 content_1 = m1*100/m1; //content of calcium bi
    carbonate in mg
13 content_2 = m2*100/(2*m2); //content of magnesium
    bi carbonate in mg
14 content_3 = m3*100/m3; // content of calsium sufate
    in mg
15 content_4 = m4*100/m4; // content of magnesium
    chloride
16
17 //part_1
18
19 temp_hardness = content_1 + content_2; //depends
    on bicarbonate only
20 total_hardness = content_1+content_2+content_3+
    content_4;
21 printf("total hardness = %f\n temporary hardness =
    %f \n",temp_hardness,total_hardness)
22
23 //part 2
24 wt_lime = (74/100)*(content_1+2*content_2+content_4)
    ;
25 actual_lime = wt_lime/0.85;
26 printf("amount of lime required = %f \n",actual_lime
    )
27
28 soda_required = (106/100)*(content_1+content_4);
29 actual_soda = soda_required/0.98;
30 printf("amount of soda required = %f \n",actual_soda
    )

```

Scilab code Exa 2.28 Water hardness

```

1  clc
2  clear
3  printf("example 2.28 page number 82\n\n")
4
5  //to find hardness of water
6
7  volume_NaCl=50      //in l
8  c_NaCl=5000        //in mg/l
9
10 m=volume_NaCl*c_NaCl;
11 equivalent_NaCl=50/58.5;
12
13 hardness=equivalent_NaCl*m;
14
15 printf("hardness of water = %f mg/l",hardness/1000)

```

Scilab code Exa 2.29 Mixture composition

```

1  clc
2  clear
3  printf("example 2.29 page number 82\n\n")
4
5  //to find the total vapor pressure and molar
   compositions
6
7  m_benzene = 55      //in kg
8  m_toluene = 28      //in kg
9  m_xylene = 17       // in kg
10
11 mole_benzene = m_benzene/78;
12 mole_toluene = m_toluene/92;
13 mole_xylene = m_xylene/106;
14
15 mole_total = mole_benzene+mole_toluene+mole_xylene;
16 x_benzene = mole_benzene/mole_total;

```

```

17 x_toluene = mole_toluene/mole_total;
18 x_xylene = mole_xylene/mole_total;
19
20 P = x_benzene*178.6+x_toluene*74.6+x_xylene*28;
21 printf("total pressure = %f kPa\n",P)
22
23 benzene = (x_benzene*178.6*100)/P;
24 toluene = (x_toluene*74.6*100)/P;
25 xylene = (x_xylene*28*100)/P;
26
27 printf("xylene = %f \n toluene = %f \n benzene = %f"
        ,xylene ,toluene ,benzene)

```

Scilab code Exa 2.30 Mixture composition

```

1  clc
2  clear
3  printf("example 2.30 page number 83\n\n")
4
5  //to find the mixture composition
6
7  vapor_pressure=8      //in kPa
8  pressure=100        //in kPa
9
10 //part 1
11 volume=1            //in m3
12 volume_ethanol=volume*(vapor_pressure/pressure);
13 volume_air=1-volume_ethanol;
14 printf("volumetric composition:- \nair composition =
        %f\n ethanol composition = %f",volume_air*100,
        volume_ethanol*100)
15
16 //part 2
17 molar_mass_ethanol=46;
18 molar_mass_air=28.9;

```

```

19 mass_ethanol=0.08*molar_mass_ethanol;    //in kg
20 mass_air=0.92*molar_mass_air;          //in kg
21 fraction_ethanol=(mass_ethanol*100)/(mass_air+
    mass_ethanol);
22 fraction_air=(mass_air*100)/(mass_air+mass_ethanol);
23 printf("\n\ncomposition by weight:-\nAir = %f
    Ethanol vapor = %f",fraction_air,fraction_ethanol
    )
24
25 //part 3
26 mixture_volume=22.3*(101.3/100)*(299/273);    //in m3
27 weight_ethanol=mass_ethanol/mixture_volume;
28 printf("\n\nweight of ethanol/cubic meter = %f Kg",
    weight_ethanol)
29
30 //part 4
31 w_ethanol=mass_ethanol/mass_air;
32 printf("\n\nweight of ethanol/kg vapor free air = %f
    Kg",w_ethanol)
33
34 //part 5
35 moles_ethanol=0.08/0.92;
36 printf("\n\nkmol of ethanol per kmol of vapor free
    air = %f",moles_ethanol)

```

Scilab code Exa 2.31 Mixture properties

```

1  clc
2  clear
3  printf("example 2.31 page number 84\n\n")
4
5  //to find relative saturation and dew point
6
7  vapor_pressure=8    //in kPa
8  volume_ethanol=0.05;

```

```

9
10 //basis 1kmol of mixture
11
12 partial_pressure=volume_ethanol*100;
13 relative_saturation=partial_pressure/vapor_pressure;
14 mole_ratio=volume_ethanol/(1-volume_ethanol);
15 printf("mole ratio = %f \nrelative saturation = %f",
        mole_ratio,relative_saturation*100)
16
17 //basis 1kmol saturated gas mixture at 100kPa
18 volume_vapor=(8/100)*100;
19 ethanol_vapor=volume_vapor/100;
20 air_vapor=1-ethanol_vapor;
21 saturation_ratio=ethanol_vapor/air_vapor;
22 percentage_saturation=mole_ratio/saturation_ratio;
23
24 printf("\n\npercentage saturation = %f",
        percentage_saturation)
25
26 //dew point
27 printf("\n\ncorresponding to partial pressure of 5
        kPa we get a dew point of 17.3 degree celcius")

```

Scilab code Exa 2.32 Humidity

```

1 clc
2 clear
3 printf("example 2.32 page number 84\n\n")
4
5 //to find the properties of humid air
6
7 p = 4.24 //in kPa
8 H_rel = 0.8;
9 p_partial = p*H_rel;
10 molal_H = p_partial/(100-p_partial);

```

```

11 printf("initial molal humidity = %f\n\n",molal_H)
12
13 //part 2
14 P = 200 //in kPa
15 p_partial = 1.70 //in kPa
16 final_H = p_partial/(P-p_partial);
17 printf("final molal humidity = %f\n\n",final_H)
18
19 //part 3
20 p_dryair = 100 - 3.39;
21 v = 100*(p_dryair/101.3)*(273/303);
22 moles_dryair = v/22.4;
23 vapor_initial = molal_H*moles_dryair;
24 vapor_final = final_H*moles_dryair;
25 water_condensed = (vapor_initial-vapor_final)*18;
26 printf("amount of water condensed = %f \n\n",
        water_condensed)
27
28 //part 4
29 total_air = moles_dryair+vapor_final;
30 final_v = 22.4*(101.3/200)*(288/273)*total_air;
31 printf("final volume of wety air = %f \n\n",final_v)

```

Chapter 3

Material and Energy Balances

Scilab code Exa 3.1 Coal consumption

```
1  clc
2  clear
3  printf("example 3.1 page number 90\n\n")
4
5  //to find the coal consumption
6  w_C = 0.6;    //amount of carbon in coal
7  N2_content = 40    //in m3 per 100m3 air
8
9  air_consumed = N2_content/0.79;
10 weight_air = air_consumed*(28.8/22.4);
11 O2_content = air_consumed*32*(0.21/22.4);    //in kg
12
13 H2_content = 20    //in m3
14
15 steam_consumed = H2_content*(18/22.4);
16
17 C_consumption1 = (12/18)*steam_consumed;    //in
    reaction 1
18 C_consumption2 = (24/32)*O2_content;    //in
    reaction 2
19
```

```

20 total_consumption = C_consumption1+C_consumption2;
21 coal_consumption = total_consumption/w_C;
22
23 printf("coal consumption = %f kg",coal_consumption)

```

Scilab code Exa 3.2 Nitric acid preparation

```

1  clc
2  clear
3  printf("example 3.2 page number 91\n\n")
4
5  //to find amount of ammonia and air consumed
6
7  NH3_required = (17/63)*1000;    //NH3 required for 1
    ton of nitric acid
8  NO_consumption = 0.96;
9  HNO3_consumption = 0.92;
10 NH3_consumed = NH3_required/(NO_consumption*
    HNO3_consumption);
11 volume_NH3 = NH3_consumed*(22.4/17);
12 printf("volume of ammonia consumed= %f cubic metre/h
    ",volume_NH3)
13
14 NH3_content = 11    //% by volume
15 air_consumption = volume_NH3*((100-11)/11);
16 printf("\n\nvolume of air consumed = %f cubic metre/
    h",air_consumption)

```

Scilab code Exa 3.3 HCl production

```

1  clc
2  clear
3  printf("example 3.3 page number 91\n\n")

```



```

4
5 //to find the consumption of NaCl and H2SO4 in HCl
  consumption
6
7 HCl_production = 500 //required to be produced in
  kg
8 NaCl_required = (117/73)*HCl_production;
9 yield = 0.92;
10 purity_NaCl= 0.96;
11
12 actual_NaCl = NaCl_required/(purity_NaCl*yield);
13 printf("amount of NaCl required = %f kg",actual_NaCl
  )
14
15 purity_H2SO4 = 0.93;
16 H2SO4_consumption = (98/73)*(HCl_production/(yield*
  purity_H2SO4));
17 printf("\n\namount of H2SO4 consumed = %f kg",
  H2SO4_consumption)
18
19 Na2SO4_produced = (142/73)*HCl_production;
20 printf("\n\namount of Na2SO4 produced = %f kg",
  Na2SO4_produced)

```

Scilab code Exa 3.4 Acetylene consumption

```

1 clc
2 clear
3 printf("example 3.4 page number 92\n\n")
4
5 //to find the period of service
6
7 C2H2_produced = (1/64)*0.86; //in kmol
8 volume_C2H2 = C2H2_produced*22.4*1000; //in l
9

```

```

10 //assuming ideal behaviour ,
11 volume = (100/101.3)*(273/(273+30));
12 time = (volume_C2H2/volume)*(1/60);
13 printf("time of service = %f hr",time)

```

Scilab code Exa 3.5 Screen effectiveness

```

1  clc
2  clear
3  printf("example 3.5 page number 92\n\n")
4
5  //to find the screen effectiveness
6
7  xv = 0.88;
8  xf = 0.46;
9  x1 = 0.32;
10 F= 100 //in kg
11
12 L = (F*(xf-xv))/(x1-xv);
13 V = F-L;
14 printf("L = %f Kg \nV = %f Kg",L,V)
15 Eo = (V*xv)/(F*xf);
16
17 printf(" \n\neffectiveness based on oversized
    partices = %f \n\n",Eo)
18 Eu = (L*(1-x1))/(F*(1-xf));
19
20 printf("effectiveness based on undersized partices =
    %f",Eu)
21 E=Eu*Eo;
22
23 printf("\n\overall effectiveness = %f",E)

```

Scilab code Exa 3.6 Absorption

```
1  clc
2  clear
3  printf("example 3.6 page number 94\n\n")
4
5  //to find the flow rate and concentration
6
7  G1 = 3600    //in m3/h
8  P = 106.6    //in kPa
9  T = 40    //in degree C
10 q = G1*(P/101.3)*(273/((273+T)));    //in m3/s
11 m = q/22.4;    //in kmol/h
12 y1 = 0.02;
13 Y1 = y1/(1-y1);
14
15 printf("mole ratio of benzene = %f kmol benzene/kmol
        dry gas",Y1)
16
17 Gs = m*(1-y1);
18 printf("\n\nmoles of benzene free gas = %f kmol
        drygas/h",Gs)
19
20 //for 95% removal
21 Y2 = Y1*(1-0.95);
22 printf("\n\nfinal mole ratio of benzene = %f kmol
        benzene/kmol dry gas",Y2)
23
24 x2 = 0.002
25 X2 = 0.002/(1-0.002);
26
27 //at equilibrium  $y^* = 0.2406X$ 
28 //part 1
29 //for oil rate to be minimum the wash oil leaving
        the absorber must be in equilibrium with the
        entering gas
30
31 y1 = 0.02;
```

```

32 x1 = y1/(0.2406);
33 X1 = x1/(1-x1);
34 min_Ls = Gs*((Y1-Y2)/(X1-X2));
35 printf("\n\nminimum Ls required = %f kg/h",min_Ls
        *260)
36
37 //for 1.5 times of the minimum
38 Ls = 1.5*min_Ls;
39 printf("\n\nflow rate of wash oil = %f kg/h",Ls*260)
40 X1 = X2 + (Gs*((Y1-Y2)/Ls));
41 printf("\n\nconcentration of benzene in wash oil =
        %f kmol benzene/kmol wash oil",X1)

```

Scilab code Exa 3.7 Extraction

```

1  clc
2  clear
3  printf("example 3.7 page number 95\n\n")
4
5  //to find the extraction of nicotine
6  xf = 0.01
7  Xf = xf/(1-xf);
8  Feed = 100 //feed in kg
9  c_nicotine = Feed*Xf; //nicotine conc in feed
10 c_water = Feed*(1-Xf) //water conc in feed
11
12 //part 1
13 function[f] = F1(x)
14     funcprot(0)
15     f = (x/150) - 0.9*((1-x)/99);
16 endfunction
17
18 //initial guess
19 x = 10;
20 y = fsolve(x,F1);

```

```

21 printf("amount of nicotine removed N = %f kg",y)
22 //part 2
23 function[f] = F1(x)
24     f = (x/50)-0.9*((1-x)/99);
25 endfunction
26
27 //initial guess
28 x = 10;
29 N1 = fsolve(x,F1);
30 printf("\n\namount of nicotine removed in stage 1,
        N1 = %f kg",N1)
31 function[f] = F1(x,N1)
32     f = (x/50)-0.9*((1-x-N1)/99);
33 endfunction
34
35 //initial guess
36 x = 10;
37 N2 = fsolve(x,F1);
38 printf("\n\namount of nicotine removed in stage 2,
        N2 = %f kg",N2)
39 function[f] = F1(x,N1,N2)
40     f = (x/50)-0.9*((1-x-N2-N1)/99);
41 endfunction
42
43 //initial guess
44 x = 10;
45 N3 = fsolve(x,F1);
46
47 printf("\n\namount of nicotine removed in stage 3,
        N3 = %f kg",N3)
48 N = N1+N2+N3;
49 printf("\n\nntotal amount of nicotine removed = %f kg
        ",N)

```

Scilab code Exa 3.8 Distillation

```

1  clc
2  clear
3  printf("example 3.8 page number 96\n\n")
4
5  //to find the amount of water in residue
6
7  vp_water = 31.06    //in kPa
8  vp_benzene = 72.92    //in kPa
9
10 P = vp_water +vp_benzene;
11 x_benzene = vp_benzene/P;
12 x_water = vp_water/P;
13
14 initial_water = 50/18;    //in kmol of water
15 initial_benzene = 50/78    //in kmol of benzene
16 water_evaporated = initial_benzene*(x_water/
    x_benzene);
17 water_left = (initial_water - water_evaporated);
18
19 printf("amount of water left in residue = %f kg",
    water_left*18)

```

Scilab code Exa 3.9 Distillation

```

1  clc
2  clear
3  printf("example 3.9 page number 97\n\n")
4
5  //to find the vapor content of dimethylaniline
6  po_D = 4.93    //in kPa
7  po_W = 96.3    //in kPa
8  n = 0.75    //vaporization efficiency
9
10 P = n*po_D+po_W;
11 printf("P = %f kPa",P)

```

```

12
13 x_water = 96.3/100;
14 x_dimethylanaline = 1-x_water;
15 wt_dimethylanaline = (x_dimethylanaline*121)/(
    x_dimethylanaline*121+x_water*18);
16 printf("\n\nweight of dimethylanaline in water = %f"
    ,wt_dimethylanaline*100)
17
18 //part 1
19 n = 0.8;
20 po_D = 32 //in kPa
21 actual_vp = n*po_D;
22 p_water = 100 - actual_vp;
23 steam_required = (p_water*18)/(actual_vp*121);
24 printf("\n\namount of steam required = %f kg steam/
    kg dimethylanaline",steam_required)
25
26 //part 2
27 x_water = p_water/100;
28 wt_water = x_water*18/(x_water*18+(1-x_water)*121);
29 printf("\n\nweight of water vapor = %f \nweight of
    dimethylanaline =%f",wt_water*100,100*(1-wt_water
    ))

```

Scilab code Exa 3.10 Crystallization

```

1 clc
2 clear
3 printf("example 3.10 page number 98\n\n")
4
5 //to find the amount of water evaporated
6 xf = 0.15;
7 xl = (114.7)/(114.7+1000);
8 xc = 1;
9

```

```

10 K2Cr207_feed = 1000*0.15;    //in kg
11
12 n = 0.8;
13 C = n*K2Cr207_feed;
14 V = (K2Cr207_feed-120 - 880*0.103)/(-0.103);
15
16 printf("amount of water evaporated = %f kg",V)

```

Scilab code Exa 3.11 crystallization

```

1 clc
2 clear
3 printf("example 3.10 page number 98\n\n")
4
5 //to find the yield of crystals
6
7 xc = 106/286;
8 xf = 0.25;
9 x1 = 27.5/127.5;
10
11 water_present = 100*(1-xf);    //in kg
12 V = 0.15*75;    //in kg
13 C = (100*xf - 88.7*x1)/(xc-x1);
14 Na2CO3_feed = 25/xc;
15
16 yield = (C/Na2CO3_feed)*100;
17
18 printf("yield = %f ",yield)

```

Scilab code Exa 3.12 Drying

```

1 clc
2 clear

```



```

3 printf("example 3.12 page number 99\n\n")
4
5 //to find the fraction of air recirculated
6
7 r = 50 //weight of dry air passing through drier
8 w1 = 1.60 //in kg per kg dry solid
9 w2 = 0.1 //in kg/kg dry solid
10 H0 = 0.016 //in kg water vapor/kg dry air
11 H2 = 0.055 //in kg water vapor/kg dry air
12
13 y = 1 - (w1-w2)/(r*(H2-H0));
14 printf("fraction of air recirculated = %f",y)
15
16 H1 = H2 - (w1-w2)/r;
17 printf("\n\nhumidity of air entering the drier = %f
      kg water vapor/kg kg dry air",H1)
18
19 //check
20 H11 = H2*y+H0*(1-y);
21 if H1 == H11 then printf("\n\nfraction of air
      recirculated = %f \n verified",y)
22 end

```

Scilab code Exa 3.13 Conditioning of air

```

1 clc
2 clear
3 printf("example 3.13 page number 100\n\n")
4
5 //to find the volumetric flow rate and fraction of
      air passing through the cooler
6
7 //basis 60m3/h of conditioned air at 25 degree C and
      60% RH
8

```

```

9 Hf = 0.012;
10 Hi = 0.033;
11 H1 = 0.0075;
12
13 water_vapor = Hf/18;    //in kmol of water vapor
14 dry_air = 1/28.9;    //in kmol
15 total_mass = water_vapor+dry_air;
16
17 volume = 22.4*(298/273)*total_mass;
18 weight = 60/volume;
19 printf("weight of dry air handled per hr = %f kg",
        weight)
20
21 //part 1
22 inlet_watervapor = 0.033/18;    //in kmol of water
        vapor
23 volume_inlet = 22.4*(308/273)*(inlet_watervapor+
        dry_air);
24 printf("\n\nvolumetric flow rate of inlet air = %f
        cubic meter", volume_inlet*weight)
25
26 //part 2
27 y = (Hf - Hi)/(H1 - Hi);
28 printf("\n\nfraction of inlet air passing through
        cooler = %f", y)

```

Scilab code Exa 3.14 Ammonia Synthesis

```

1 clc
2 clear
3 printf("example 3.14 page number 102\n\n")
4
5 //to find the fraction of purged recycle and total
        yield
6

```

```

7 //x- moles of N2 and H2 recycled; y - moles of N2 H2
  purged
8
9 Ar_freshfeed = 0.2;
10 //argon in fresh feed is equal to argon in purge
11
12 y = 0.2/0.0633; //argon in purge = 0.0633y
13 x = (0.79*100 - y)/(1-0.79);
14 printf("y = %f kmol\nx = %f kmol",y,x)
15
16 //part 1
17 fraction = y/x;
18 printf("\n\nfraction of recycle that is purged = %f",
  fraction)
19
20 //part 2
21 yield = 0.105*(100+x);
22 printf("\n\noverall yield of ammonia = %f kmol",
  yield)

```

Scilab code Exa 3.15 Enthalpy calculation

```

1 clc
2 clear
3 printf("example 3.15 page number 107\n\n")
4
5 //to find change in enthalpy
6 H0_CH4 = -74.9 //in kJ
7 H0_CO2 = -393.5 //in kJ
8 H0_H2O = -241.8 //in kJ
9
10 delta_H0 = H0_CO2+2*H0_H2O-H0_CH4;
11 printf("change in enthalpy = %f kJ",delta_H0)

```

Scilab code Exa 3.16 Enthalpy calculation

```
1 clc
2 clear
3 printf("example 3.16 page number 107\n\n")
4
5 //to compare the enthalpy change in two reactions
6
7 H0_glucose = -1273 //in kJ
8 H0_ethanol = -277.6 //in kJ
9 H0_CO2 = -393.5 //in kJ
10 H0_H2O = -285.8 //in kJ
11
12 //for reaction 1
13 delta_H1 = 2*H0_ethanol+2*H0_CO2-H0_glucose;
14 printf("enthalpy change in reaction 1 = %f KJ",
        delta_H1)
15
16 //for reaction 2
17 delta_H2 = 6*H0_H2O+6*H0_CO2-H0_glucose;
18 printf("\n\nenthalpy change in reaction 2 = %f kJ",
        delta_H2)
19
20 if delta_H1>delta_H2 then disp ("reaction 2 supplies
        more energy")
21     else disp ("reaction 1 supplies more energy")
22 end
```

Scilab code Exa 3.17 Enthalpy of formation

```
1 clc
2 clear
```

```

3 printf("example 3.17 page number 108\n\n")
4
5 //to find enthalpy of formation of CuSO4.5H2O
6
7 delta_H2 = 11.7 //in kJ/mol
8 m_CuSO4 = 16 //in gm
9 m_H2O = 384 //in gm
10
11 delta_H3 = -((m_CuSO4+m_H2O)*4.18*3.95*159.6)
12 //((16*10^3)
13
14 delta_H1 = delta_H3 - delta_H2;
15
16 printf("enthalpy of formation = %f kJ/mol",delta_H1)

```

Scilab code Exa 3.18 Combustion

```

1 clc
2 clear
3 printf("example 3.18 page number 108\n\n")
4
5 //to find the temperature of combustion
6
7 H_combustion = 1560000 //in kJ/kmol
8 H0_CO2 = 54.56 //in kJ/kmol
9 H0_O2 = 35.2 //in kJ/kmol
10 H0_steam = 43.38 //in kJ/kmol
11 H0_N2 = 33.32 //in kJ/kmol
12
13 t = H_combustion/(2*H0_CO2+3*H0_steam+0.875*H0_O2
14 //+16.46*H0_N2);
15
16 printf("theoretical temperature of combustion = %f
17 //degree C",t)

```

Scilab code Exa 3.19 Heat of reaction

```
1 clc
2 clear
3 printf("example 3.19 page number 109\n\n")
4
5 //to find the heat of reaction and consumption of
  coke
6
7 H_NaCl = 410.9 //in MJ/kmol
8 H_H2SO4 = 811.3 //in MJ/kmol
9 H_Na2SO4 = 1384 //in MJ/kmol
10 H_HCl = 92.3 //in MJ/kmol
11
12 Q = H_Na2SO4 + 2*H_HCl -2*H_NaCl -H_H2SO4;
13 printf("heat of reaction = %f MJ\n\n",Q)
14
15 heat_required = 64.5*(500/73);
16 coke_consumption = heat_required/19
17 printf("amount of coke oven gas consumed = %f cubic
  meter",coke_consumption)
```

Scilab code Exa 3.20 Heat transfer

```
1 clc
2 clear
3 printf("example 3.20 page number 109\n\n")
4
5 //to find the rate of heat flow
6
7 cp_water = 146.5 //in kj/kg
8 cp_steam = 3040 //in kJ/kg
```

```

9 d = 0.102 //in m
10 u = 1.5 //in m/s
11 density = 1000 //in kg/m3
12
13 m = (3.14/4)*d^2*u*density;
14 Q = m*(cp_steam-cp_water);
15
16 printf("rate of heat flow = %f kW",Q)

```

Scilab code Exa 3.21 Calorific value

```

1 clc
2 //EXAMPLE 3.21
3 //To find the calorific value of coal
4 disp('this is a theoretical problem.Refer the book
      for solution')

```

Scilab code Exa 3.22 Coal combustion

```

1 clc
2 clear
3 printf("example 3.22 page number 110\n\n")
4
5 //to find the amount of air required for combustion
  and composition of flue gas
6 wt_C = 0.75 //in kg
7 wt_H2 = 0.05 //in kg
8 wt_O2 = 0.12 //in kg
9 wt_N2 = 0.03 //in kg
10 wt_S = 0.01 //in kg
11 wt_ash = 0.04 //in kg
12
13 O2_C = wt_C*(32/12); //in kg

```

```

14 O2_H2 = wt_H2*(16/2); //in kg
15 O2_S = wt_S*(32/32); //in kg
16 O2_required = O2_C+O2_H2+O2_S;
17
18 oxygen_supplied = O2_required - wt_O2;
19 air_needed = oxygen_supplied/0.23;
20 printf("amount of air required = %f kg",air_needed)
21
22 volume = (22.4/28.8)*air_needed;
23 printf("\n\nvolume of air needed = %f cubic meter",
    volume)
24
25 air_supplied = 1.20*air_needed;
26 N2_supplied = air_supplied*0.77;
27 total_N2 = N2_supplied+wt_N2;
28
29 O2_fluegas = air_supplied*0.23 - oxygen_supplied;
30
31 wt_CO2 = wt_C+O2_C;
32 wt_SO2 = wt_S+O2_S;
33
34 moles_CO2 = wt_CO2/44;
35 moles_SO2 = wt_SO2/64;
36 moles_N2 = total_N2/28;
37 moles_O2 = O2_fluegas/32;
38 total_moles = moles_CO2+moles_SO2+moles_N2+moles_O2;
39
40 x_CO2 = moles_CO2/total_moles;
41 x_SO2 = moles_SO2/total_moles;
42 x_N2 = moles_N2/total_moles;
43 x_O2 = moles_O2/total_moles;
44
45 printf("\n\nCO2 = %f",x_CO2*100)
46 printf("\n\nSO2 = %f",x_SO2*100)
47 printf("\n\nN2 = %f",x_N2*100)
48 printf("\n\nO2 = %f",x_O2*100)

```

Scilab code Exa 3.23 Coal combustion

```
1  clc
2  clear
3  printf("example 3.23 page number 110\n\n")
4
5  //to find the composition of flue gas
6
7  C = 0.8    //in kg
8  H2 = 0.05  //in kg
9  S = 0.005  //in kg
10 ash = 0.145 //in kg
11
12 //required oxygen in kg
13 C_O2 = C*(32/12);
14 H2_O2 = H2*(16/2);
15 S_O2 = S*(32/32);
16 O2_supplied = C_O2+S_O2+H2_O2;
17 printf("amount of O2 supplied = %f kg\n\n",
        O2_supplied)
18
19 wt_air = O2_supplied*(100/23);
20 wt_airsupplied = 1.25*wt_air;
21 printf("amount of air supplied = %f kg\n\n",
        wt_airsupplied)
22
23 //flue gas composition
24 m_N2 = wt_airsupplied*0.77;    //in kg
25 mole_N2 = m_N2/28;
26
27 m_O2 = (wt_airsupplied-wt_air)*0.23;    //in kg
28 mole_O2 = m_O2/32;
29
30 m_CO2 = C*(44/12);    //in kg
```

```

31 mole_CO2 = m_CO2/44;
32
33 m_H2O = H2*(18/2); //in kg
34 mole_H2O = m_H2O/18;
35
36 m_SO2 = S*(64/32); //in kg
37 mole_SO2 = m_SO2/64;
38
39 m = m_N2+m_O2+m_CO2+m_H2O+m_SO2
40
41 //percent by weight
42 w_N2 = m_N2/m;
43 printf("percentage of N2 by weight = %f\n\n",w_N2
    *100)
44
45 w_O2 = m_O2/m;
46 printf("percentage of O2 by weight = %f\n\n",w_O2
    *100)
47
48 w_CO2 = m_CO2/m;
49 printf("percentage of CO2 by weight = %f\n\n",w_CO2
    *100)
50
51 w_H2O = m_H2O/m;
52 printf("percentage of H2O by weight = %f\n\n",w_H2O
    *100)
53
54 w_SO2 = m_SO2/m;
55 printf("percentage of SO2 by weight = %f\n\n",w_SO2
    *100)
56
57 m1 = mole_N2+mole_O2+mole_CO2+mole_H2O+mole_SO2
58
59 //percent by mole
60 x_N2 = mole_N2/m1;
61 printf("percentage of N2 by mole = %f\n\n",x_N2*100)
62
63 x_O2 = mole_O2/m1;

```

```

64 printf("percentage of O2 by mole = %f\n\n",x_O2*100)
65
66 x_CO2 = mole_CO2/m1;
67 printf("percentage of CO2 by mole = %f\n\n",x_CO2
        *100)
68
69 x_H2O = mole_H2O/m1;
70 printf("percentage of H2O by mole = %f\n\n",x_H2O
        *100)
71
72 x_SO2 = mole_SO2/m1;
73 printf("percentage of SO2 by mole = %f\n\n",x_SO2
        *100)

```

Scilab code Exa 3.24 Petrol combustion

```

1  clc
2  clear
3  printf("example 3.24 page number 112\n\n")
4
5  //to find volumetric composition of flue glass
6
7  wt_H2 = 0.15;
8  wt_C = 0.85;
9  O2_H2 = wt_H2*(16/2);
10 O2_C = wt_C*(32/12);
11
12 total_O2 = O2_H2+O2_C;
13
14 wt_air = total_O2/0.23;
15
16 air_supplied = 1.15*(wt_air);
17 N2_supplied = 0.77*air_supplied/28;
18 O2_supplied = 0.23*(air_supplied-wt_air)/32;
19 moles_CO2 = 0.85/12;

```

```

20
21 printf(" moles of CO2 = %f kmol\n\n",moles_CO2)
22 printf(" moles of N2 = %f kmol \n\n",N2_supplied)
23 printf(" moles of O2 = %f kmol\n\n",O2_supplied)
24
25 total_moles = N2_supplied+O2_supplied+moles_CO2;
26
27 printf("percentage of CO2 = %f\n\n",(moles_CO2/
    total_moles)*100)
28 printf("percentage of N2 = %f\n\n",(N2_supplied/
    total_moles)*100)
29 printf("percentage of O2 = %f", (O2_supplied/
    total_moles)*100)

```

Scilab code Exa 3.25 Air supply

```

1  clc
2  clear
3  printf("example 3.25 page number 113\n\n")
4
5  //to find the excess air supplied
6
7  N2 = 80.5    //in m3
8  air_supplied = N2/0.79    //in m3
9  volume_O2 = air_supplied*0.21;    //in m3
10 O2_fluegas = 6.1    //in m3
11
12 O2_used = volume_O2 - O2_fluegas;
13 excess_air_supplied = (O2_fluegas/O2_used)*100;
14
15 printf("percentage of excess air supplied = %f",
    excess_air_supplied)

```

Scilab code Exa 3.26 CO2 cooling

```
1  clc
2  clear
3  printf("example 3.26 page number 114\n\n")
4
5  //to find the outlet temperature of water
6
7  q_NTP = 10*(200/101.3)*(273/313);
8  m_CO2 = 44*(q_NTP/22.4);
9  s_CO2 = 0.85    //in kJ/kg K
10
11 Q = m_CO2*s_CO2*(40-20)    //Q = ms*delta_T
12
13 d0 = 0.023    //in mm
14 A0 = (3.14/4)*d0^2;
15 di = 0.035    //in mm
16 Ai = (3.14/4)*di^2;
17
18 A_annular = Ai-A0;
19 u = 0.15    //in m/s
20 m_water = A_annular*(u*3600)*1000    //in kg/hr
21
22 s_water = 4.19    //in kJ/kg K
23 t = 15+(Q/(m_water*s_water));
24
25 printf("exit water temperature = %f degree C",t)
```

Scilab code Exa 3.27 Heating area

```
1  clc
2  clear
3  printf("example 3.27 page number 114\n\n")
4
5  //to find the area of heating surface
```

```

6 F = 1000 //in kg
7 xF = 0.01
8
9 solid_feed = F*xF;
10 water_feed = F - solid_feed;
11
12 tF = 40 //in degree C
13 hF = 167.5 //in kJ/kg
14 xL = 0.02;
15
16 solid_liquor = 10 //in kg
17 L = solid_liquor/xL;
18 tL = 100 //in degree C
19 hL = 418.6 //in kJ/kg
20
21 V = F -L;
22
23 tv = 100 //in degree C
24 Hv = 2675 //in kJ/kg
25 ts = 108.4 //in degree C
26 Hs = 2690 //in kJ/kg
27 tc = 108.4 //in degree C
28 hc = 454 //in kJ/kg
29
30 //applying heat balance
31 S = (F*hF-V*Hv-L*hL)/(hc-Hs);
32 printf("weight of steam required = %f kg/hr",S)
33
34 Q = S*(Hs-hc);
35 U = 1.4 //in kW/m2K
36 delta_t = ts-tL;
37 A = 383.2/(U*delta_t);
38 printf("\n\narea of heating surface = %f square
meter",A)

```

Scilab code Exa 3.28 Distillation column

```
1  clc
2  clear
3  printf("example 3.28 page number 115\n\n")
4
5  //to find the top and bottom product ,condenser duty ,
   heat input to rebpoiler
6  hF = 171 //in kJ/kg
7  hD = 67 //in kJ/kg
8  hL = hD;
9
10 hW = 200 //in kJ/kg
11 H = 540 //in kJ/kg
12
13 disp('part 1')
14 F = 1000 //in kg/h
15 xF = 0.40
16 xW = 0.02;
17 xD = 0.97;
18 D = F*(xF-xW)/(xD-xW);
19 W = F-D;
20
21 printf("bottom product = %f kg/hr",W)
22 printf("\ntop product = %f kg/hr\n\n",D)
23
24 disp('part 2')
25 L = 3.5*D;
26 V = L+D;
27 Qc = V*H-L*hL-D*hD;
28 printf("condenser duty = %f KJ/hr\n\n",Qc)
29
30 disp('part 3')
31 Qr = Qc - 24200;
32 printf("rate of heat input to reboiler = %f kJ/hr",
   Qr)
```

Scilab code Exa 3.29 Crystallization

```
1  clc
2  clear
3  printf("example 3.29 page number 117\n\n")
4
5  //to find the rate of crystal formation , cooling
   water rate , required area
6
7  F = 1000;    //in kg
8  V = 0.05*F; //in kg
9  xF = 0.48;
10 xL = 75/(100+75);
11 xC = 1;
12 C = (F*xF-950*xL)/(1-0.429);
13 printf ("rate of crystal formation = %f kg",C)
14
15 L = F-C-V;
16
17 //cooling water
18 W = (F*2.97*(85-35)+126.9*75.2-V*2414)/(4.19*11);
19 printf("\n\nrate of cooling water = %f kg",W)
20
21 delta_T1 = 56;
22 delta_T2 = 17;
23 delta_Tm = (delta_T1-delta_T2)/(log(delta_T1/
   delta_T2))
24 U = 125;
25
26 A=(F*2.97*(85-35)+126.9*75.2-V*2414)/(U*delta_Tm
   *3.6);
27 printf("\n\narea = %f square meter",A)
```

Scilab code Exa 3.30 Combustion

```
1  clc
2  clear
3  printf("example 3.30 page number 118\n\n")
4
5  //to find the heat of combustion
6
7  delta_n = 10-12; //mole per mole naphthalene
8
9  //basis 1g
10 moles_napthalene = (1/128);
11
12 disp('part 1')
13 Qv = 40.28 //in kJ
14 Qp = Qv-(delta_n*moles_napthalene*8.3144*298/1000);
15 printf("heat of combustion = %f kJ\n\n",Qp)
16
17 disp('part 2')
18 delta_H = 44.05 //in kJ/gmol
19 water_formed = 4/128; //in g mol
20 Qp1 = Qp - (delta_H*water_formed);
21 printf("heat of combustion = %f kJ",Qp1)
```

Chapter 4

Flow Of Fluids

Scilab code Exa 4.1 Water compressibility

```
1 clc
2 clear
3 printf("example 4.1 page number 125\n\n")
4
5 //to find water compressibility
6 delta_p=70; //in bar
7 Et=20680 //in bar
8 compressibility = delta_p/Et;
9 printf("compressibility of water = %f",
    compressibility)
```

Scilab code Exa 4.2 Isothermal Compressibility

```
1 clc
2 clear
3 printf("example 4.2 page number 125\n\n")
4
5 disp("this is a theoretical problem, book shall be
    referred for solution")
```

Scilab code Exa 4.3 Viscosity

```
1 clc
2 clear
3 printf("example 4.3 page number 128\n\n")
4
5 //to find the viscosity of oil
6
7 F=0.5*9.8; //in N
8 A=3.14*0.05*0.15; //in m2
9 shear_stress=F/A; //in Pa
10 printf("shear_stress = %f Pa",shear_stress)
11
12 velocity_distribution =0.1/(0.05*10^-3);
13 viscosity=shear_stress/velocity_distribution;
14 printf("\n\nviscosity = %f Pa-s",viscosity)
```

Scilab code Exa 4.4 Streamline flow

```
1 clc
2 clear
3 printf("example 4.4 page number 130\n\n")
4 printf("this is a theoretical problem ,book shall be
   referred for solution")
```

Scilab code Exa 4.5 Frictional losses

```
1 clc
2 clear
```

```
3 printf("example 4.5 page number 133\n\n")
4
5 //to find variation of losses with velocity
6 loss_ratio=3.6; //delta_P2/delta_P1=3.6
7 velocity_ratio=2; //u2/u1=2
8 n=log2(loss_ratio); //delta_P2/delta_P1=(u2/u1)^n
9 printf("power constant = %f flow is turbulent",n)
```

Scilab code Exa 4.6 Velocity profile

```
1 clc
2 clear
3 printf("example 4.6 page number 133\n\n")
4 printf("this is a theoretical problem,book shall be
   referred for solution")
```

Scilab code Exa 4.7 Velocity profile

```
1 clc
2 clear
3 printf("example 4.7 page number 134")
4 disp("this is a theoretical problem,book shall be
   referred for solution")
```

Scilab code Exa 4.8 Boundary layer

```
1 clc
2 clear
3 printf("example 4.8 page number 137\n\n")
4
```

```

5 //to find the boundary layer properties
6
7 disp('part 1')
8 x=0.05 //in m
9 density=1000 //in kg/m3
10 viscosity=1*10^-3 //in Pa-s
11 u=1 //in m/s
12 Re=(density*u*x)/viscosity;
13
14 printf("Reynolds Number = %f",Re)
15
16 thickness=4.65*x*(Re)^-0.5;
17 printf("\nboundary layer thickness = %f m\n",
    thickness)
18
19 disp('part 2')
20 Re_x=3.2*10^5;
21 x_cr=(Re_x*viscosity)/(density*u);
22 printf("transition takes place at x = %f m\n",x_cr)
23
24 disp('part 3')
25 x=0.5 //in m
26 Re=(density*u*x)/viscosity;
27 thickness=0.367*x*(Re)^-0.2;
28 printf("boundary layer thickness= %f m",thickness)
29
30 t_sublayer=71.5*x*(Re)^-0.9;
31 printf("\nsub layer thickness= %f m",t_sublayer)

```

Scilab code Exa 4.9 Pipe flow

```

1 clc
2 clear
3 printf("example 4.9 page number 138\n\n")
4

```

```

5 //to find the flow properties
6 d1=0.05 //in m
7 A1=(3.14*d1^2)/4;
8 density_1=2.1 //in kg/m3
9 u1=15 //in m/s
10 P1=1.8; //in bar
11 P2=1.3; //in bar
12
13 w=density_1*A1*u1;
14 density_2=density_1*(P2/P1);
15 printf("density at section 2 = %f kg/cubic meter",
        density_2)
16
17 u2=u1*(density_1/density_2)*(0.05/0.075)^2;
18 printf("\n\nvelocity at section 2 = %f m/s",u2)

```

Scilab code Exa 4.10 Temperature rise

```

1 clc
2 clear
3 printf("example 4.10 page number 139\n\n")
4
5 //to find the temperature increase
6
7 Q=0.001*10^5 //in J/s
8 w=0.001*1000 //in kg/s
9 density=1000 //in kg/m3
10 cp=4.19*10^3 //in J/kg K
11
12 delta_T=Q/(w*cp);
13 printf("Temperature increase = %f degree celcius",
        delta_T)

```

Scilab code Exa 4.11 Bernoulli equation

```
1  clc
2  clear
3  printf("example 4.11 page number 142\n\n")
4
5  //to find the pressure
6
7  u1=0;    //in m/s
8  ws=0;
9  P1=0.7*10^5    //in Pa
10 P3=0
11 density=1000    //in kg/m3
12
13 u3=((2*(P1-P3))/density)^0.5;
14 printf("u3 = %f m/s",u3)
15
16 ratio_area=0.5;
17 u2=u3/ratio_area;
18 printf("\n\nu2 = %f m/s",u2)
19
20 //applying bernoulli's equation
21 P2=1.7*10^5-((density*u2^2)/2)
22 printf("\n\nP2 = %f Pa",P2)
23 printf("\n\nthis flow is physically unreal")
```

Scilab code Exa 4.12 Power requirements

```
1  clc
2  clear
3  printf("example 4.12 page number 143\n\n")
4
5  //to find the power requirements
6
7  Q=3800/(24*3600)    //in m3/s
```

```

8 d=0.202      //in m
9
10 u=Q/((3.14/4)*d^2);    //in m/s
11 delta_P=5.3*10^6      //in Pa
12 density=897          //in kg/m3
13 F=delta_P/density;    //in J/kg
14 ws=9.8*30+F;
15 mass_flow_rate= Q*density;
16 power=(ws*mass_flow_rate)/0.6;
17
18 printf("power required = %f kW",power/1000)

```

Scilab code Exa 4.13 Hagen Poiseulle equation

```

1 clc
2 clear
3 printf("example 4.13 page number 146\n\n")
4
5 //to find the tube length
6 density=1000      //in kg/m3
7 viscosity=1*10^-3 //in Pa s
8 P=100*1000       //in Pa
9
10 vdP=P/density;
11
12 Q=2.5*10^-3/(24*3600)
13 A=3.14*(0.0005)^2/4;
14 u=Q/A;
15 printf("u = %f m/s",u)
16
17 Re=density*u*0.0005/viscosity;
18 printf("\n\nRe = %f",Re)
19
20 //F=18.86*L
21 L=(-u^2+vdP)/18.86;

```



```
22 printf("\n\nL = %f m",L)
```

Scilab code Exa 4.14 Pressure Head calculation

```
1 clc
2 clear
3 printf("example 4.14 page number 151\n\n")
4
5 //to find the discharge pressure
6 d=0.025 //in m
7 u=3 //in m/s
8 density=894 //in kg/m3
9 viscosity=6.2*10^4 //in Pa-s
10
11 Re=(u*d*density)/viscosity;
12 f=0.0045;
13 L=50;
14
15 delta_P=2*f*density*u^2*(L/d)
16 printf("frictional head loss = %f kPa",delta_P/1000)
17
18 required_P=25*density*9.8;
19 total_head=delta_P+required_P;
20 printf("\n\ntotal pressure head = %f bar",total_head
    /10^5)
```

Scilab code Exa 4.15 Level difference calculation

```
1 clc
2 clear
3 printf("example 4.15 page number 152\n\n")
4
5 //to find the level difference
```

```

6
7 Q=0.8*10^-3;    //in m3/s
8 d=0.026        //in m
9 A=(3.14*(d^2))/4    //in m2
10
11 u=Q/A;        //in m/s
12 density=800    //in kg/m3
13 viscosity=0.0005    //in Pa-s
14
15 Re=(u*density*d)/viscosity;
16 f=0.079*(Re)^-0.25;
17 L=60
18 h_f=2*f*((u^2)/9.8)*(L/d);
19
20 printf("level difference = %f m",h_f)

```

Scilab code Exa 4.16 Energy cost calculation

```

1 clc
2 clear
3 printf("example 4.16 page number 153\n\n")
4
5 //to find the engery cost
6 delta_z=50;    //in m
7 L=290.36      //in m
8 d=0.18        //in m
9 Q=0.05        //in m3/s
10
11 A=(3.14*d^2)/4;    //in m2
12 u=Q/A;        //in m/s
13 density=1180;    //in kg/m3
14 viscosity=0.0012    //in Pa-s
15 Re=u*density*d/viscosity;
16
17 f=0.004;

```

```

18 sigma_F=2*f*u^2*L/d;
19 ws=((9.8*50)+sigma_F)/0.6;
20 mass_flow_rate=Q*density; //in Kg/s
21 power=mass_flow_rate*ws/1000; //in KW
22 energy_cost=power*24*0.8;
23
24 printf("Energy cost = Rs %f",energy_cost)

```

Scilab code Exa 4.17 Pressure loss

```

1 clc
2 clear
3 printf("example 4.17 page number 154\n\n")
4
5 //to find the pressure loss
6 density=998 //in kg/m3
7 viscosity=0.0008 //in Pa-s
8 d=0.03 //in m
9 u=1.2 //in m/s
10
11 Re=density*d*u/viscosity;
12
13 f=0.0088;
14 D=1 //in m
15 N=10
16 L=3.14*D*N;
17 delta_P=(2*f*u^2*L)/d; //in Pa
18 delta_P_coil=delta_P*(1+(3.54*(d/D)));
19
20 printf("frictional pressure drop = %f kPa",
    delta_P_coil)

```

Scilab code Exa 4.18 Pressure gradient

```

1  clc
2  clear
3  printf("example 4.18 page number 154\n\n")
4
5  //to find pressure drop per unit length
6
7  b=0.050    //in m
8  a=0.025    //in m
9  d_eq=b-a   //in m
10 density=1000 //in kg/m3
11 u=3        //in m/s
12 viscosity = 0.001
13
14 Re=d_eq*u*density/viscosity;
15
16 e=40*10^6   //in m
17 f=0.0062;
18 P_perunit_length=2*f*density*u^2/d_eq; //in Pa/m
19
20 printf("pressure per unit length = %f Pa/m",
        P_perunit_length)

```

Scilab code Exa 4.19 Flow rate

```

1  clc
2  clear
3  printf("example 4.19 page number 155\n\n")
4
5  //to find the flow rate
6  d = 0.3    //in m
7  u = 17.63  //avg velocity in m/s
8
9  q = (3.14/4)*d^2*u;
10 printf("volumetric flow rate = %f cubic meter per
        second",q)

```

Scilab code Exa 4.20 Pipe dimensions

```
1 clc
2 clear
3 printf("example 4.20 page number 156\n\n")
4
5 //to find the size of pipe required
6 d = 0.15 //in m
7 u = (0.0191/0.15^2); //in m/s
8
9 q = (3.14/4)*d^2*u;
10 printf("volumetric flow rate = %f cubic meter/s",q)
```

Scilab code Exa 4.21 Pressure gradient

```
1 clc
2 clear
3 printf("example 4.21 page number 160\n\n")
4
5 //to find the pressure gradient
6
7 Q=0.0003 //in m3/s
8 d=0.05 //in m
9 A=(3.14*d^2)/4;
10
11 u=Q/A;
12
13 density=1000; //in kg/m3
14 viscosity=0.001; //in Pa-s
15 e=0.3;
16 dp=0.00125; //particle diameter in m
```

```

17
18 Re=(dp*u*density)/(viscosity*(1-e));
19 fm=(150/Re)+1.75;
20 L=0.5 //in m
21 delta_Pf=fm*((density*L*u^2)/dp)*((1-e)/e^3); //in
    Pa
22
23 //applying bernoulli's equation, we get
24 delta_P=delta_Pf-(density*9.8*L);
25 pressure_gradient=delta_P/(L*1000); //in kPa/m
26 printf("required pressure gradient = %f kPa/m of
    packed height",pressure_gradient)

```

Scilab code Exa 4.22 Minimum fluidization velocity

```

1 clc
2 clear
3 printf("example 4.22 page number 163\n\n")
4
5 //to find minimum fluidization velocity
6
7 d=120*10^-6 //in m
8 density=2500 //particle density in kg/m3
9 e_min=0.45;
10 density_water=1000 //in kg/m3
11 viscosity=0.9*10^-3; //in Pa-s
12 umf=(d^2*(density-density_water)*9.8*e_min^3)/(150*
    viscosity*(1-e_min));
13 printf("minimum fluidization velocity = %f m/s",umf)
14
15 Re_mf=(d*umf*density_water)/(viscosity*(1-e_min));
16
17
18 //given that uo/umf=10
19 function [f] = F(e)

```

```

20     f = e^3+1.657*e-1.675;
21 endfunction
22
23 //initial guess
24 x = 10;
25 e = fsolve(x,F);
26
27 printf("\n\ne = %f",e)
28 length_ratio=(1-e_min)/(1-e);
29 printf("\n\nratio of heights = %f",length_ratio)

```

Scilab code Exa 4.23 Pumping of fluids

```

1  clc
2  clear
3  printf("example 4.23 page number 167\n\n")
4
5  //to find the power requirements
6
7  P=9807 //in Pa
8  density=1000 //in kg/m3
9  Q=250/(60*density)
10 head=25 //in m
11
12 w= head*Q*P; //in kW
13 power_delivered=w/0.65;
14 power_taken=power_delivered/0.9;
15
16 printf("power_delivered = %f kW",power_delivered
17 /1000)
17 printf("\n\npower taken by motor = %f kW",
18 power_taken/1000)

```

Chapter 5

Heat Transfer

Scilab code Exa 5.1 Heat conduction

```
1 clc
2 clear
3 printf("example 5.1 page number 171\n\n")
4
5 //to find the rate of heat loss
6 A=5*4 //in m2
7 T1=100; //in K
8 T2=30; //in K
9
10 delta_T=T1-T2;
11
12 x=0.25 //in m
13 k=0.70 //in W/mK
14 Q=k*A*(delta_T/x);
15
16 printf("rate of heat loss = %f W",Q)
```

Scilab code Exa 5.2 Heat conduction


```

1  clc
2  clear
3  printf("example 5.1 page number 171\n\n")
4
5  //to find the heat loss
6
7  d1=0.15    //in m
8  d2=0.16    //in m
9  l=1       //in m
10
11  A1=3.14*d1*l;
12  A2=3.14*d2*l
13  Am=(A1-A2)/log (A1/A2);
14
15  T1=120;    //in K
16  T2=119.8; //in K
17
18  delta_T=T1-T2;
19  x=(d2-d1)/2;
20  k=50      //in W/mK
21  Q=k*Am*(delta_T/x);
22
23  printf("rate of heat loss per unit length = %f W/m",
        Q)

```

Scilab code Exa 5.3 Heat conduction through sphere

```

1  clc
2  clear
3  printf("example 5.3 page number 172\n\n")
4
5  //to find the rate of heat loss
6
7  ri=0.5     //in m
8  ro=0.6;    //in m

```

```

 9 A1=4*3.14*ri^2;
10 A2=4*3.14*ro^2;
11
12 Am=(A1*A2)^0.5;
13
14 Ti=140;    //in K
15 To=50;    //in K
16 delta_T=Ti-To;
17 x=0.1     //in m
18 k=0.12    //in W/mK
19
20 Q=k*Am*(delta_T/x);
21 printf("Heat loss through sphere = %f W",Q)

```

Scilab code Exa 5.4 Composite wall

```

1  clc
2  clear
3  printf("example 5.4 page number 173\n\n")
4
5  //to find the heat loss from composite wall
6  //for the red brick layer
7
8  x1=0.250; //in m
9  k1=0.7;   //in W/mK
10 A1=1;     //in m2
11 R1=x1/(k1*A1); //in K/W
12
13 //for the felt layer
14 x2=0.020; //in m
15 k2=0.046; //in W/mK
16 A2=1;     //in m2
17 R2=x2/(k2*A2); //in K/W
18 R=R1+R2;
19 printf("Total resistance = %f K/W",R)

```

```

20
21 T1=110;    //in K
22 T2=25     //in K
23 delta_T=T1-T2;
24 Q=delta_T/R;
25 printf("\n\nheat loss through wall = %f W/square m",
        Q)

```

Scilab code Exa 5.5 Composite Pipeline

```

1  clc
2  clear
3  printf("example 5.5 page number 173\n\n")
4
5  //to find the rate of heat loss through pipeline
6  //resistance by pipeline
7
8  d1=0.15    //in m
9  d2=0.16    //in m
10 l=1    //in m
11 A1=3.14*d1*l;
12 A2=3.14*d2*l
13 Am1=(A2-A1)/log (A2/A1);
14 x1=(d2-d1)/2;
15 k1=50    //in W/mK
16 R1=x1/(k1*Am1);
17
18 //resistance by insulation
19 d2=0.16    //in m
20 d3=0.26    //in m
21 l=1    //in m
22 A2=3.14*d2*l;
23 A3=3.14*d3*l
24 Am2=(A3-A2)/log (A3/A2);
25 x2=(d3-d2)/2;

```

```

26 k2=0.08    //in W/mK
27 R2=x2/(k2*Am2);
28 R=R1+R2;
29
30 printf("total resistance = %f K/W",R)
31
32 T1=120;    //in K
33 T2=40;    //in K
34 delta_T=T1-T2;
35 Q=delta_T/R;
36
37 printf("\n\nheat loss = %f W/m",Q)

```

Scilab code Exa 5.6 Parellel Resistance

```

1  clc
2  clear
3  printf("example 5.6 page number 174\n\n")
4
5  //to find the increase in heat transfer rate
6
7  x1=0.1;    //in m
8  x2= 0.25;  //in m
9  k_rb=0.93; //in W/mK
10 k_ib=0.116 //in W/mK
11 k_al=203.6 //in W/mK
12 A=0.1     //in m2
13
14 //to find resistance without rivets
15 R=(1/A)*((x1/k_rb)+(x2/k_ib));
16 T1=225    //in K
17 T2=37     //in K
18 delta_T=T1-T2;
19 Q=delta_T/R;
20 printf("heat transfer rate = %f W",Q)

```

```

21
22 //to find resistance with rivet
23 d=0.03 //in m
24 rivet_area= (3.14/4)*d^2;
25 R_r=(x1+x2)/(k_al*rivet_area);
26 area_norivet=A-rivet_area;
27 R_cl=(A/area_norivet)*R;
28 R_eq=1/(1/R_r+1/R_cl);
29 Q_new=delta_T/R_eq;
30
31 printf("\n\nRate of heat transfer with rivet = %f W"
, Q_new)
32 increase=((Q_new-Q)/Q)*100;
33 printf("\n\npercentage increase in heat transfer
rate = %f", increase)

```

Scilab code Exa 5.7 Heat transfer coefficient

```

1 clc
2 clear
3 printf("example 5.6 page number 174\n\n")
4
5 //to find the increase in heat transfer rate
6
7 x1=0.1; //in m
8 x2= 0.25; //in m
9 k_rb=0.93; //in W/mK
10 k_ib=0.116 //in W/mK
11 k_al=203.6 //in W/mK
12 A=0.1 //in m2
13
14 //to find resistance without rivets
15 R=(1/A)*((x1/k_rb)+(x2/k_ib));
16 T1=225 //in K
17 T2=37 //in K

```

```

18 delta_T=T1-T2;
19 Q=delta_T/R;
20 printf("heat transfer rate = %f W",Q)
21
22 //to find resistance with rivet
23 d=0.03 //in m
24 rivet_area= (3.14/4)*d^2;
25 R_r=(x1+x2)/(k_al*rivet_area);
26 area_norivet=A-rivet_area;
27 R_cl=(A/area_norivet)*R;
28 R_eq=1/(1/R_r+1/R_cl);
29 Q_new=delta_T/R_eq;
30
31 printf("\n\nRate of heat transfer with rivet = %f W"
, Q_new)
32 increase=((Q_new-Q)/Q)*100;
33 printf("\n\npercentage increase in heat transfer
rate = %f",increase)

```

Scilab code Exa 5.8 Heat transfer coefficient

```

1 clc
2 clear
3 printf("example 5.8 page number 188\n\n")
4
5 //to find the heat transfer coefficient
6 density=984.1 //in kg/cubic meter
7 v=3 //in m/s
8 viscosity=485*10^-6; //in Pa-s
9 k=0.657 //in W/mK
10 cp=4178 //in J/kg K
11 d=0.016 //in m
12
13 Re=(density*v*d)/viscosity;
14 Pr=(cp*viscosity)/k;

```

```

15
16 //dittus boelter equation
17 h=0.023*Re^0.8*Pr^0.3*(k/d);
18 printf("heat transfer coefficient = %f W/sq meter K"
        ,h)
19
20 //Sieder Tate equation
21 viscosity_w=920*10^-6
22 h1=0.023*Re^0.8*Pr^(1/3)*(k/d)*(viscosity/
        viscosity_w)^0.14;
23 printf("\n\nheat transfer coefficient = %f W/sq
        meter K",h1)

```

Scilab code Exa 5.9 Earth Temperature

```

1 clc
2 clear
3 printf("example 5.9 page number 191\n\n")
4
5 //to find the surface temperature of earth
6 T_sun = 5973 //in degree C
7 d = 1.5*10^13 //in cm
8 R = 7.1*10^10; //in cm
9
10 T_earth = ((R/(2*d))^0.5)*T_sun;
11 printf("Temperature of earth = %f C",T_earth-273)

```

Scilab code Exa 5.10 Earth Temperature

```

1 clc
2 clear
3 printf("example 5.10 page number 191\n\n")
4

```

```

5 //to find temperature of earth
6 R=7*10^10; //in cm
7 Ts=6000; //in K
8 l=1.5*10^13; //in m
9 To=((R^2/(4*l^2))^0.25)*Ts;
10 printf("temperature of earth = %f K",To)

```

Scilab code Exa 5.11 Equilibrium temperature

```

1 clc
2 clear
3 printf("example 5.11 page number 192\n\n")
4
5 //to find the equilibrium temperature
6 R=6.92*10^5 //in km
7 l=14.97*10^7 //in km
8 Ts=6200; //in K
9 To=(R^2/l^2)^0.25*Ts;
10 printf("Equilibrium temperature = %f K",To)

```

Scilab code Exa 5.12 Equilibrium temperature

```

1 clc
2 clear
3 printf("example 5.12 page number 192\n\n")
4
5 //to find the equilibrium temperature
6 view_factor=0.5;
7 R=6.92*10^5 //in km
8 l=14.97*10^7 //in km
9 Ts=6200; //in K
10 To=(view_factor*(R^2/l^2))^0.25*Ts;
11 printf("Equilibrium temperature = %f K",To)

```

Scilab code Exa 5.13 Temperature calculation

```
1 clc
2 clear
3 printf("example 5.13 page number 193\n\n")
4
5 //to find the surface temperature
6 view_factor=0.25;
7 R=7.1*10^10 //in cm
8 l=1.5*10^13 //in cm
9 Ts=5973; //in K
10 alpha=0.2;
11 epsilon=0.1;
12
13 ratio=alpha/epsilon;
14 To=(ratio*view_factor*(R^2/l^2))^0.25*Ts;
15 printf("Equilibrium temperature = %f K",To)
```

Scilab code Exa 5.14 Solar constant

```
1 clc
2 clear
3 printf("example 5.14 page number 193\n\n")
4
5 //to find the solar constant
6 R=7*10^10; //in cm
7 l=1.5*10^13; //in cm
8 sigma=5.3*10^-5; //in erd/s(cm2)(K)4
9 T=6000; //in K
10
11 S=(R/l)^2*(sigma)*(T^4)*60;
12 printf("solar constant = %f J/sq cm min",S/10^7)
```

Scilab code Exa 5.15 Evaporator

```
1 clc
2 clear
3 printf("example 5.15 page number 207\n\n")
4
5 //to find the amount of vapor and liquid and amount
  of heat transfer
6
7 F = 5000 //in kg/hr
8 xF = 0.01
9 xL = 0.02;
10
11 L = F*xF/xL;
12 V = F-L;
13 printf("L = %f Kg/hr\n V = %f kg/hr" ,L,V)
14
15 TF= 303 //in K
16 hF = 125.9 //in KJ/kg
17 T1 = 373.2 //in K
18 Hv = 2676.1 //in kJ/kg
19 hL = 419.04; //in kJ/kg
20 Ts = 383.2 //in K
21 Hs = 2691.5 //in kJ/kg
22 hs = 461.30 //in kJ/kg
23
24 S = (F*hF-L*hL-V*Hv)/(hs-Hs);
25 printf("\n\n amount of steam = %f kg steam/h" ,S)
26
27 q = S*(Hs - hs);
28 q = q*1000/3600 //conversion to Watt
29 U = q/(69.9*10);
30 printf("\n\n heat reansfer coefficient = %f W/sq m K"
  ,U)
```

Scilab code Exa 5.16 Evaporator

```
1 clc
2 clear
3 printf("example 5.16 page number 208\n\n")
4
5 //to find the amount of liquid and vapor leaving and
   outlet concentration
6 //we have two linear equations in L and V so we will
   write them in form of a matrix and then solve
   using principles of linear algebra
7
8 b1 = 6000*125.79+3187.56*2691.5-3187.56*461.30;
   //data from previous problem
9 b2 = 6000;
10 A = [419.04 2676.1;1 1];
11
12 b = [b1;b2];
13 x = A\b;
14 L = x(1);
15 V = x(2);
16
17 printf("L = %f kg/hr\nV = %f kg/hr",L,V)
18
19 F = 6000 //in kg/hr
20 xF = 0.01;
21 xL = F*xF/L;
22 printf(" \n\npercentage increase in outlet
   concentration = %f",xL*100)
```

Scilab code Exa 5.17 Evaporator

```

1  clc
2  clear
3  printf("example 5.17 page number 209\n\n")
4
5  //to find the change in heat trnasfer area
6
7  Hv=2635.3      //kJ/kg
8  hL=313.93     //in kJ/kg
9  S=(2500*313.93+2500*2635.3-5000*125.79)
      /(2691.5-461.30);
10 printf("steam flow rate = %f kg steam/hr",S)
11
12 q = S*(2691.5 - 461.30);
13 q = q*1000/3600      //in W
14 U = 2833.13;        //in W/m2 K
15 delta_T = 383.2-348.2;    //in K
16 A = q/(U*delta_T);
17
18 printf("\n\nArea = %f sq meter",A)
19 printf("\n\nin this case a condensor and vaccum pump
      should be used")

```

Chapter 6

Mass Transfer

Scilab code Exa 6.1 Diffusivity

```
1 clc
2 clear
3 printf("example 6.1 page number 213\n\n")
4
5 printf("This is a theoretical problem and book shall
        be referred for solution")
```

Scilab code Exa 6.2 Absorption

```
1 clc
2 clear
3 printf("example 6.2 page number 214\n\n")
4
5 printf("This is a theoretical problem and book shall
        be referred for solution")
```

Scilab code Exa 6.3 Equimolar counter diffusion

```
1 clc
2 clear
3 printf("example 6.3 page number 215\n\n")
4
5 //to find the flux and pressure difference
6
7 D_AB=6.75*10^-5 //in m2/s
8 Z=0.03 //in m
9 R=8314
10 p_A1=5.5*10^4 //in Pa
11 p_A2=1.5*10^4 //in Pa
12 T=298 //in K
13
14 N_A=D_AB*(p_A1-p_A2)/(R*T*Z);
15 printf("flux = %f kmol/sq m s",N_A)
16
17 //for partial pressure
18 Z=0.02; //in m
19 p_A2=p_A1-((N_A*R*T*Z)/D_AB);
20 printf("\n\npressure = %f Pa",p_A2)
```

Scilab code Exa 6.4 Resistane to diffusion

```
1 clc
2 clear
3 printf("example 6.4 page number 216\n\n")
4
5 //to find the flux of NH3 and equimolar counter
6 //diffusion flux
7
8 Z=0.15 //in m
9 P=1.103*10^5 //in Pa
10 p_A1=1.5*10^4 //in Pa
```

```

10 p_A2=5*10^3    //in Pa
11
12 p_B1=P-p_A1;
13 p_B2=P-p_A2;
14
15 D_AB=2.30*10^-5 //in m2/s
16 R=8314
17 T=298    //in K
18
19 //for non diffusing N2
20 p_BM=(p_B2-p_B1)/log (p_B2/p_B1);
21 N_A=D_AB*(p_A1-p_A2)*P/(R*T*Z*p_BM);
22 printf("flux = %f kmol/sq m s",N_A)
23
24 //for diffusing N2
25 N_A=D_AB*(p_A1-p_A2)/(R*T*Z);
26 printf("\n\nflux = %f kmol/sq m s",N_A)

```

Scilab code Exa 6.5 Vapor diffusion

```

1 clc
2 clear
3 printf("example 6.5 page number 216\n\n")
4 printf("This is a theoretical problem and book shall
   be referred for solution")

```

Scilab code Exa 6.6 Flux of HCl

```

1 clc
2 clear
3 printf("example 6.6 page number 218\n\n")
4
5 M_A=36.5    //molar mass of HCl

```

```

6 M_B=18      //molar masss of water
7 w_A1=12;    //weight % of HCL
8 w_A2=4      //weight % of HCL
9 x_A1=(w_A1/M_A)/((w_A1/M_A)+((100-w_A1)/M_B));
10 printf('x_A1 =%f',x_A1)
11
12 x_B1=1-x_A1;
13 M1=100/((w_A1/M_A)+((100-w_A1)/M_B));
14 printf("\n\nmolar mass at point 1 = %f kg/kmol",M1)
15
16 //at point 2
17 x_A2=(w_A2/M_A)/((w_A2/M_A)+((100-w_A2)/M_B));
18 x_B2=1-x_A2;
19 M2=100/((w_A2/M_A)+((100-w_A2)/M_B));    //avg
    molecular weight at point 2
20 printf("\n\nmolar mass at point 2 = %f Kg/kmol",M2)
21
22 density_1=1060.7;    //in kg/m3
23 density_2=1020.15;  //in kg/m3
24 C_av=((density_1/M1)+(density_2/M2))/2;
25 printf("\n\nC_av = %f kmol/cubic m",C_av)
26
27 x_BM=(x_B2-x_B1)/(log (x_B2/x_B1));
28 Z=0.004    //in m
29 D_AB=2.5*10^-9;
30 N_A=(D_AB*C_av*(x_A1-x_A2))/(x_BM*Z);
31 printf("\n\nflux = %f kmol/sq m-s",N_A)

```

Scilab code Exa 6.7 Vaporization

```

1 clc
2 clear
3 printf("example 6.7 page number 220\n\n")
4
5 printf("This is a theoritical problem and book shall

```


be referred for solution”)

Scilab code Exa 6.8 Gas Absorption

```
1  clc
2  clear
3  printf("example 6.8 page number 229\n\n")
4
5  //to find the mean driving force and mass transfer
   area
6
7  Gs=700/22.4    //in kmol of dry air/hr
8  Ls=1500/18    //in kmol of dry air/hr
9  y1=0.05
10 Y1=y1/(1-y1);
11 Y2=0.02*Y1;
12 X2=0
13 X1=(Gs/Ls)*(Y1-Y2);
14 m=Gs*(Y1-Y2);
15
16 //driving force
17 delta_Y1=Y1-1.68*X1;
18 delta_Y2=Y2-1.68*X2;
19 delta_Y=(delta_Y1-delta_Y2)/(log (delta_Y1/delta_Y2)
   );
20 printf("driving force = %f kmol acetone/kmol dry air
   ",delta_Y)
21
22 //mass transfer area
23 K_G=0.4    //in kmol acetone/kmol dry air
24 A=m/(K_G*delta_Y);
25 printf("\n\narea = %f sq m",A)
```

Scilab code Exa 6.9 Equilibrium Composition

```
1  clc
2  clear
3  printf("example 6.9 page number 229\n\n")
4
5  //to calculate minimum oil circulation rate
6
7  G1=(855/22.4)*(106.6/101.3)*(273/299.7);
8  y1=0.02;
9  Y1=y1/(1-y1);
10 Gs=G1*(1-y1);
11
12 //for 95% removal
13 Y2=0.05*Y1;
14 x2=0.005;
15 X2=x2/(1-x2);
16 Y=0.204;
17 X1=0.176;    //in kmol bgenzene/kmol benzene free
                oil
18
19 Ls_molar=(Gs*(Y1-Y2))/(X1-X2);
20 Ls=Ls_molar*260;
21
22 printf("minimum oil circulation rate = %f kg/hr",Ls)
```

Scilab code Exa 6.10 Equilibrium Composition

```
1  clc
2  clear
3  printf("example 6.10 page number 231\n\n")
4
5  // to find the equilibrium composition
6  P_M = 53.32    //kPa
7  P_W = 12.33   //in kpA
```

```

8 P = 40 //IN K pA
9 x = (P - P_W)/(P_M-P_W);
10
11 printf("liquid phase composition = %f",x)
12
13 y = P_M*x/P;
14 printf("\n\nvapor phase composition = %f",y)

```

Scilab code Exa 6.11 Vapor Liquid Equilibrium

```

1 clc
2 clear
3 printf("example 6.11 page number 232\n\n")
4
5 printf("this is a theoretical question , book shall
   be referred for solution")

```

Scilab code Exa 6.12 Distillation Column

```

1 clc
2 clear
3 printf("example 6.12 page number 231\n\n")
4
5 //to find the top and bottom composition
6 x = [1;0.69;0.40;0.192;0.045;0];
7 y = [1;0.932;0.78;0.538;0.1775;0];
8 plot(x,y)
9 xlabel("x")
10 ylabel("y")
11 title("distillation curve")
12 x = 0:0.1:1;
13 y = 0:0.1:1;
14 plot(x,y)

```

```

15 x = [0.5,0.31];
16 y = [0.5,0.7];
17 plot (x,y)
18 Z=0.5;
19 y_D=0.69;
20 x_W=0.31;
21
22
23 printf("composition of top product = %f mole percent
      of hexane",y_D*100)
24 printf("\n\ncomposition of bottom product = %f mole
      percent of hexane",x_W*100)

```

Scilab code Exa 6.13 Distillation

```

1  clc
2  clear
3  printf("example 6.13 page number 237\n\n")
4
5  //to find the composite distillate and residue
6
7  F = 100    //moles
8  xf = 0.4;
9  D = 60    //moles
10 W = 40    //moles
11
12 x = 0.2:0.05:0.45;
13 for i =1:((0.45-0.2)/0.05)+1
14
15 y(i) = 2.16*x(i)/(1+1.16*x(i));
16
17
18 z(i) = (y(i)-x(i))^-1;
19
20 end

```

```

21
22 plot (x,z'/10)
23 title('Batch Distillation Curve')
24 xlabel('x')
25 ylabel('y')
26 xw = 0.22; //from the graph
27 yd = (F*xf-W*xw)/D;
28
29 printf("composition of distillate = %f",yd)
30 printf("\n\ncomposition of residue = %f",xw)

```

Scilab code Exa 6.14 Steam Distillation

```

1 clc
2 clear
3 printf("example 6.14 page number 238\n\n")
4
5 printf('this is a theoretical question and solution
        can be referred from the book')

```

Scilab code Exa 6.15 McCabe Thiele Method

```

1 clc
2 clear
3 printf("example 6.15 page number 249\n\n")
4
5 //to find the top and bottom product composition
6
7 //part 1
8 x=0.4;
9 y=0.8;
10 x_D=y;
11 x_W=0.135; //bottom concentration

```

```

12 D=(100*x-100*x_W)/(y-x_W); // distillate amount
13 printf("amount of distillate =%f moles/h",D)
14
15 //part 2
16 alpha=6; //relative volatility
17 x_R=y/(y+(alpha*(1-y))); //liquid leaving partial
    condensor
18 printf("\n\nliquid leaving partial condenser = %f",
    x_R)
19
20 y1=(1/3)*y+(2/3)*x;
21 x1=y1/(y1+(alpha*(1-y1)));
22 y_W = (1/3)*x_D+(2/3)*x1;
23 x_W=y_W/(y_W+(alpha*(1-y_W)));
24 D=(100*(x-x_W))/(y-x_W);
25
26 printf("\n\namount of distillate = %f moles/h",D)

```

Scilab code Exa 6.16 Liquid liquid extraction

```

1 clc
2 clear
3 printf("example 6.16 page number 264\n\n")
4
5 //to find the percentage extraction of nicotine
6 x=0.01; //% of nicotine
7 X0 = x/(1-x);
8 w=150 //weight of nicotine water solution
9 A0=w*(1-X0);
10 B0=250; //kg keroscene
11 X1 = A0*X0/(A0+B0*0.798);
12 printf("final concentration of nicotine = %f",X1)
13
14 c=A0*(X0-X1);
15 printf("\n\namount of nicotine removed = %f kg",c)

```

```

16
17 percentage = (c*100)/(A0*x);
18 printf("\n\npercentage recovery = %f percent",
    percentage)

```

Scilab code Exa 6.17 Liquid liquid extraction

```

1  clc
2  clear
3  printf("example 6.17 page number 264\n\n")
4
5  //to find the number of stages
6  x=0.01    //mole fraction of nicotine
7  yN = 0.0006;    //mole fraction in solvent
8  xN = 0.001;    //final mole fraction in water
9
10 X0=x/(1-x);    //in kg nicotine/kg water
11 YN =yN/(1-yN);    //in kg nicotine/kg keroscene
12 XN = xN/(1-xN);
13 A0=100*(1-X0);    //kgwater/h
14 B0=150*(1-YN);    //in kg kerosene/h
15
16 Y1=((A0*(X0-XN))/B0)+YN;    //in kg nicotine/kg
    kerosene
17 printf("Y1 = %f kg nicotine/kg kerosene",Y1)
18
19 //for graph refer to the book
20 number_of_stages = 8.4;
21 printf("\n\nnumber of stages = %f",number_of_stages
    )

```

Scilab code Exa 6.18 Humidity calculation

```

1  clc
2  clear
3  printf("example 6.18 page number 274\n\n")
4
5  //to calculate the humidity
6  P = 101.3    //in kPa
7  pA = 3.74    //in kPa
8  p_AS = 7.415 //in kPa
9  H = (18.02/28.97)*(pA/(P-pA));
10 printf("humidity = %f kg H2O/kg air",H)
11
12 Hs = (18.02/28.97)*(p_AS/(P-p_AS));
13 printf("\n\nSaturated humidity = %f kg H2O/kg air",
        Hs)
14
15 %_humidity = 100*(H/Hs);
16 printf("\n\npercentage humidity = %f percent",
        %_humidity)
17
18 relative_humidity = 100*(pA/p_AS);
19 printf("\n\npercentage relative humidity = %f
        percent",relative_humidity)

```

Scilab code Exa 6.19 Drying operation

```

1  clc
2  clear
3  printf("example 6.17 page number 264\n\n")
4
5  //to find the air flow rate and outlet humidity
6  S=425.6    //in kg/h
7  X1 = 0.035 //in kgwater/kg dry solid
8  t_s1=25    //in degree C
9  X2 = 0.017 //in kg H2O/kg dry air
10 t_s2=60    //in degree C

```



```

11 H2 = 0.0175 //in kg H2O/kg dry air
12 t_G2 = 84.2 //in degree C
13 t_G1= 32.8 //in degree C
14 C_pS = 1.465 //in kJ/kg dry solid
15 C_pA = 4.187 //in kg/ kg H2O K
16
17 H_G2=(1.005+1.88*H2)*(t_G2-0)+H2*2501;
18 H_S1 = C_pS*(t_s1-0)+X1*C_pA*(t_s1-0); //in kJ/kg
19 H_S2 = C_pS*(t_s2-0)+X2*C_pA*(t_s2-0); //in kJ/kg
20 Q=9300; //in kJ/h
21
22 printf("Latent heat of water at 0C, HG2 = %f kJ/kg
dryair",H_G2)
23 printf("\n\nEnthalpy of entering solid , HS1 = %f kJ/
kg dryair",H_S1)
24 printf("\n\nEnthalpy of exit solid , HS2 = %f kJ/kg
dryair",H_S2)
25
26 //applying GHg2 + SHs1 = GHg1 +SHs2 +Q, we get two
linear equations
27 //0.0175G+14.17248 = GH1 and 98.194G-29745.398 =
2562.664GH1
28 A = [0.0175 -1;98.194 -2562.664];
29 b = [-14.17248;29745.398];
30 x = A\b;
31 G = x(1);
32 H1 = x(2)/G;
33 printf("\n\nAir flow rate , G = %f kg dryair/hr",G)
34 printf("\n\nHumidity , H1 = %f kg dryair/hr",H1)

```

Scilab code Exa 6.20 Crystallization

```

1 clc
2 clear
3 printf("example 6.20 page number 291\n\n")

```

```

4
5 //to find the crystal yield
6
7 M_Na2CO3 = 106
8 M_10H2O = 180.2
9 M_Na2CO3_10H2O = 286.2;
10 w_Na2CO3 = 5000 //in kg
11 %_water = 0.05 //% of water evaporated
12
13 W = %_water*w_Na2CO3;
14 //solving material balance, we have two equations
15 //equation 1 -> 0.8230L +0.6296C = 3500
16 //equation 2 -> 0.1769L + 0.3703C = 1250
17
18 A=[0.8230 0.6296;0.1769 0.3703]
19 b = [3500;1250]
20 x = A\b;
21 L = x(1);
22 C = x(2);
23 printf("L = %f kg solution",L)
24 printf("\n\nC = %f kg of Na2CO3.10H2O crystals",C)

```

Scilab code Exa 6.21 Crystallization

```

1 clc
2 clear
3 printf("example 6.21 page number 291\n\n")
4
5 //to find the crystal yield
6 //from material balance, we have two linear
   equations
7 //1400 = 0.7380L+0.5117C and 600 = 0.2619L+0.4882C
8
9 A=[0.7380 0.5117;0.2619 0.4882]
10 b = [1400;600]

```

```

11 x = A\b;
12 L = x(1);
13 C = x(2);
14 printf("L = %f kg solution",L)
15 printf("\n\nC = %f kg of MgSO4.7H2O crystals",C)
16
17 F = 2000 //in kg/h
18 cv = 2.93 //in kJ/kg K
19 H1 = F*cv*(330-293);
20 printf("\n\nenthalpy of feed = %f kJ",H1)
21
22 wt = 246.49 //molar mass MgSO4.7H2O
23 heat_soln = -13.31*10^3; //in kJ/kg mol
24 heat = heat_soln/wt;
25 heat_crystallization = abs(heat);
26 H2 = heat_crystallization*C; //total heat
27 q = -H1-H2;
28 printf("\n\nheat absorbed = %f kJ\nthus heat shall
    be removed",q)

```

Chapter 7

Chemical Kinetics

Scilab code Exa 7.1 Constant volume reaction

```
1 clc
2 clear
3 printf("example 7.1 page number 305\n\n")
4
5 printf("it is a theoretical problem, book shall be
    referred for solution")
```

Scilab code Exa 7.2 Rate of reaction

```
1 clc
2 clear
3 printf("example 7.2 page number 306\n\n")
4 printf("it is a theoretical problem, book shall be
    referred for solution")
```

Scilab code Exa 7.3 Rate of reaction

```

1  clc
2  clear
3  printf("example 7.3 page number 305\n\n")
4
5  //to find the change on rate of reaction
6  //part 1
7  //rate equation  $r = kC_{NO}^2 \cdot C_{O2}$ 
8  //if pressure increases 3 times
9
10 r = 3^2*3;    //according to the rate reaction
11 printf("reaction reate will be increased by with 3
        times increase in pressure = %f times",r)
12
13 //part 2
14 r = 3^2*3;    //according to the rate reaction
15 printf("\n\nreaction reate will be increased by with
        3 times decrease in volume = %f times",r)
16
17 r = 3^2;    //according to the rate reaction
18 printf("\n\nreaction reate will be increased by with
        3 times increase in conc of NO = %f times",r)

```

Scilab code Exa 7.4 Order of reaction

```

1  clc
2  clear
3  printf("example 7.4 page number 308\n\n")
4
5  printf("it is a theoritical problem , book shall be
        referred for solution")

```

Scilab code Exa 7.5 Rate Expression

```
1 clc
2 clear
3 printf("example 7.5 page number 308\n\n")
4
5 printf("it is a theoretical problem, book shall be
  referred for solution")
```

Scilab code Exa 7.6 Volume function

```
1 clc
2 clear
3 printf("example 7.6 page number 308\n\n")
4
5 printf("it is a theoretical problem, book shall be
  referred for solution")
```

Scilab code Exa 7.7 Pressure time relation

```
1 clc
2 clear
3 printf("example 7.7 page number 309\n\n")
4
5 printf("it is a theoretical problem, book shall be
  referred for solution")
```

Scilab code Exa 7.8 Entropy changes

```
1 clc
2 clear
3 printf("example 7.8 page number 312\n\n")
```

```
4
5 printf("it is a theoretical problem , book shall be
    referred for solution")
```

Scilab code Exa 7.9 Hydrocarbon cracking

```
1 clc
2 clear
3 printf("example 7.9 page number 312\n\n")
4
5 printf("this is a theoretical question , book shall
    be referred for solution")
```

Scilab code Exa 7.10 Equilibrium conversion

```
1 clc
2 clear
3 printf("example 7.10 page number 316\n\n")
4
5 //to find the % transformation
6 moles_A = 3;
7 moles_B = 5;
8 K = 1;
9
10 function [f] = F(x)
11     f = 15-8*x;
12 endfunction
13
14 //initial guess
15 x = 10;
16 y = fsolve(x,F);
17 printf("amount of A transformed = %f percent",y
    *(100/3))
```

Scilab code Exa 7.11 Equilibrium conversion

```
1 clc
2 clear
3 printf("example 7.11 page number 316\n\n")
4
5 //to find the product concentration
6 printf("this is a theoretical question , book shall
   be referred for solution")
```

Scilab code Exa 7.12 Concentration calculation

```
1 clc
2 clear
3 printf("example 7.11 page number 316\n\n")
4
5 //to find the initial conc of A and B
6 Cp = 0.02;
7 Cq = 0.02;
8 K = 4*10^-2;
9 Cb = 0.05;
10 Cb_i = Cb+Cp;
11 a = (Cp*Cq)/(K*Cb);
12 funcprot(0)
13 function [f] = F(x,a)
14     f = x-0.02-a;
15 endfunction
16
17 //initial guess
18 x = 10;
19 y = fsolve(x,F);
```



```
20 printf("conc of A= %f mol/l",y)
21 printf("\n\nconc of B= %f mol/l",Cb_i)
```

Scilab code Exa 7.13 Equilibrium conversion

```
1  clc
2  clear
3  printf("example 7.11 page number 316\n\n")
4
5  //to find the % transformation
6
7  moles_A = 0.02;
8  K = 1;
9
10 //part 1
11 moles_B = 0.02;
12 function[f] = F(x)
13     f = moles_A*moles_B-(moles_A+moles_B)*x;
14 endfunction
15
16 //initial guess
17 x = 10;
18 y = fsolve(x,F);
19 printf("amount of A transformed = %f percent",y
        *(100/0.02))
20
21 //part 2
22 moles_B = 0.1;
23 y = fsolve(x,F);
24 printf("\n\namount of A transformed = %f percent",y
        *(100/0.02))
25
26 //part 1
27 moles_B = 0.2;
28 y = fsolve(x,F);
```

```
29 printf("\n\namount of A transformed = %f percent ',y
    *(100/0.02))
```

Scilab code Exa 7.14 Equilibrium shifts

```
1 clc
2 clear
3 printf("example 7.14 page number 317\n\n")
4
5 //to find the initial concentration and shift in
  equilibrium
6
7 Ce_N2 = 3;    //equilibrium conc of N2
8 Ce_H2 = 9;    //equilibrium conc of H2
9 Ce_NH3 = 4;   //equilibrium conc oh NH3
10 C_N2 = Ce_N2 + 0.5*Ce_NH3;
11 C_H2 = Ce_H2 + 1.5*Ce_NH3;
12
13 printf("concentration of N2 = %f mol/l\n
    nconcentration of H2 = %f mol/l",C_N2,C_H2)
14 printf("\n\nsecond part is theoretical, book shall
    be referred for solution")
15
16 n_H2 = 3;    //stotiometric coefficient
17 n_N2 = 1;    //stotiometric coefficient
18 n_NH3= 2;    //stotiometric coefficient
19 delta_n = n_H2+n_N2-n_NH3;
20 if delta_n > 0 then printf ("\n\ndelta_n =%f\nsince
    delta_n is greater than 0,equilibrium will shift
    to right with increase in volume",delta_n)
21 end
```

Scilab code Exa 7.15 Rate equation

```

1  clc
2  //example 7.15
3  //to find the rate equation
4  t = [0;5;10;15;20;25]
5  C_A = [25;18.2;13.2;9.6;7;5.1]
6
7  //integral method of rate determination
8  s = 0;
9  for i = 2:6
10     k(i) = (1/t(i))*log(25/C_A(i))
11     //disp (k(i),"k values for various conc.")
12     s = s+k(i)
13 end
14
15 printf("average value of k = %f",s/5)
16 disp ("ra = 0.06367*CA", "since its a first order
        reaction ,")
17
18 subplot(221)
19 plot(t,C_A)
20 xlabel("time")
21 ylabel("concentration")
22 title("integral method")
23
24 //differential method of rate determination
25 ra = [-1.16;-0.83;-0.60;-0.43];
26 C_A = [18.2;13.2;9.6;7];
27
28 subplot(222)
29 plot(ra,C_A)
30 xlabel("Concentration")
31 ylabel("-ra")
32 title("differential method")
33
34 printf("\n\nrate from differential method = -0.064*
        CA")

```

Scilab code Exa 7.16 Rate of reaction

```
1  clc
2  clear
3  //example 7.16
4  //to find the rate of reaction
5  E = 75200    //in J/mol
6  E1 = 50100   //in J/mol
7  R = 8.314    //in J/mol K
8  T = 298     //in K
9  ratio = exp((E1-E)/(R*T));
10 rate_increase = ratio^-1
11 disp ("times",rate_increase,"increase in rate of
      reaction =")
```

Chapter 8

Measuring Devices

Scilab code Exa 8.1 Specific gravity

```
1 clc
2 clear
3 printf("example 8.1 page number 334\n\n")
4
5 printf("this is a theoretical problem, book shall be
        referred for solution")
```

Scilab code Exa 8.2 Specific gravity

```
1 clc
2 clear
3 printf("example 8.2 page number 335\n\n")
4
5 printf("this is a theoretical problem, book shall be
        referred for solution")
```

Scilab code Exa 8.3 Specific gravity

```
1 clc
2 clear
3 printf("example 8.3 page number 335\n\n")
4
5 printf("this is a theoretical problem, book shall be
        referred for solution")
```

Scilab code Exa 8.4 Mixture density

```
1 clc
2 clear
3 printf("example 8.4 page number 336\n\n")
4
5 //Chapter 8 : Measuring Devices
6 //Given: Balance Height=4m
7 //side 1-air, side 2:- N2-H2 mixture
8
9 pressure_difference = 3.4 //in mm water
10 pressure = 1.0133*10^5 //in pa
11 temperatue = 293 //in K
12 mass_of_air = 29 //in Kg
13 density_air = pressure/(temperatue*8314)*mass_of_air
//in kg/m3
14 printf("Density of air = %f kg/cu m",density_air)
15
16 delta_p = pressure_difference*9.8 //in
pascal, acceleration due to gravity, g=9.8
17 Height=4
18 density_difference = delta_p/(9.8*Height);
19 printf("\n\nDensity difference = %f kg/cu m",
density_difference)
20
21 density_mixture= density_air-density_difference;
```

```

                //in kg/m3
22 printf("\n\nDensity of mixture = %f kg/cu m",
        density_mixture)

```

Scilab code Exa 8.5 Viscosity calculation

```

1  clc
2  clear
3  printf("example 8.5 page number 341\n\n")
4
5  //to find viscosity of oil
6  diameter=0.6;           //in m
7  disk_distance=1.25*10^-3; //in m
8  speed=5;               //revolutions/min
9  torque=11.5;           //in Joules
10
11 //we know that torque= pi*omega*viscosity*radius
    ^4/2*disc_distance
12 viscosity=(2*disk_distance*torque)/(3.14*(10*3.14)*(
    diameter/2)^4);
13 printf("viscosity = %f Pa-s",viscosity)

```

Scilab code Exa 8.6 Solution viscosity

```

1  clc
2  clear
3  printf("example 8.6 page number 342\n\n")
4
5  //to find the viscosity of solution using given
    parameters
6
7  diameter =10;           //in mm
8  density_of_solution = 1750; //in kg/m3

```

```

9 density_of_air = 1.2;           //in kg/m3
10 velocity = 0.9;               //in mm/s
11 viscosity = (density_of_solution-density_of_air)
    *9.8*(diameter*10^-3)^2/(18*velocity*10^-3);
    //expression for finding viscosity
12
13 printf("viscosity of solution = %f Pa-s",viscosity)
14
15
16 //checking stoke's region validity
17 v=(0.2*viscosity)/(density_of_solution*diameter
    *10^-3);
18 if v>0.9 then printf("\n\nsystem follows stokes law"
    )
19 end

```

Scilab code Exa 8.7 Flow rate calculation

```

1 clc
2 clear
3 printf("example 8.7 page number 367\n\n")
4
5 //to find the flow rate in an orifice
6 density_of_water = 1000;       //in kg/m3
7 viscosity = 1*10^-3;          //in Pa-s
8 pipe_diameter = 250;           //in mm
9 orifice_diameter = 50;         // in mm
10 density_of_mercury = 13600;   // in mm
11 manometer_height = 242;       //in mm
12
13 //calculation
14 height_water_equivalent = (density_of_mercury-
    density_of_water)*(manometer_height*10^-3)/(
    density_of_water)           //in m
15

```



```

16 //assuming Re>30000
17 Co = 0.61;
18 velocity = Co*(2*9.8*height_water_equivalent/(1-(
    orifice_diameter/pipe_diameter)^4))^0.5; //in
    m/s
19
20 //checking Reynold's number
21 Re = (orifice_diameter*10^-3*velocity*
    density_of_water)/viscosity;
22 printf("reynolds number = %f\nwhich is greater than
    30000",Re)
23
24 if Re>30000 then printf("\n\nvelocity of water = %f
    m/s",velocity)
25
26 end
27
28 rate_of_flow = (3.14*(orifice_diameter*10^-3)^2/4)*
    velocity*density_of_water;
29 printf("\n\nrate of flow = %f litre/s",rate_of_flow)

```

Scilab code Exa 8.8 Venturi meter

```

1 clc
2 clear
3 printf("example 8.8 page number 368\n\n")
4
5 //to find the coefficient of discharge for
    converging cone
6
7 pipe_diameter=0.15; //in m
8 venturi_diameter=0.05; //in m
9 pressure_drop=0.12; //m of water
10 flow_rate=3; //in kg/s
11 density = 1000; //in kg/m3

```

```

12 viscosity = 0.001           //in Pa-s
13
14 velocity = ((4/3.14)*flow_rate)/(venturi_diameter^2*
    density);
15 printf("velociy = %f m/s",velocity)
16
17 //calculating coefficient of discharge
18 Cv=velocity*((1-(venturi_diameter/pipe_diameter)^4)
    /(2*9.8*pressure_drop))^0.5;
19 printf("\n\ncoefficient of discharge = %f",Cv)
20
21 //calculating reynold's number
22 Re = velocity*(venturi_diameter/pipe_diameter)^2*
    pipe_diameter*density/viscosity;
23 printf("\n\nreynolds No = %f",Re)

```

Scilab code Exa 8.9 Venturi meter

```

1  clc
2  clear
3  printf("example 8.9 page number 369\n\n")
4
5  //to find pA and pB
6  //part 1
7
8  h1=0.66;           //in m
9  h2=0.203;         //in m
10 h3=0.305           //in m
11 density=1000;     //in kg/m3
12 pB=68900;         //in Pa
13 s1=0.83;
14 s2=13.6;
15 disp("part 1")
16 pA=pB+(h2*s2-(h1-h3)*s1)*density*9.81; //in Pa
17 printf("\npressure at A = %f Pa\n",pA)

```

```

18
19 disp(" part 2")
20 pA1=137800           //in Pa
21 pressure=735        //mm Hg
22 pB1=pA1-(h2*s2-(h1-h3)*s1)*density*9.81;
23 pressure_B=(pB1-pressure*133.3)/9810;           //m of
           water
24 printf("\npressure at B = %f m of water",pressure_B)

```

Scilab code Exa 8.10 Pitot tube

```

1  clc
2  clear
3  printf("example 8.10 page number 370\n\n")
4
5  //to find the rate of oil flow in l/s
6
7  density_oil=900;           //in kg/m3
8  viscosity_oil=38.8*10^-3; //in Pa-s
9  density_water = 1000;     //in kg/m3
10 diameter=0.102           //in m
11 manometer_reading=0.9;    //m of water
12 delta_H=manometer_reading*(density_water-density_oil
           )/density_oil;
13 printf("manometer reading as m of oil = %f m",
           delta_H)
14
15 maximum_velocity=(2*9.8*delta_H)^0.5;
16 printf("\n\nmaximum_velocity (Vmax) = %f m/s",
           maximum_velocity)
17
18 Re=diameter*maximum_velocity*density_oil/
           viscosity_oil;
19 printf("\n\nif Re<4000 then v=0.5*Vmax Re = %f",Re)
20 if Re<4000 then velocity=maximum_velocity*0.5;

```

```

21 end
22
23 printf("\n\nvelocity = %f m/s",velocity)
24
25 flow_rate=(3.14/4)*diameter^2*velocity*1000;
26 printf("\n\nflow rate =%f litre/s",flow_rate)

```

Scilab code Exa 8.11 Rotameter capacity

```

1 clc
2 clear
3 printf("example 8.11 page number 372\n\n")
4
5 //to find the maximum capacity of keroscene
6 flow_rate_steel=1.2; //l/s
7 density_steel=7.92;
8 density_kerosene=0.82;
9 density_water=1;
10 flow_rate_kerosene =(((density_steel -
    density_kerosene)/density_kerosene)/((
    density_steel-density_water)/density_water))^0.5*
    flow_rate_steel
11 printf("maximum_flow rate of kerosene = %f litre/s",
    flow_rate_kerosene)

```

Scilab code Exa 8.12 Flow rate calculation

```

1 clc
2 clear
3 printf("example 8.12 page number 373\n\n")
4
5 //to find the rate of flow of flue gas
6

```

```

7 initial_CO2 = 0.02;           //weight fraction
8 flow_rate_CO2 = 22.5;        //gm/s
9 final_CO2=0.031;           //weight fraction
10
11 //flow rate of flue gas =x
12 //amount of CO2 entering = 0.02*x
13 //amount of CO2 leaving = 0.02x+0.0225
14 //amount of gas leaving = x+0.0225
15 //amount of CO2 leaving = 0.031*(x+0.0225)
16
17 deff('y=f(x)', 'y=initial_CO2*x+0.0225 - 0.031*(x
      +0.0225)');
18
19 flow_rate_flue_gas=fsolve(0,f)
20
21 printf("flow rate of flue gas = %f kg/s",
      flow_rate_flue_gas)

```

Chapter 9

Computers and their application

Scilab code Exa 9.1 Coiled tube pressure drop

```
1 clc
2 clear
3 printf("example 9.1 page number 384\n\n")
4
5 //to find the pressure drop in the coil
6
7 D = 38*10^-3; //in m
8 U = 1 //in m/s
9 density = 998 //in kg/cubic m
10 viscosity = 8*10^-4 //in Pa-s
11 DC = 1 //in m
12 N = 10
13 e = 4*10^-6; //in m
14
15 Re = (density*U*D)/viscosity;
16 printf ("Reynolds number = %f",Re)
17
18 f = (4*log10((e/D)/3.7+(6.81/Re)^0.9))^-2;
19 printf("\n\nfriction factor = %f",f);
```

```

20
21 L = 3.14*DC*N;
22
23 delta_Pstr = (2*f*U*density*L)/D;
24 printf("\n\npressure drop through straight pipe = %f
      Pa",delta_Pstr)
25
26 S = 1+3.54*(D/DC);
27 printf("\n\ncorrection factor = %f",S)
28
29 delta_P = S*delta_Pstr
30 printf("\n\npressure drop of coil = %f Pa",delta_P)

```

Scilab code Exa 9.2 Heat exchanger pressure drop

```

1  clc
2  clear
3  printf("example 9.2 page number 384\n\n")
4
5  //to find the shell side pressure drop in heat
   exchanger
6
7  U = 0.5    //in m/s
8  N = 19;
9  DT = 0.026 //in m
10 L = 2.7    //in m
11 DS = 0.2   //in m
12 e = 0.0002 //in m
13 density = 836 //in kg/cu m
14 viscosity = 0.00032 //in Pa s
15 Pr = 6.5;
16 Prw = 7.6;
17
18
19 HYDIA = (DS^2-N*DT^2)/(DS+N*DT);

```

```

20
21 Re = HYDIA*U*density/viscosity;
22 printf ("Reynolds number = %f",Re)
23
24 f = (4*log10((e/HYDIA)/3.7+(6.81/Re)^0.9))^-2;
25 printf("\n\nfriction factor = %f",f);
26
27 L = 3.14*DT*N;
28
29 delta_Pstr = (2*f*U*density*L)/HYDIA;
30 printf("\n\npressure drop through straight pipe = %f
    Pa",delta_Pstr)
31
32 S = (Prw/Pr)^0.33;
33 printf("\n\n correction factor = %f",S)
34
35 delta_P = S*delta_Pstr
36 printf("\n\n pressure drop of coil = %f Pa",delta_P)

```

Scilab code Exa 9.3 Heat exchanger area

```

1 clc
2 clear
3 printf("example 9.3 page number 385\n\n")
4
5 MH = 10 //in kg/s
6 MC = 12.5 //in kg/s
7 CPH = 4.2 //in kJ/kg
8 CPC = 4.2 //in kJ/kg
9 THI = 353 //in K
10 THO = 333 //in K
11 TCI = 300 //in K
12 U = 1.8 //in kW/sq m K
13
14 Q = MH*CPH*(THI-THO);

```



```

15 printf("heat load = %f J",Q)
16
17 TCO = Q/(MC*CPC)+TCI;
18 printf("\n\ncold fluid outlet temperature = %f K",
        TCO)
19
20 //for co current flow
21
22 DT1 = THI-TCO;
23 DT2 = THO-TCO;
24
25 LMTD = (DT1-DT2)/log(DT1/DT2);
26
27 A = Q/(U*LMTD);
28 printf("\n\nfor co current flow , area = %f sq m",A);
29
30 //for counter current flow
31
32 DT1 = THI-TCO;
33 DT2 = THO-TCI;
34
35 LMTD = (DT1-DT2)/log(DT1/DT2);
36
37 A = Q/(U*LMTD);
38 printf("\n\nfor counter current flow , area = %f sq m
        ",A);

```

Scilab code Exa 9.4 Batch distillation

```

1 clc
2 clear
3 printf("example 9.4 page number 387\n\n")
4
5 printf("this is a theoretical question , book shall
        be referred for solution")

```

Scilab code Exa 9.5 Gas mixture exit temperature

```
1 clc
2 clear
3 printf("example 9.5 page number 392\n\n")
4
5 printf("this is a theoretical question , book shall
    be referred for solution")
```

Scilab code Exa 9.6 Friction factor calculation

```
1 clc
2 clear
3 printf("example 9.6 page number 395\n\n")
4
5 printf("this is a theoretical problem , book shall be
    referred for solution")
```
