

Scilab Textbook Companion for
Chemical Engineering - Fluid Flow, Heat
Transfer And Mass Transfer - Vol. 1
by J. M. Coulson, J. F. Richardson, J. R.
Backhurst And J. H. Harker¹

Created by
Harshit Amya
B.Tech
Chemical Engineering
IT-BHU, VARANASI
College Teacher
R.S.singh
Cross-Checked by

August 10, 2013

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

Book Description

Title: Chemical Engineering - Fluid Flow, Heat Transfer And Mass Transfer
- Vol. 1

Author: J. M. Coulson, J. F. Richardson, J. R. Backhurst And J. H. Harker

Publisher: Elsevier India

Edition: 6

Year: 2006

ISBN: 9788181473868

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 Units and Dimensions	9
2 Flow of Fluids Energy and Momentum Relationships	11
3 Flow of liquids in Pipes and Open Channels	16
4 Flow of Compressible Fluids	28
5 Flow of Multiphase Mixtures	33
6 Flow and Pressure Measurement	38
7 Liquid Mixing	44
8 Pumping of Fluids	47
9 Heat Transfer	59
10 Mass Transfer	99
11 The Boundary Layer	111
12 Momentum Heat and Mass Transfer	114
13 Humidification and Water Cooling	120

List of Scilab Codes

Exa 1.1	Conversion of poise into British Engineering and SI units	9
Exa 1.2	Conversion of kW to hp	9
Exa 2.1	Estimation of volume of vessel by 3 methods	11
Exa 2.3	Calculation of reaction force	12
Exa 2.4	Calculation of resultant force	13
Exa 2.5	Calculation of jet velocity neglecting frictional effects .	13
Exa 2.6	Calculation of pressure at the wall	14
Exa 3.1	Pressure drop calculation in pipeline	16
Exa 3.2	Maximum allowable water velocity calculation	17
Exa 3.3	Calculation of time for drop in water level	18
Exa 3.4	Calculation of volumetric flow rate	19
Exa 3.5	Calculation of radius for given condition	20
Exa 3.6	Calculation of loss in head due to sudden enlargement of pipe	21
Exa 3.7	Calculation of power supplied to the pump	22
Exa 3.8	Calculation of initial rate of discharge of water	23
Exa 3.9	Calculation of velocity and liquid depth	24
Exa 3.10	Calculation of yeild stress plastic viscosity pressure drop and centre line velocity	24
Exa 3.11	Calculation of velocity and ratio of volumetric flow rates	26
Exa 4.1	Plot the rate of discharge of air vs cylinder pressure and vs Downstream Pressure	28
Exa 4.2	Calculation of approximate flow rate of fluid	29
Exa 4.3	Calculation of Pressure to be developed at the compres- sor in order to achieve a given flowrate	30
Exa 4.4	Calculation of rate of discharge of gas upstream and downstream pressure and mach number	31

Exa 5.1	Calculation of Pressure drop per unit length of pipe under adiabatic conditions	33
Exa 5.2	Calculation of maximum mass flow of sand	34
Exa 5.3	Calculation of Pressure drop	36
Exa 6.1	Calculation of difference in level on a water manometer	38
Exa 6.2	Calculation of mass flow rate and drop in pressure	39
Exa 6.3	Calculation of Coefficient for the converging cone of the meter at given flowrate	40
Exa 6.4	Calculation of flow rate of water	41
Exa 6.5	Calculation of volumetric flow rate of water	42
Exa 6.6	Calculation of Height of liquid flowing over the weir	43
Exa 7.2	Calculation of the power provided by the propeller to the liquid	44
Exa 7.3	Determining new power consumption reynolds number and rotor speed	45
Exa 8.1	Calculation of the maximum speed at which the pump can run	47
Exa 8.2	Calculation of the minimum height required between the liquid level in the reboiler and the pump	48
Exa 8.3	Calculation of the theoretical power requirements for the compression	50
Exa 8.4	Calculation of the work of compression isothermal and isentropic efficiency and the raio of swept volumes in two cylinders	51
Exa 8.5	Calculation of the power requirement of the pump	53
Exa 8.6	Calculation of the efficiency of the pump and the mean velocity of the mixture in the pipe	54
Exa 8.7	Calculation of power to be supplied to the pump	55
Exa 8.8	Estimation of the rate of flow and the power to be supplied to the pump	56
Exa 8.10	Calculation of flow rate and power required by the pump	57
Exa 9.1	Calculation of surface area for counter and cocurrent flow in concentric heat exchanger	59
Exa 9.2	Calculation of the heat loss per square meter of the surface	60
Exa 9.3	Estimation of Heat loss and temperature at firebrick insulating brick interface	60

Exa 9.4	Calculation of the time taken for distant face of brick wall to rise from 295 to 375 K	61
Exa 9.5	Calculation of the time taken for distant face of brick wall to rise from 295 to 375 K using Schmidts method	62
Exa 9.6	Calculation of final temperature of a sphere and a cube	63
Exa 9.7	Calculation of minimum time required for heating a sheet	64
Exa 9.8	Calculation of the temperature difference between the surface and the centre of the uranium element	65
Exa 9.9	Calculation of value of scale resistance	65
Exa 9.10	Calculation of pressure drop over tube bundle	67
Exa 9.11	Calculation of heat transfer coefficient	68
Exa 9.12	Calculation of temperature of a surface coated with carbon black	68
Exa 9.13	Calculation of number of heating elements	69
Exa 9.14	Calculation of emissivity of a grey surface	70
Exa 9.15	Calculation of the view factor and net radiation transfer	71
Exa 9.16	Calculation of view factor for 3 dimensional geometry using figures	71
Exa 9.17	Calculation of view factor of two parallel rings	72
Exa 9.18	Calculation of net rate of heat transfer by radiation to the plate	73
Exa 9.19	Calculation of radiant heat transfer to the vessel	74
Exa 9.20	Calculation of radiosity net rate of heat transfer and coefficient of heat transfer	76
Exa 9.21	Estimation of the electrical input to the heater and the net rate of heat transfer to the plate	77
Exa 9.22	Calculation of net radiation to the walls	79
Exa 9.23	Estimation of the overall coefficient of heat transfer and the dirt factor for the condenser	81
Exa 9.24	Calculation of overall heat transfer coefficient	84
Exa 9.25	To calculate the time required for heating the liquid	86
Exa 9.26	Calculation of the surface area required to effect the given duty using a multipass heat exchanger	87
Exa 9.27	Estimation of the heat transfer area required for the system	87
Exa 9.28	Using Kerns method to design Shell n tube heat exchanger	88

Exa 9.29	Estimation of Effectiveness of the given double pipe heat exchanger	92
Exa 9.30	Estimate heat transfer surfaces of One shell pass two tube pass heat exchanger	93
Exa 9.31	Calculation of gas temperature flowing through a copper pipe	94
Exa 9.32	Determination of the heat loss per metre run of the tube	95
Exa 9.33	Calculation of heat loss to the air	96
Exa 9.34	Determination of the economic thickness of lagging . .	97
Exa 10.1	Estimation of the rate of diffusion of ammonia through the layer	99
Exa 10.2	Calculation of the diffusivity of carbon tetrachloride vapour in air	100
Exa 10.3	Calculation of the mass transfer rate per unit area . .	101
Exa 10.4	Calculation of the maximum length of column to which penetration theory can be applied	102
Exa 10.5	Calculation of equivalent resistance and time required	103
Exa 10.6	Calculating proportion of absorbed carbon dioxide . .	104
Exa 10.8	Calculation of overall mass transfer coefficient based on gas phase	105
Exa 10.9	Calculation of numerical value of a given ratio	105
Exa 10.10	Determination of order of chemical reaction	106
Exa 10.11	Calculate By what factor will the mass transfer rate across the interface change	107
Exa 10.12	Estimation of the Thiele modulus and the effectiveness factor for a reactor	108
Exa 10.13	Calculation of the effectiveness factor and the concentration of reactant at a given position	109
Exa 10.14	Calculation of the effectiveness factor	110
Exa 11.1	Calculation of total drag force acting on a surface . . .	111
Exa 11.2	Calculation of thickness and displacement thickness of the boundary layer	111
Exa 11.3	Calculation of the thickness of the laminar sublayer and velocity of the benzene	112
Exa 12.1	Calculation of thickness of laminar sub layer	114
Exa 12.2	Estimation of the air temperature at a given point along the pipe	115
Exa 12.3	Estimation of outlet temperature by different methods	116

Exa 12.4	Estimation of outlet temperature by different methods	118
Exa 13.1	Pressure calculation	120
Exa 13.2	Calculation of partial pressure specific volumes humidity humid volume and percentage humidity	121
Exa 13.3	Estimation of the humidity of the air and the percentage relative humidity	122
Exa 13.4	Determination of the temperature of the material on each tray the amount of water removed and the temperature to which the inlet air would have to be raised	123
Exa 13.5	Calculation of enthalpy humidity and temperature of resultant stream	125
Exa 13.6	Calculation of temperature enthalpy and relative humidity of mixed stream	126

Chapter 1

Units and Dimensions

Scilab code Exa 1.1 Conversion of poise into British Engineering and SI units

```
1  clc;
2
3  printf("Example 1.1\n");
4  // 1 Poise = 1g/cm s = ((1/453.6) lb) / ((1/30.48) ft * 1s
   )
5  be=30.48/453.6*3600; //be->british engineering unit
6  printf("\n 1 Poise = %.4f lb/ft s",be/3600);
7  printf("\n          = %.0f lb/ft h",be);
8
9  // 1 Poise = 1g/cm s = ((1/1000)kg) / ((1/100)m*1s)
10 si=100/1000; //si->SI units
11 printf("\n 1 Poise = %.1f kg/m s ",si);
12 printf("\n          = %.1f N s/m^2 ",si);
```

Scilab code Exa 1.2 Conversion of kW to hp

```
1  clc;
```

```

2
3 printf("Example 1.2\n");
4 // 1 kW= 103 W = 103 J/s = 10^3 * (1 kg*1 m^2)/1 s^3
5 //                               = (10^3 * (1/0.4536) lb x
6   (1/0.3048)^2 ft^2)/1 s^3
7 lfs=(10^3*(1/0.4536)*(1/.3048)^2); //lfs->lb ft^2/s
   ^3
8 printf("\n 1 kW = %.0f lb ft^2/s^3",lfs);
9 sfs=lfs/32.2; //sfs->slug ft^2/s^3
10 printf("\n 1 kW = %.0f slug ft^2/s^3",sfs);
11 printf("\n 1 kW = %.0f lbf ft/s",sfs);
12 hp=sfs/550;
13 printf("\n 1 kW = %.2f h.p.",hp);
14 printf("\n 1 h.p.= %.3f kW",1/hp)

```

Chapter 2

Flow of Fluids Energy and Momentum Relationships

Scilab code Exa 2.1 Estimation of volume of vessel by 3 methods

```
1  clc;
2
3  printf("Example 2.1\n");
4  //For 1 kmol of methane
5
6  //(a) PV = 1 * RT, where
7  R=8314;
8  P=60*10^6;
9  T=320;
10 Tc=191;
11 Pc=4.64*10^6;
12 printf("\n Given\n R=8314 J/kmol K.\n P=60*10^6 N/m
        ^2\n T=320 K;")
13 printf("\n Tc=191 K \n Pc=4.64*10^6 N/m^2")
14 V1=8314*T/P;
15 printf("\n(a)\n Volume of vessel (ideal gas law) = %
        .4 f m^3",V1);
16
17
```

```

18 //(b) In van der Waals equation (2.32), the
    constants may be taken as:
19 a=27*R^2*Tc^2/(64*Pc);
20 b=R*Tc/(8*Pc);
21 printf("\n(b)\n a = %d (N/m^2)*(m^3)^2/(kmol)^2",a);
22 printf("\n b = %.4f m^3/kmol",b);
23 //Thus using equation 2.32:
24 x=poly([0], 'x');
25 p=roots((60*10^6*x^2+a)*(x-0.0427)-(8314*320*x^2));
26 printf("\n Volume of vessel(van der waals eq.) = %
    .3f m^3",p(1,1));
27
28
29 //(c) Tr=T/Tc ,Pr=P/Pc
30 Tr=T/Tc;
31 printf("\n(c)\n Tr = %.2f",Tr);
32 Pr=P/Pc;
33 printf("\n Pr = %.2f",Pr);
34 //Thus from Figure 2.1,
35 Z=1.33;
36 //V = ZnRT/P (from equation 2.31)
37 V3=Z*R*T/P;
38 printf("\n Volume of vessel(generalised
    compressibility-factor chart) = %.4f m^3",V3);

```

Scilab code Exa 2.3 Calculation of reaction force

```

1 clc;
2
3 printf("Example 2.3\n");
4 //Mass rate of discharge of water, G = rho*u*A
5 rho=1000; //Density of Water
6 d=25*10^-3; //Diameter of nozzle
7 u=25; //Velocity of water at nozzle
8 printf("\n Given:\n Density of water = %d kg/m^3\n

```

```

        Nozzle diameter = %.3f m \n Velocity = %d m/s",
        rho,d,u);
9 G=rho*u*pi/4*d^2;
10 printf("\n\n Mass rate of discharge of water, G = %
        .2f kg/s",G);
11 //Momentum of fluid per second = Gu
12 F=G*25;
13 printf("\n Momentum of fluid per second = %.0f N",F)
        ;
14 printf("\n Reaction force = Rate of change of
        momentum = %.0f N",F);

```

Scilab code Exa 2.4 Calculation of resultant force

```

1 clc;
2
3 printf("Example 2.4\n");
4 //Momentum per second of approaching liquid in Y-
        direction = rho*u^2*A
5 rho=1000; //Density of water
6 d=50*10^-3; //Diameter of pipe
7 u=5; //Velocity of water in pipe
8 printf("\n Given\n Density of water = %d kg/m^3\n
        Pipe diameter = %.3f m \n Velocity = %d m/s",rho,
        d,u);
9 M=rho*u^2*pi/4*d^2;
10 printf("\n\n Momentum per second of approaching
        liquid in Y-direction = %.1f N",M);
11 Rf=M*(cos(pi/4)+sin(pi/4));
12 printf("\n The resultant force in direction of arm
        of bracket = %.1f N",Rf);

```

Scilab code Exa 2.5 Calculation of jet velocity neglecting frictional effects

```

1  clc;
2
3  printf("Example 2.5\n");
4  //From equation 2.68:
5  // 0.5*((u2)^2-(u1)^2)=g*(z1-z2)+((P1-P2)/rho)
6  //Suffix 1 to denote conditions in the pipe and
   suffix 2 to denote conditions in the jet
7  //Symbols have their usual meaning
8  u1=0;
9  z1=0;
10 z2=0;
11 P1=250*10^3;
12 P2=0;
13 rho=1000; //Density of water
14 printf("\n Suffix 1 to denote conditions in the pipe
   and suffix 2 to denote conditions in the jet")
15 printf("\n Given:\n u1=%d m/s\n z1= %d m\n z2= %d m\n
   n P1= %.3f kN/m^2\n P2= %d kN/m^2\n Density of
   water= %d kg/m^3", u1, z1, z2, P1, P2, rho);
16 g=9.81;
17 x=poly([0], 'x');
18 u2=roots((0.5*(x)^2)-((P1-P2)/rho));
19 printf("\n\n Ans:\n Velocity of the jet, u2 = %.1f m
   /s", u2(1,1));

```

Scilab code Exa 2.6 Calculation of pressure at the wall

```

1  clc;
2
3  printf("Example 2.6\n");
4  id=0.5; //internal diameter of pipe
5  rs=50; //revolution speed
6  ir=0.15; //internal radius of water
7  rho=1000; //density of water
8  printf("\n Given:\n Internal diameter = %.1f m\n

```

```

    Rotating speed = %d rev/s\n Inner radius of
    liquid = %.2f m\n Density of water= %d kg/m^3",id
    ,rs,ir,rho);
9  omega=2*%pi*rs;
10 printf("\n\n Angular speed of rotation = %d rad/s",
    omega);
11 //The wall pressure is given by equation 2.82 as:
12 wall_pressure=rho*(omega)^2/2*((id/2)^2-ir^2);
13 printf("\n The wall pressure is = %f N/m^2 \n\t\t\t\t=
    %.2f x 10^6 N/m^2",wall_pressure,wall_pressure
    /10^6);

```

Chapter 3

Flow of liquids in Pipes and Open Channels

Scilab code Exa 3.1 Pressure drop calculation in pipeline

```
1  clc;
2
3  printf("Example 3.1\n\n");
4  sap=1.25; //Sulphuric acid pumped
5  d=25e-3; //Diameter of pipe
6  l=30; //length of pipe
7  meu=25e-3; //Viscosity of acid
8  rho_a=1840; //Density of acid
9  printf(" Given :\n Sulphuric acid pumped = %.2f kg/s
        \n Diameter of pipe = %.3f m\n length of pipe =
        %d m\n Viscosity of acid = %d x 10^-3 N s/m^2\n
        Density of acid = %d kg/m^3",sap,d,l,meu*1000,
        rho_a);
10 Re=4*sap/(%pi*meu*d);
11 printf("\n\n\n Reynolds number , Re=(u*d*rho)/meu =
        4G/(pi*meu*d)= %d",Re);
12
13 //For a mild steel pipe, suitable for conveying the
        acid , the roughness e will be between 0.05 and
```

```

    0.5 mm (0.00005 and 0.0005 m).
14 //The relative roughness is thus: e/d = 0.002 to
    0.02
15 //From Figure 3.7: R/(rho*u^2) = 0.006 over this
    range of e/d
16 u=sap/(rho_a*%pi/4*d^2);
17 printf("\n Velocity is , u=G/(rho *A) = %.2 f m/s" ,u)
    ;
18
19 //calculating pressure drop from the energy balance
    equation and equation 3.19
20 Dp=rho_a*((0.5+4*0.006*30/0.025)*u^2+9.81*12);
21 printf("\n Pressure Drop = %.0 f N/m^2" ,Dp);
22 printf("\n Pressure drop = %.0 f kN/m^2" ,(Dp/10^3));

```

Scilab code Exa 3.2 Maximum allowable water velocity calculation

```

1  clc;
2
3  printf("Example 3.2\n\n");
4  d=50e-3; //Diameter of pipe
5  l=100; //length of pipe
6  e=0.013; //Roughness of pipe
7  DPf=50e3; //Maximum pressure drop
8  rho=1000; //density of water
9  meu=1e-3; //viscosity of water
10 printf(" Given:\n Diameter of pipe = %.3 f m\n length
    of pipe = %d m\n Roughness of pipe = %.3 f \n
    Maximum pressure drop = %d kN/m^2\n Density of
    water = %d kg/m^3\n Viscosity of water = %.1 f mN
    s/m^2" ,d,l,e,DPf/10e3,rho,meu*10e3);
11 //From Equation 3.23
12 // phi*Re^2=R*Re^2/(rho*u^2)=-(DPf)*d^3*rho/(4*l*meu
    ^2)
13 phi_re2=(DPf)*d^3*rho/(4*l*meu^2);

```

```

14 e_d=e/(d*1e3);
15 printf("\n\n phi*Re^2 = %.2f*10^7\n e/d = %.5f",
        phi_re2*1e-7,e_d);
16 //From Figure 3.8, for given phi*Re^2 = 1.56 x 10^7
    and (e/d) = 0.00026, then:
17 //Re=7.9*10^4
18 Re=7.9e4;
19 u=Re*meu/(rho*d);
20 printf("\n\n Ans \n The maximum allowable velocity
        is = %.1f m/s",u);

```

Scilab code Exa 3.3 Calculation of time for drop in water level

```

1  clc;
2
3  printf("Example 3.3\n");
4  Dia_tank=5; //Diameter of the tank
5  len_pipe=100; //Length of pipe
6  dia_pipe=225e-3; //Diameter of pipe
7  printf("\n Given:\n Diameter of the tank = %d m\n
        Length of pipe = %d m\n Diameter of pipe = %.2f m
        ",Dia_tank,len_pipe,dia_pipe);
8
9  //If at time t the liquid level is D m above the
    bottom of the tank, then
10 // designating point 1 as the liquid level and point
    2 as the pipe outlet ,
11 // and applying the energy balance equation (2.67)
    for turbulent flow, then:
12
13 // The equation becomes (u2^2/2)-D*g+(4*R*len_pipe*
    u2^2/(rho*u^2*dia_pipe))
14
15 //As the level of liquid in the tank changes from D
    to (D + dD), the quantity

```

```

16 // of fluid discharged = (pi/4)5^2(-dD) = -19.63dD m
    ^3.
17 // The time taken for the level to change by an
    amount dD is given by:
18 //dt=19.63dD/((pi/4)0.225^2 x 4.43D^0.5/([1 + 3552(/
    R/rho*u^2)])^0.5)
19
20 // calculating the value of X=R/rho*u^2 as given in
    example (refer to book)
21 X=.0020;
22 t=integrate('111.5*(1+(3552*X))^0.5*D^-0.5','D'
    ,0.3,3);
23 printf("\n\n The time taken for the level to fall is
    therefore about %d s",t);

```

Scilab code Exa 3.4 Calculation of volumetric flow rate

```

1  clc;
2
3  printf("Example 3.4\n");
4
5  d1=0.3; //diameter of pipe from junction A to D or B
    to D
6  l1=1.5e3; //length of pipe from junction A to D or B
    to D
7  d2=0.5; // diameter of pipe from junction D to C
8  l2=0.75e3; // length of pipe from junction D to C
9  h_A=10; // height of tank A above C
10 h_B=h_A+6; // height of tank A above C
11 rho=870; // density of liquid
12 Meu_1=0.7e-3; // viscosity of liquid
13
14 //It may be assumed, as a first approximation, that
    R/(rho*u^2) is the same in each pipe and that the
    velocities in pipes AD, BD, and DC are u1,u2 and

```

```

    u3
15 //respectively ,
16 //Taking the roughness of mild steel pipe e as
    0.00005 m, e/d varies from
17 //0.0001 to 0.00017. As a first approximation , R/(
    rho*u^2) may be taken as 0.002
18 //Then applying the energy balance equation between
    D and the liquid level in
19 //each of the tanks gives
20 //On forming and solving the equations
21
22 x=poly([0], 'x');
23 u2=roots(x^4-(7.38*x^2)+13.57);
24 u1=(u2^2-1.47)^0.5;
25 u3=(u1+u2)/2.78;
26 //taking the positive values and which satisfy
    equation 7
27 U1=u1(4);
28 U2=u2(4);
29 U3=u3(4);
30 Q=%pi/4*d2^2*U3;
31 printf("\n The volumetric flow rate = %.2f m^3/s",Q)
    ;

```

Scilab code Exa 3.5 Calculation of radius for given condition

```

1  clc;
2
3  printf("Example 3.5\n");
4
5  // Ux = Ucl*(y/r)^1/7      equation 3.59 (Prandtl one
    -seventh power law)
6  //where UCL is the velocity at the centre line of
    the pipe, and r is the radius of the pipe.
7  // Then total flow , Q = 49/60*pi*r^2*Ucl

```

```

equation 3.62
8
9 //When the flow in the central core is equal to the
  flow in the surrounding annulus, then taking a =
  y/r, the flow in the central core is:
10 //Qc=pi*r^2*Ucl*(105*a^(8/7)-56*a^(15/7))/60
11 //flow in the core = 0.5 (flow in the whole pipe)
12
13 r=50;
14 a=poly([0], 'a');
15 p=roots((a^8*(105-56*a)^7)-24.5^7);
16
17 printf("\n a = %.2 f",p(8));
18 y=p(8)*r;
19 printf("\n y = %.1 f mm",y)

```

Scilab code Exa 3.6 Calculation of loss in head due to sudden enlargement of pipe

```

1 clc;
2
3 printf("Example 3.6\n");
4
5 Q=7.2; //Water flow rate
6 d1=40e-3; //initial pipe diameter
7 d2=50e-3; //diameter of pipe after enlargement
8 g=9.81;
9 printf("\n Given\n Water flow rate = %.1 f m^3/h\n d1
  = %d mm\n d1 = %d mm",Q,d1*1e3,d2*1e3);
10
11 u1=(Q/3600)/(%pi/4*d1^2); //Velocity in 40 mm pipe
12 u2=(Q/3600)/(%pi/4*d2^2); //Velocity in 50 mm pipe
13 printf("\n Velocity in 40 mm pipe = %.2 f m/s\n
  Velocity in 50 mm pipe = %.2 f m/s",u1,u2);
14 // The head lost is given by equation 3.77 as:

```

```

15 hf=(u1-u2)^2/(2*g);
16 printf("\n\n Ans\n Head lost = %.1f mm of water",hf
    *1e3);

```

Scilab code Exa 3.7 Calculation of power supplied to the pump

```

1  clc;
2
3  printf("Example 3.7\n");
4  Q_h=2.27; // flow rate of water in m^3/h
5  T=320; //Temperature of water to be pumped
6  id=40e-3; //internal diameter of pipe
7  l_h=150; //length of pipe horizontally
8  l_v=10; //length of pipe vertically
9  e=0.2e-3;
10 g=9.81;
11 rho=1000;
12 printf("\n Given\n flow rate of water in m^3/h = %.2
    f m^3/h\n Temperature of water to be pumped = %d
    K\n internal diameter of pipe = %d mm\n length of
    pipe horizontally = %d m\n length of pipe
    vertically = %d m",Q_h,T,id*1e3,l_h,l_v);
13
14 rel_rough=e/id; //Relative roughness
15 printf("\n\n Relative roughness = %.3f",rel_rough);
16 meu=0.65e-3; //Viscosity at 320 K
17 Q_s=Q_h/3600; //flow rate of water in m^3/s
18 area=%pi/4*id^2; // Area for flow
19 printf("\n Area for flow = %.2f * 10^-3 m^2",area*1
    e3);
20 u=Q_s/area; //Velocity
21 printf("\n Velocity = %.2f m/s",u);
22 Re=(id*u*rho)/meu;
23 printf("\n Reynolds No. = %d",Re);
24

```

```

25 //X=R/(rho*u^2)=0.004 (from Figure 3.7)
26 X=.004;
27 equi_len=l_h+l_v+(260*id); // Equivalent length of
    pipe
28 printf("\n Equivalent length of pipe = %.1f m",
    equi_len);
29 hf=4*X*equi_len*u^2/(id*g);
30 tot_head=hf+1.5+10; // Total head to be developed
31 printf("\n Total head to be developed = %.2f m",
    tot_head);
32 mass_thr=Q_s*rho; //Mass throughput
33 printf("\n Mass throughput = %.2f kg/s",mass_thr);
34 power_reqd=(mass_thr*tot_head*g)/0.60;
35 printf("\n\n Power required = %.1f W = %.3f kW",
    power_reqd,power_reqd*1e-3);

```

Scilab code Exa 3.8 Calculation of initial rate of discharge of water

```

1  clc;
2  printf("Example 3.8\n");
3
4  d=0.15; // diameter of pipe
5  g=9.81;
6  printf("\n Given\n Diameter of pipe = %.2f",d);
7  // From equation 3.20, the head lost due to friction
    is given by:
8  // hf = 4*phi*l*u^2/(d*g)m water
9  // The total head loss is:
10 // h=(u^2/(2*g))+hf+loss in fittings
11 // From Table 3.2., the losses in the fittings are:
    From Table 3.2., the losses in the fittings are:
12 //6.6*u^2/(2*g)
13 //Taking
14 phi=.0045;
15 x=poly([0], 'x');

```



```

16 u=roots((7.6+4*phi*(105/.15))*x^2/(2*g)-10);
17 printf("\n\n Velocity = %.2f m/s",u(1));
18 rate_dis=u(1)*%pi*d^2/4;
19 printf("\n Rate of discharge = %.3f m^3/s = %d kg/s"
    ,rate_dis ,rate_dis*1e3);

```

Scilab code Exa 3.9 Calculation of velocity and liquid depth

```

1  clc;
2
3  printf("Example 3.9\n");
4
5  u1=1.5; // velocity
6  D1=75e-3; //depth
7  g=9.81;
8  printf("\n Given\n velocity before jump= %.1f m/s\n
    depth before jump= %d mm",u1 ,D1*1e3);
9  //The depth of fluid in the channel after the jump
    is given by:
10 D2=0.5*(-D1+(D1^2+(8*u1^2*D1/g)^0.5)); //equation
    3.113
11 printf("\n\n The depth of fluid in the channel after
    the jump is = %.1f mm",D2*1e3);
12 //If the channel is of uniform cross-sectional area,
    then:
13 u2=u1*D1/D2;
14 printf("\n The velocity of fluid in the channel
    after the jump is = %.2f m/s",u2);

```

Scilab code Exa 3.10 Calculation of yeild stress plastic viscosity pressure drop and centre line velocity

```

1  clc;

```

```

2
3 printf("Example 3.10\n");
4
5 k=10;
6 n=0.2;
7 //Using the power-law model (equation 3.121):
8 printf("\n Given:\n Consistency coefficient k = %d N
    . s^n/m^-2",k);
9 printf("\n Flow behaviour index = %.1f",n);
10 Ucl=1; // centre line velocity
11 printf("\n Centre line velocity = %d m/s",Ucl);
12 l=200; // length of pipe
13 printf("\n Length of pipe = %d m",l)
14 r=.02; // radius of pipe
15 printf("\n Radius of pipe = %.2f m",r);
16 dux_dy_1=10;
17 dux_dy_2=50;
18 Ry_1=k*dux_dy_1^0.2;
19 Ry_2=k*dux_dy_2^0.2;
20 //Using the Bingham-plastic model (equation 3.125):
21 A=[1 10;1 50]
22 B=[15.85;21.87]
23 C=inv(A)*B;
24 Ry=C(1);
25 Meu_p=C(2);
26 printf("\n\n Plastic viscosity (Meu_p) = %.3f N s/m
    ^2",C(2));
27 printf("\n Yeild stress (Ry) = %.2f N s/m^2",C(1));
28 // Using Equation 3.131
29 DP=2*k*l*Ucl^n*((n+1)/n)^n*r^(-n-1);
30 printf("\n Pressure drop (Bingham plastic model)= %
    .0f kN/m^2",DP/1e3);
31 // For a Bingham-plastic fluid:
32 // The centre line velocity is given by equation
    3.145:
33 X=(1*2*Ry)/(r*DP);
34 Up=(DP*r^2*(2-4*X+2*X^2))/(8*Meu_p*1);
35 printf("\n centre line velocity (Bingham plastic

```

```
model) = %.2f m/s",Up);
```

Scilab code Exa 3.11 Calculation of velocity and ratio of volumetric flow rates

```
1  clc;
2
3  printf("Example 3.11\n");
4  // given:
5  Meu=0.1; // Viscosity of liquid
6  printf("\n Given \n Viscosity of liquid = %.1f N s/m
       ^2",Meu);
7  d=25e-3; // Diameter of pipe
8  printf("\n Diameter of pipe = %.3f m",d);
9  l=20; // length of pipe
10 printf("\n length of pipe = %d m",l);
11 DP=1e5; // Pressure drop
12 printf("\n Pressure drop = %d N/m^2",DP);
13 n=1/3; // flow index of polymer solution
14 printf("\n flow index = %.2f",n);
15 dux_dy=1000;
16 k=Meu;
17 Meu_a=Meu;
18 k_poly_sol=Meu_a/(dux_dy)^(n-1);
19 Ry=10*(dux_dy)^n;
20 //From equation 3.136:
21 //For a power-law fluid:
22 u2=((DP/(4*k_poly_sol*l))^3)*(n*(d^((n+1)/n)))
    /(2*(3*n+1));
23 printf("\n\n Velocity for polymer solution = %.4f m/
    s",u2);
24 u1=(DP/(4*k*l))*(d^2)/8
25 printf("\n Velocity for original solution = %.3f m/s
    ",u1);
26 ratio=u2/u1;
```

```
27 printf("\n Ratio of the volumetric flow rates of the  
two liquids = %.3f",ratio);
```

Chapter 4

Flow of Compressible Fluids

Scilab code Exa 4.1 Plot the rate of discharge of air vs cylinder pressure and vs Downstream Pressure

```
1  clc;
2  clear;
3
4  printf("\n Example 4.1\n");
5
6  d=0.006; //Diameter of the cylinder
7  Gamma=1.4;
8  //The critical pressure ratio for discharge through
   the valve
9  C_r=(2/(Gamma+1))^(Gamma/(Gamma-1));
10 printf("\n The graphs are plotted between\n (1) Rate
   of discharge of air from the cylinder against
   cylinder pressure\n (2) For a constant pressure
   of 5 MN/m^2 in the cylinder , the discharge rate
   vs Downstream pressure.")
11 // (i) Sonic velocity will occur until the
12 P_c=101.3/C_r; //pressure at which sonic velocity
   will occur
13 M=29; //molecular mass of air
14 //The rate of discharge for cylinder pressures
```

```

    greater than 191.1 kN/m2 is
15 //given by equation 4.30: taking mean value for
    Gamma i.e. 1.47
16 //we get
17 //G_max=4.23e-8*P1 kg/s
18 //For cylinder pressures below 191.1 kN/m2, the mass
    flowrate is given by equation 4.20
19 P1a=[0.1 0.125 0.15 0.17 0.19 0.2 0.5 1.0 2.0 3.0
    4.0 5.0 6.0]
20 for i=5:13
21     G(i)=4.23e-2*P1a(i);
22 end
23 for j=1:4
24     G(j)=0.0314*P1a(j)^0.286*((1-0.519*P1a(j))
        ^(-0.286)))^0.5;
25 end
26 xset('window',1)
27 plot(P1a,G);
28 xtitle('Rate of discharge of air vs Cylider Pressure
    ', 'Cylinder pressure P1a (MN/m )', 'Mass flow (kg/
    s)');
29 xset('window',2);
30 P2a=[0 1 2 2.65 3 3.5 4 4.5 4.9 4.95 5];
31 for j=5:11
32     G2(j)=0.2548*P2a(j)^0.714*((1-0.631*P2a(j))
        ^0.286))^0.5;
33 end
34 for i=1:4
35     G2(i)=0.210;
36 end
37 plot(P2a,G2);
38 xtitle('Rate of discharge of air vs Downstream
    Pressure', 'Downstream pressure P2a (MN/m )', 'Mass
    flow (kg/s)')

```

Scilab code Exa 4.2 Calculation of approximate flow rate of fluid

```
1  clc;
2
3  printf("\n Example 4.2\n");
4
5  l=30; //Length of the tube
6  d=150e-3; //Diameter of the tube
7  P1=0.4e3; //Initial Pressure
8  P2=0.13e3; //final Pressure
9  //X=e/d, Relative roughness
10 //Y=R/(rho*u^2) = 0.004
11 X=0.003;
12 Y=0.005;
13 v1=21.15e1;
14
15 G_A=poly([0], 'G_A');
16 f=(G_A^2*log(P1/P2))+((P2^2-P1^2)/(2*P1*v1))+4*(Y*1
    /d)*G_A^2);
17 A=roots(f);
18 printf("\n The approximate flow rate = %.2f kg/m^2 s
    ",A(1));
```

Scilab code Exa 4.3 Calculation of Pressure to be developed at the compressor in order to achieve a given flowrate

```
1  clc;
2
3  printf("\n Example 4.3\n");
4
5  Q=50; //volumetric flow rate of methane
6  P=101.3e3; //Given Pressure
7  T1=288; //Given Temperature
8  d=0.6; //Diameter of pipeline
9  l=3e3; //length of the pipe line
```

```

10 R_R=0.0001; //Relative roughness
11 P2=170e3; //Pressure at which methane is to be
    discharged
12 T2=297; //Temperature at which methane leaves the
    compressor
13 M=16; //molecular mass of methane
14 R=8314; //Gas constant
15 Meu=1e-5; //Viscosity of methane at 293 K
16
17 T=(T1+T2)/2; //Mean temperature
18 P1_v1=R*T/(M);
19 //At 288 K and 101.3 kN/m^2
20 v=P1_v1/P*T1/T;
21 G=Q/v; //Mass flow rate of methane
22 A=%pi/4*d^2; //cross sectional area of pipeline
23 G_A=G/A;
24 Re=G_A*d/Meu;
25 //Y=R/(rho*u^2) = 0.0015
26 Y=0.0015; //(from fig 3.7)
27 //The upstream pressure is calculated using equation
    4.55:
28 function [y]=pressure(P1)
29     y=G_A^(2)*log(P1/P2)+(P2^2-P1^2)/(2*1.5525e5)+4*
        Y*(1/d)*G_A^2;
30     funcprot(0);
31 endfunction
32 P1 = 1e5;
33 z = fsolve(P1,pressure);
34 printf("\n Pressure to be developed at the
    compressor in order to achieve this flowrate = %
    .2f * 10^5 N/m^2",z*1e-5);

```

Scilab code Exa 4.4 Calculation of rate of discharge of gas upstream and downstream pressure and mach number


```

1  clc;
2  clear;
3
4  printf("\n Example 4.4");
5
6  A1=0.007; //cross sectional area of stack pipe
7  A2=4000e-6; //flow area of ruptured disc
8  P1=10e6; //Pressure of the gas in the vessel
9  Gamma=1.4;
10 M=40; //mean molecular weight of gas
11
12 w_c=(2/(Gamma+1))^(Gamma/(Gamma-1));
13 P_c=P1*w_c;
14 V1=(22.4/M)*(500/273)*(101.3e3/P1); // Specific volume
    of the gas in the reactor
15 V=V1*(1/w_c)^(1/Gamma); // Specific volume of gas at
    the throat
16 u=(Gamma*P_c*V)^0.5; //velocity at the throat
17 G=u*A2/V; //initial rate of discharge
18
19 printf("\n (a) Initial rate of discharge of gas = %.1
    f kg/s",G);
20 //obtaining the equations as given in book and
    solving for 'w' we get
21 w=0.0057; //Pressure ratio
22 P_u=P1*w;
23 printf("\n (b) The pressure upstream from the
    shockwave = %.0 f kN/m^2",P_u*1e-3);
24 Mach_no=2.23*(w^(-0.29)-1)^0.5;
25 printf("\n The mach number is = %.2 f",Mach_no);
26 P_s=56.3*w*(w^(-0.29)-1)*1e6;
27 printf("\n (c) The pressure downstream from the
    shockwave = %.0 f kN/m^2",P_s*1e-3);

```

Chapter 5

Flow of Multiphase Mixtures

Scilab code Exa 5.1 Calculation of Pressure drop per unit length of pipe under adiabatic conditions

```
1  clc;
2  clear;
3
4  printf("Example 5.1\n");
5  id=75e-3; // internal diameter of pipe
6  printf("\n Given: \n Internal diameter of pipe = %d
       mm",id*1e3);
7  f_r_s=0.05; // Flow rate of steam in (kg/s)
8  printf("\n Flow rate of steam = %.2f kg/s",f_r_s);
9  f_r_w=1.5; // Flow rate of water in (kg/s)
10 printf("\n Flow rate of water = %.1f kg/s",f_r_w);
11 T=330; // Mean Temperature
12 printf("\n Mean Temperature = %d K",T);
13 P=120; // Mean Pressure drop
14 printf("\n Mean Pressure drop = %d kN/m^2",P);
15 area=%pi*id^2/4; // Cross-sectional area for flow
16 f_r_w_m3s=f_r_w/1000; // Flow of water
17 wtr_vel=f_r_w_m3s/area; //Water velocity
18 rho_steam=18*273*120/(22.4*330*101.3); // density of
       steam at 330 K and 120 kN/m^2
```

```

19 f_r_s_m3s=f_r_s/rho_steam; //Flow of Steam
20 steam_vel=f_r_s_m3s/area; //Steam velocity
21 printf("\n\n Calculations:\n Cross-sectional area
    for flow = %.5f m^2\n Water velocity = %.3f m/s\n
    Steam velocity = %.2f m/s",area,wtr_vel,
    steam_vel);
22 meu_steam=0.0113e-3;
23 meu_water=0.52e-3;
24 Rel=id*wtr_vel*1000/meu_water;
25 Reg=id*steam_vel*rho_steam/meu_steam;
26 printf("\n Reynolds no.(water) = %.2f *10^4",Rel*1e
    -4);
27 printf("\n Reynolds no.(steam) = %.2f *10^4",Reg*1e
    -4);
28 // That is, both the gas and liquid are in turbulent
    flow. From the friction chart (Figure 3.7),
    assuming e/d = 0.00015:
29 // R/(rho*u^2) liq=0.0025      R/(rho*u^2) gas=0.0022
30 // From equation 3.18:
31 DP1=4*0.0025*(1000*wtr_vel^2)/id;
32 DPg=4*0.0022*(rho_steam*steam_vel^2)/id;
33 X=(DP1/DPg)^0.5;
34 phi_l=4.35;
35 phi_g=3.95;
36 DP_tpf=phi_g^2*DPg;
37 printf("\n Pressure drop per unit length of pipe = %
    .0f N/m^2",DP_tpf);

```

Scilab code Exa 5.2 Calculation of maximum mass flow of sand

```

1  clc;
2  clear;
3
4  printf("\n Example 5.2\n");
5

```

```

6 M_p_d=0.2e-3; // Mean particle diameter
7 printf("\n Given:\n Mean particle diameter = %.1f mm
   ",M_p_d*1e3);
8 f_r_w=0.5; //Flow rate of water
9 printf("\n Flow rate of water = %.1f kg/s",f_r_w);
10 id=25e-3; //Diameter of pipe
11 printf("\n Diameter of pipe = %d mm",id*1e3);
12 l=100; //length of pipe
13 printf("\n length of pipe = %d m",l);
14 t_vel=0.0239; //Terminal velocity of falling sand
   particles
15 printf("\n Terminal velocity of falling sand
   particles = %.4f m/s",t_vel);
16 //Assuming the mean velocity of the suspension is
   equal to the water velocity, that is, neglecting
   slip, then:
17 Um=f_r_w/(1000*%pi*id^2/4);
18 printf("\n\n Calculations:\n Mean velocity of
   suspension = %.2f m/s",Um);
19 Re=id*Um*1000/0.001;
20 printf("\n Reynolds no. of water alone = %d",Re);
21 //Assuming e/d = 0.008, then, from Figure 3.7:
22 phi=0.0046;
23 f=0.0092;
24 //From, equation 3.20, the head loss is:
25 hf=4*phi*l*Um^2/(9.81*id);
26 printf("\n Head loss = %.1f m water",hf);
27 iw=hf/l;
28 printf("\n Hydraulic gradient = %.3f m water/m",iw);
29 i=300*1000/(1000*9.81*100);
30 // Substituting in equation 5.20:
31 C=(iw/(i-iw)*(1100*9.81*id*(2.6-1)*t_vel)/(Um^2*Um))
   ^-1;
32 printf("\n C = %.2f",C);
33 //If G kg/s is the mass flow of sand, then:
34 G=poly([0], 'G');
35 p=2600^-1*G-0.30*(2600^-1*G+.0005);
36 printf("\n Mass flow of sand = %.2f kg/s",roots(p));

```

```
37 printf("")
```

Scilab code Exa 5.3 Calculation of Pressure drop

```
1 clc;
2 clear;
3
4 printf("\n Example 5.3");
5
6 p_s=1.25e-3; // Particle size of sand
7 printf("\n Given:\n Particle size of sand = %.2f mm"
      ,p_s*1e3);
8 rho_sand=2600; //Density of sand
9 printf("\n Density of sand = %d kg/m^3",rho_sand);
10 flow_sand=1; //flow rate of sand in air
11 printf("\n flow rate of sand in air = %d kg/s",
      flow_sand);
12 l=200; //length of pipe
13 printf("\n length of pipe = %d m",l);
14 // Assuming a solids:gas mass ratio of 5, then:
15 flow_air=flow_sand/5;
16 vol_flow_air=1*flow_air;
17 printf("\n\n Calculations:\n Volumetric flow rate of
      air = %.2f m^3/s",vol_flow_air);
18 //In order to avoid an excessive pressure drop, an
      air velocity of 30 m/s is acceptable
19 d=100e-3; // taking nearest standard size of pipe
20 // For sand of particle size 1.25 mm and density
      2600 kg/m3, the free-falling velocity is given in
      Table 5.3 as:
21 Uo=4.7;
22 // In equation 5.37:
23 area=%pi*d^2/4;
24 printf("\n The cross-sectional area of a 100 mm ID.
      pipe = %.5f m^2",area);
```

```

25 Ug=flow_air/area;
26 Us=Ug-(Uo/(0.468+(7.25*(Uo/rho_sand)^0.5)));
27 printf("\n Air velocity = %.1f m/s",Ug);
28 printf("\n solids velocity = %.1f m/s",Us);
29 //Taking
30 Meu_air=1.7e-5; // viscosity of air
31 rho_air=1; // Density of air
32 Re=(d*Ug*rho_air/Meu_air);
33 printf("\n Reynolds no. of air alone = %d",Re);
34 phi=0.004;
35 //Assuming isothermal conditions and incompressible
    flow , then , in equation 3.18:
36 DP_air=(4*phi*l/d)*rho_air*Ug^2/2;
37 printf("\n Pressure drop due to air = %.1f kN/m^2",
    DP_air*1e-3);
38 //and in equation 5.38:
39 DP_x=2805*DP_air/(Uo*Us^2);
40 printf("\n Pressure drop due to sand particles = %.1
    f kN/m^2",DP_x*1e-3);
41 DP=DP_air+DP_x;
42 printf("\n The total pressure drop = %.1f kN/m^2",DP
    *1e-3);

```

Chapter 6

Flow and Pressure Measurement

Scilab code Exa 6.1 Calculation of difference in level on a water manometer

```
1  clc;
2  clear;
3
4  printf("\n Example 6.1");
5  d_o=25e-3;//Diameter of orifice
6  printf("\n\n Given:\n Diameter of orifice = %d mm",
       d_o*1e3);
7  d_p=75e-3;//Diameter of pipe
8  printf("\n Diameter of pipe = %d mm",d_p*1e3);
9  flow_o=300e-6;//Flow rate through pipe
10 printf("\n Flow rate through pipe = %d m^3/s",flow_o
       *1e6);
11 Meu_watr=1e-3;//Viscosity of water
12 printf("\n Viscosity of water = %d mN s/m^2",
       Meu_watr*1e3);
13 area_o=%pi/4*d_o^2;//Area of orifice
14 printf("\n\n Calculations:\n Area of orifice = %.2f
       * 10^-4 m^2",area_o*1e4);
```

```

15 vel_o=flow_o/area_o;//Velocity of water through the
    orifice
16 printf("\n Velocity of water through the orifice = %
    .2f m/s",vel_o);
17 Re_o=d_o*vel_o*1000/Meu_watr;//Re at the orifice
18 printf("\n Re at the orifice = %d",Re_o);
19 //From Figure 6.16, the corresponding value of Cd =
    0.61 (diameter ratio = 0.33)
20 Cd=0.61;
21 G=flow_o*1e3; //mass flow rate water
22 //Equation 6.21 may therefore be applied:
23 ho=poly([0], 'ho');
24 p=G^2-((Cd*area_o*1000)^2*2*9.81*ho);
25 printf("\n Difference in level on a water manometer
    = %.0f mm of water",roots(p)*1e3);

```

Scilab code Exa 6.2 Calculation of mass flow rate and drop in pressure

```

1  clc;
2  clear;
3
4  printf("\n Example 6.2\n");
5  rho_sul=1300;//Density of sulphuric acid
6  printf("\n Given:\n Density of sulphuric acid = %d
    kg/ m^3",rho_sul);
7  id=50e-3;//Internal diameter of pipe
8  printf("\n Internal diameter of pipe = %d mm",id*1e3
    );
9  d_o=10e-3;//Diameter of orifice
10 printf("\n Diameter of orifice = %d mm",d_o*1e3);
11 h=.1;//Differential pressure shown on a mercury
    manometer
12 printf("\n Differential pressure shown on a mercury
    manometer = %.1f m",h);
13 Cd=0.61//Coeffecient of discharge

```



```

14 printf("\n Coeffecient of discharge = %.2f",Cd);
15 rho_merc=13550;//Density of mercury
16 printf("\n Density of mercury = %d kg/m^3",rho_merc)
    ;
17 rho_watr=1000;//Density of water
18 printf("\n Density of water = %d kg/m^3",rho_watr);
19 printf("\n\n Calculations:\n (a)");
20 area_o=%pi/4*d_o^2;//area of orifice
21 //The differential pressure is given by:
22 h_sul=h*(rho_merc-rho_sul)/rho_sul;//
23 //The mass flow-rate G is given by:
24 //substituting in equation 6.21 gives the mass
    flowrate as:
25 G_sul=Cd*area_o*rho_sul*(2*9.81*h_sul)^0.5;
26 printf("\n The mass flow rate of acid = %.3f kg/s\n
    (b)",G_sul);
27 DP=rho_sul*9.81*h_sul;
28 printf("\n The drop in pressure = %.0f kN/m^2",DP*1e
    -3);

```

Scilab code Exa 6.3 Calculation of Coefficient for the converging cone of the meter at given flowrate

```

1 clc;
2 clear;
3
4 printf("\n Example 6.3\n");
5 d=150e-3;//Diameter of pipe
6 printf("\n Given:\n Diameter of pipe = %d mm",d*1e3)
    ;
7 d_t=50e-3;//Throat diameter
8 printf("\n Throat diameter = %d mm",d_t*1e3);
9 hv=121e-3;//Pressure drop over the converging
    section
10 printf("\n Pressure drop over the converging section

```

```

        = %d mm of water",hv*1e3);
11 G=2.91; //Mass Flow rate of water
12 printf("\n Mass Flow rate of water = %.2f kg/s",G);
13 //From equation 6.32, the mass rate of flow
14 A1=%pi*d^2/4;
15 A2=%pi*d_t^2/4;
16 Cd=G*(A1^2-A2^2)^0.5/(1000*A1*A2*(2*9.81*hv)^0.5);
17 printf("\n\n Calculations:\n Coefficient for the
        converging cone of the meter at given flowrate =
        %.3f",Cd);

```

Scilab code Exa 6.4 Calculation of flow rate of water

```

1  clc;
2  clear;
3
4  printf("\n Example 6.4\n");
5  l=0.3; //length of tube
6  printf("\n Given:\n length of tube = %.1f m",l);
7  id_t=25e-3; //Top internal diameter of tube
8  printf("\n Top internal diameter of tube = %d mm",
        id_t*1e3);
9  id_b=20e-3; //Bottom internal diameter of tube
10 printf("\n Bottom internal diameter of tube = %d mm"
        ,id_b*1e3);
11 d_f=20e-3; //Diameter of float
12 printf("\n Diameter of float = %d mm",d_f*1e3);
13 v_f=6e-6; //Volume of float
14 printf("\n Volume of float = %d cm^3",v_f*1e6);
15 Cd=0.7; //Coefficient of discharge
16 printf("\n Coefficient of discharge = %.1f",Cd);
17 rho=1000; //Density of water
18 printf("\n Density of water = %d kg/m^3",rho);
19 rho_f=4800; //Density of float
20 printf("\n Density of float = %d kg/m^3",rho_f);

```

```

21 area_t=%pi/4*id_t^2; //Cross-sectional area at top of
    tube
22 printf("\n\n Calculations:\n Cross-sectional area at
    top of tube = %.2f *10^-4 m^2",area_t*1e4);
23 area_b=%pi/4*id_b^2; //Cross-sectional area at bottom
    of tube
24 printf("\n Cross-sectional area at bottom of tube =
    %.2f *10^-4 m^2",area_b*1e4);
25 A_f=%pi/4*d_f^2; //Area of float
26 printf("\n Area of float = %.2f *10^-4 m^2",A_f*1e4)
    ;
27 //When the float is halfway up the tube, the area at
    the height of the float A1 is given by:
28 A1=%pi/4*((id_t+id_b)/2)^2;
29 printf("\n The area of the height of the float A1 is
    = %.2f *10^-4 m^2",A1*1e4)
30 //The area of the annulus A2 is given by:
31 A2=A1-A_f;
32 printf("\n The area of the annulus A2 is = %.2f
    *10^-4 m^2",A2*1e4)
33 //Substituting into equation 6.36:
34 //The flow rate of water =
35 G=Cd*A2*((2*9.81*v_f*(rho_f-rho)*rho)/(A_f*(1-(A2/A1
    )^2)))^0.5;
36 printf("\n\n The flow rate of water = %.3f kg/s",G);

```

Scilab code Exa 6.5 Calculation of volumetric flow rate of water

```

1 clc;
2 clear;
3
4 printf("\n Example 6.5\n");
5
6 L=0.5; // Length of the weir
7 printf("\n Given\n Length of the weir = %.1f m",L);

```

```

8 D=100e-3; //Height of water over the weir
9 printf("\n Height of water over the weir = %d",D*1e3
   );
10 n=0;
11 //Using Francis formula:
12 Q=1.84*(L-(0.1*n*D))*D^1.5;
13 printf("\n\n Calculations:\n Volumetric flowrate of
   water = %.2 f m^3/s",Q);

```

Scilab code Exa 6.6 Calculation of Height of liquid flowing over the weir

```

1 clc;
2 clear;
3
4 printf("\n Example 6.6\n");
5
6 G=15; //Mass flow rate of organic liquid
7 printf("\n Given:\n Mass flow rate of organic liquid
   = %d kg/s",G)
8 L_ow=2; //Length of the weir
9 printf("\n Length of the weir = %.1 f m",L_ow);
10 rho_l=650;
11 printf("\n Density of liquid = %d kg/m^3",rho_l);
12 Q=G/rho_l;
13 //Use is made of the Francis formula (equation 6.43)
   ,
14 h_ow=(2/3)*(Q/L_ow)^(2/3);
15 printf("\n\n Calculations:\n Height of liquid
   flowing over the weir = %.2 f mm",h_ow*1e3);

```

Chapter 7

Liquid Mixing

Scilab code Exa 7.2 Calculation of the power provided by the propeller to the liquid

```
1  clc;
2  clear;
3
4  printf("\n Example 7.2\n");
5
6  rho_sol=1650;//Density of the solution
7  printf("\n Given \n Density of the solution = %d kg/
      m^3",rho_sol);
8  Meu_sol=50e-3;//Viscosity of the solution
9  printf("\n Viscosity of the solution = %d mN s/m^2",
      Meu_sol*1e3);
10 Dt=2.28;//Density of the tank
11 printf("\n Density of the tank = %.2 f m",Dt);
12 D=0.5;//Diameter of the propeller mixer
13 printf("\n Diameter of the propeller mixer = %.2 f m"
      ,D);
14 H=2.28;//Liquid depth
15 printf("\n Liquid depth = %.2 f m",H);
16 Za=0.5;//Height of the propeller
17 printf("\n Height of the propeller = %.1 f m",Za);
```

```

18 N=2; //Rotational speed
19 //In this problem the geometrical arrangement
    corresponds with the configuration for which the
    curves in Figure 7.6 are applicable.
20 Re=D^2*N*rho_sol/(Meu_sol);
21 Fr=N^2*D/9.81;
22 //From figure 7.6
23 Np=0.5;
24 P=Np*rho_sol*N^3*D^5;
25 printf("\n\n Calculations:\n Power provided by
    propeller to the liquid = %.0f W",P);

```

Scilab code Exa 7.3 Determining new power consumption Reynolds number and rotor speed

```

1  clc;
2  clear;
3
4  printf('Example 7.3\n');
5
6  d=0.6; //Tank diameter
7  N1=4; //Rotor speed in Hertz
8  P1=0.15; //Power consumption
9  Re1=160000; //Reynold's number
10 //The correlation of power consumption and Reynolds
    number is given by:
11 //equation (7.13)
12 printf("\n For Constant impeller tip speed \n");
13 D1=d/3;
14 D2=6*D1;
15 N2=%pi*N1*D1/(%pi*D2);
16 printf("\n The new rotor speed = %.2f Hz",N2);
17 //from eq(1) of the solution
18 P2=7.32*N2^3*D2^5;
19 printf("\n The new power required = %.2f kW",P2);

```

```

20 //For thermal similarity , that is the same
    temperature in both systems:
21 Re2=Re1*(N2*D2^2/(N1*D1^2));
22 printf("\n The new reynolds number = %d",Re2);
23
24 printf("\n\n For Constant power input per unit
    volume\n");
25 V2=(%pi/4)*(6*d)^3;
26 P=V2*0.884;
27 printf("\n The new power required = %.2 f kW",P);
28 N=(P/(7.32*(6*0.6/3)^5))^(1/3);
29 printf("\n The new rotor speed = %.2 f Hz",N);
30 Re=Re1*(N*D2^2/(N1*D1^2));
31 printf("\n The new reynolds number = %d",Re);

```

Chapter 8

Pumping of Fluids

Scilab code Exa 8.1 Calculation of the maximum speed at which the pump can run

```
1  clc;
2  clear;
3  printf("\n Example 8.1\n");
4
5  dia_cy=110e-3; //Cylinder diameter
6  printf("\n Given\n Cylinder diameter = %d mm",dia_cy
       *1e3)
7  str=230e-3; //stroke
8  printf("\n Stroke length = %d mm",str*1e3);
9  l_su=6; //Suction line length
10 printf("\n Suction line length = %d m",l_su);
11 d_su=50e-3; //Suction line diameter
12 printf("\n Suction line diameter = %d mm",d_su*1e3);
13 lvl_wtr=3; //level of the water in the suction tank
14 printf("\n level of the water in the suction tank =
       %d m",lvl_wtr);
15 atm_P=10.36;
16 printf("\n Atmospheric pressure is equivalent to = %
       .2f m of water",atm_P);
17 //If the maximum permissible speed of the pump is N
```



```

    Hz:
18 //Angular velocity of the driving mechanism = 2*pi*N
    radians/s
19 //Acceleration of piston = 0.5x0.230(2*pi*N)^2*cos
    (2*pi*N*t) m/s^2
20 //Maximum acceleration(when t=0) = 4.54*N^2 m/s^2
21 //Maximum acceleration of the liquid in the suction
    pipe
22 //          =(.110/.05)^2 x 4.54*N
    ^2 = 21.91*N^2 m/s^2
23 //Accelerating force acting on the liquid
24 //          = 21.97*N^2*pi
    /4*(0.050)^2*(6*1000) N
25 //Pressure drop in suction line due to acceleration
26 //          = 21.97*N^2 *6*1000 N/
    m^2
27 //          = 1.32x 10^5*N^2 N/m^2
28 //          = 13.44*N^2 m water
29 //Pressure head at cylinder when separation is about
    to occur ,
30 x=poly([0], 'x');
31 N=roots(1.20-(10.36-3.0-13.44*x^2));
32 printf("\n\n Calculations:\n Maximum speed at which
    the pump can run = %.3f Hz",N(1));

```

Scilab code Exa 8.2 Calculation of the minimum height required between the liquid level in the reboiler and the pump

```

1  clc;
2  clear;
3  printf("\n Example 8.2\n");
4
5  rho_1=800; //Density of liquid
6  printf("\n Given\n Density of liquid = %d kg/m^3",
    rho_1);

```

```

7 Meu_1=0.5e-3;//Viscosity of liquid
8 printf("\n Viscosity of liquid = %.1f * 10^-3 N s/m
    ^2",Meu_1*1e3);
9 Q=0.0004;//Volumetric flow rate
10 printf("\n Volumetric flow rate = %d m^3/s",Q*1e6);
11 liq_depth=0.07;
12 d=25e-3;//Diameter of pipe used
13 printf("\n Diameter of pipe used = %d",d*1e3);
14 p_v_r=1e3;//Pressure of vapor in reboiler
15 printf("\n Pressure of vapor in reboiler = %d kN/m^2
    ",p_v_r*1e-3);
16 Z=2;//Net Positive Suction Head
17 printf("\n Net Positive Suction Head = %d m",Z);
18 A=%pi/4*d^2;//Cross sectional area of pipe
19 printf("\n\n Calculations:\n Cross sectional area of
    pipe = %.5f m^2",A);
20 u=Q/A;//Velocity in pipe
21 printf("\n Velocity in pipe = %.3f m/s",u);
22 Re=d*u*rho_1/Meu_1;//Reynolds no.
23 printf("\n Reynolds no. = %d ",Re);
24 //From Figure 3.7, the friction factor for a smooth
    pipe is:
25 phi=0.0028;
26 hf_l=(4*phi*u^2)/(d*9.81);
27 printf("\n head loss due to friction per unit length
    = %.4f m/m of pipe",hf_l);
28 //It should be noted that a slightly additional
    height will be required if the kinetic energy at
    the pump inlet cannot be utilised.
29 //Thus the height between the liquid level in the
    reboiler and the pump, HQ, depends on the length
    of pipe between the reboiler and the pump. If
    this is say 10 m
30 l=10;
31 hf=hf_l*l;
32 //equation 8.26 becomes:
33 ho=Z+hf;
34 printf("\n The minimum height required = %.1f m",ho)

```

;

Scilab code Exa 8.3 Calculation of the theoretical power requirements for the compression

```
1 clc;
2 clear;
3 printf("\n Example 8.3\n");
4
5 Q=0.1; //Flow rate of air supplied by compressor
6 printf("\n Given:\n Flow rate of air supplied by
   compressor = %.1f m^3/s",Q);
7 T=273; //Temperature
8 printf("\n Temperature = %d K",T);
9 P=101.3e3; //Pressure
10 printf("\n Pressure = %.1f kN/m^2",P*1e-3);
11 P2=380e3; //Air compressed to a pressure
12 printf("\n Air compressed to a pressure = %d kN/m^2"
   ,P2*1e-3);
13 T2=289; //Suction Temperature
14 printf("\n Suction Temperature = %d K",T2);
15 l=0.25; //Length of the stroke
16 printf("\n Length of the stroke = %.2f m",l);
17 u=4; //Speed
18 printf("\n Speed = %d Hz",u);
19 c=4/100; //Cylinder clearance
20 printf("\n Cylinder clearance = %.2f",c);
21 Gamma=1.4;
22 V=Q*T2/(u*T); //Volume per stroke
23 printf("\n\n Calculations:\n Volume per stroke = %.4
   f m^3",V);
24 R=P2/P; //Compression ratio
25 printf("\n Compression ratio = %.2f",R);
26 //The swept volume is given by equation 8.42
27 Vs=V/(1+c-(c*(R)^(1/Gamma)));
```

```

28 printf("\n The swept volume is = %.4f m^3",Vs);
29 A=Vs/l;//Cross sectional Area of cylinder
30 printf("\n Cross sectional Area of cylinder = %.3f m
    ^2",A);
31 d=(A/%pi*4)^0.5;//Diameter of cylinder
32 printf("\n Diameter of cylinder = %.2f m",d);
33 //From equation 8.41, work of compression per cycle
34 W=P*V*(Gamma/(Gamma-1))*((R)^((Gamma-1)/Gamma)-1);
35 printf("\n Work of compression per cycle = %.0f J",W
    );
36 printf("\n Theoretical power requirements = %.1f kW"
    ,W*4/1e3);

```

Scilab code Exa 8.4 Calculation of the work of compression isothermal and isentropic efficiency and the ratio of swept volumes in two cylinders

```

1  clc;
2  clear;
3  printf("\n Example 8.4\n");
4
5  T=290;//Temperature at which compression takes place
6  printf("\n Given:\n Temperature at which compression
    takes place = %d K",T);
7  P1=101.3e3;//Initial pressure
8  P2=2065e3;//Final pressure
9  printf("\n Compressed from a Pressure of %.1f kN/m^2
    to %d kN/m^2",P1*1e-3,P2*1e-3);
10 eta=.85;//Mechanical efficiency
11 printf("\n Mechanical efficiency = %d percent",eta*1
    e2);
12 c1=4/100;//Clearance in cylinder 1
13 printf("\n Clearance in cylinder 1 = %d percent",c1
    *1e2);
14 c2=5/100;//Clearance in cylinder 1
15 printf("\n Clearance in cylinder 2 = %d percent",c2

```

```

    *1e2);
16 R=P2/P1; // Overall compression ratio
17 printf("\n\n Overall compression ratio = %.1f",R);
18 V_spe=22.4/28.8*T/273; // Specific volume of air at
    290 K
19 printf("\n Specific volume of air at 290 K = %.3f m
    ^3/kg\n (a)",V_spe);
20 W=P1*V_spe*2*(1.25/(1.25-1))*(R^.1-1);
21 //Energy supplied to the compressor, that is the
    work of compression
22 W_act=W/0.85;
23 printf("\n Energy supplied to the compressor, that
    is the work of compression = %.1f kJ/kg",W_act*1e
    -3);
24 printf("\n (b)");
25 //the work done in isothermal compression of 1 kg of
    gas
26 W_it=P1*V_spe*log(R);
27 //Isothermal efficiency
28 eta_it=100*W_it/W_act;
29 printf("\n Isothermal efficiency = %.0f percent",
    eta_it);
30 printf("\n (c)");
31 Gamma=1.4;
32 //the work done in isentropic compression of 1 kg of
    gas
33 W_ie=P1*V_spe*(Gamma/(Gamma-1))*((R)^((Gamma-1)/
    Gamma)-1);
34 //Isentropic efficiency
35 eta_ie=100*W_ie/W_act;
36 printf("\n Isentropic efficiency = %d percent",
    eta_ie);
37 printf("\n (d)");
38 //From equation 8.47, volume swept out in first
    cylinder in compression of 1 kg of gas is given
    by:
39 Vs1=V_spe/(1+c1-(c1*(R)^(1/(2*2.5))));
40 //Similarly, the swept volume of the second cylinder

```

```

        is given by:
41 Vs2=V_spe*(1/R)^0.5/(1+c2-(c2*(R)^(1/(2*2.5))));
42 ratio=Vs1/Vs2;
43 printf("\n the ratio of the swept volumes in the two
        cylinders = %.2f",ratio);

```

Scilab code Exa 8.5 Calculation of the power requirement of the pump

```

1  clc;
2  clear;
3  printf("\n Example 8.5\n");
4
5  Q_1=7.5e-4;
6  printf("\n Given:\n Volume flow rate of liquid = %.1
        f m^3/s",Q_1*1e4);
7  rho_1=1200;
8  printf("\n Density of liquid = %d kg/m^3",rho_1);
9  h=20;
10 printf("\n height to which liquid is raised = %d m",
        h);
11 P=450e3;
12 printf("\n Air is available at pressure = %d kN/m^2"
        ,P*1e-3);
13 eta=30/100;
14 printf("\n Efficiency = %d percent",eta*100);
15 P_atm=101.3e3;
16 Gamma=1.4;
17 G=Q_1*rho_1;//Mass flow of liquid
18 //Work per unit time done by the pump
19 W=G*9.81*h;
20 printf("\n\n Calculations:\n Work per unit time done
        by the pump = %.1f W",W);
21 //Actual work of expansion of air per unit time
22 W_act=W/eta;
23 printf("\n Actual work of expansion of air per unit

```

```

        time = %.1f W',W_act);
24 //Taking the molecular weight of air
25 M=28.9;
26 //the specific volume of air at 101.3 kN/m2 and 273
    K
27 va=22.4/M;
28 //and in equation 8,49:
29 x=poly([0], 'x');
30 Ga=roots(P_atm*va*x*log(P/P_atm)-W_act);
31 Q=Ga*va;
32 printf("\n volume flow rate of air = %.4f m^3/s",Q);
33 //From equation 8.37
34 //Power for compression
35 Power=(P_atm*Q)*(Gamma/(Gamma-1))*((P/P_atm)^((Gamma
    -1)/Gamma)-1);
36 Power_reqd=Power/1000;
37 printf("\n power requirement of the pump = %.3f kW",
    Power_reqd);

```

Scilab code Exa 8.6 Calculation of the efficiency of the pump and the mean velocity of the mixture in the pipe

```

1  clc;
2  clear;
3  printf("\n Example 8.6\n");
4
5  P1=101.3e3;
6  Q_watr=0.01;
7  printf("\n Given:\n Flow rate of Water = %.2f m^3/s"
    ,Q_watr);
8  depth=100;
9  printf("\n Depth of well = %d m",depth);
10 d=100e-3;
11 printf("\n Diameter of pipe = %d mm",d*1e3);
12 depth_watr=40;

```

```

13 printf("\n Level of water below water = %d m",
    depth_watr);
14 Q_air=0.1;
15 printf("\n Flow rate of Air = %.2f m^3/s",Q_air);
16 P2=800e3;
17 Gamma=1.4;
18 //Vl=Q_air;
19 G_watr=Q_watr*1000;//Mass flow of water
20 W=G_watr*depth_watr*9.81;
21 //The energy needed to compress 0. 1 m^3/s of air is
    given by:
22 E=P1*Q_air*(1.4/0.4)*((P2/P1)^(0.4/1.4)-1);//
    equation 8.37
23 printf("\n\n Calculations:\n The power required for
    this compression is = %d W",E);
24 effi=W/E*100;
25 printf("\n Efficiency = %.1f per cent",effi);
26 //the mean pressure
27 P=345e3;
28 printf("\n The mean pressure = %d kN/m^2",P);
29 v1=8314*273/(29*P);
30 printf("\n The specific volume v of air at 273 K and
    given pressure is = %.3f m^3/kg",v1);
31 v2=8314*273/(29*P1);
32 printf("\n The specific volume v of air at 273 K and
    101.3 kN/m^2 is = %.3f m^3/kg",v2);
33 G_air=Q_air/v2; //mass flowrate of the air is:
34 Q_mean=G_air*v1;//Mean volumetric flowrate of air
35 Q_tot=Q_watr+Q_mean;//Total volumetric flowrate
36 A=%pi/4*d^2;//Area of pipe
37 v_mean=Q_tot/A;
38 printf("\n Mean velocity of the mixture = %.2f m/s",
    v_mean);

```

Scilab code Exa 8.7 Calculation of power to be supplied to the pump


```

1  clc;
2  clear;
3  printf("\n Example 8.7\n");
4
5  d=40e-3; //Internal Diameter of the pipe
6  l_p=150; //Lendth of pipe
7  Q_watr=600e-6; //Flow of water
8  h1=10; //Vertical Height
9  h2=2; //head lost across heat exchanger
10 eta=60/100; //Efficiency of pump
11
12 A=%pi/4*d^2; //Area for flow
13 printf("\n Area for flow = %.4 f m^2",A);
14 u=Q_watr/A; //Velocity
15 //At 320 K,
16 Meu=0.65e-3;
17 rho=1000;
18 Re=d*u*rho/Meu;
19 printf("\n Reynolds no. = %d",Re);
20 phi=0.004; //for a relative roughness of 0.005
21 l=l_p+h1+(260*d);
22 printf("\n Equivalent length of pipe = %.1 f m",l);
23 hf=4*phi*l*u^2/(d*9.81);
24 h_tot=hf+h1+h2; //Total head to be developed
25 printf("\n Total head to be developed = %.2 f m",
        h_tot);
26 G=Q_watr*rho; // Mass flow of water
27 P_r=G*h_tot*9.81; //Power Required
28 P_s=P_r/eta; //Power Supplied
29 printf("\n Power Required = %.0 f W",P_s);

```

Scilab code Exa 8.8 Estimation of the rate of flow and the power to be supplied to the pump

```

1  clc;

```

```

2 clear;
3 printf("\n Example 8.8\n");
4
5 eta=0.50;
6 Q=[0.0028 0.0039 0.0050 0.0056 0.0059]
7 h=[23.2 21.3 18.9 15.2 11.0]
8 plot(Q,h,'o-');
9 //The head to be developed , h=10+4.12*u^2 m water
10 //h=10+2.205e5*Q^2
11 Q1=0.0015:0.0001:0.0060
12 h1=10+2.205e5*Q1^2;
13 plot2d(Q1,h1,style=1);
14 xtitle("Data for Example 8.8", "Discharge (Q m^3/s)",
        "Head (m water)");
15 legend("Pump characteristics", "h=10+2.205e5*Q^2");
16 //showing the intersection point
17 x1=[0 0.0054];
18 y1=[16.43 16.43];
19 x2=[0.0054 0.0054];
20 y2=[0 16.43];
21 plot(x1,y1,x2,y2);
22 Q_r=0.0054;
23 printf("\n The discharge at the point of
        intersection between\n the purnp characteristic
        equation = %.4f m^3/s",Q_r);
24 h_r=10+2.205e5*Q_r^2;
25 printf("\n The total head developed = %.2f m",h_r);
26 P=Q_r*1000*h_r*9.81/eta;
27 printf("\n Power required = %.0f W = %.2f kW",P,P*1e
        -3);

```

Scilab code Exa 8.10 Calculation of flow rate and power required by the pump

```

1 clc;

```

```

2 clear;
3 printf("\n Example 8.10\n");
4
5 Meu_H2=0.009e-3; //Viscosity of hydrogen
6 P2=2e6; //Downstream Pressure
7 P1=2.5e6; //Upstream pressure
8 P_m=(P1+P2)/2; //Mean Pressure
9 T=295; //Temperature of the gas
10 l=500; //Length of the pipe used
11 d=50e-3; //diameter of pipe used
12 rho_H2=2*P_m*273/(22.4*101.3e3*T); //Density of
    hydrogen at the mean pressure
13 A=%pi*d^2/4; //Area of the pipe
14 eta=0.60; //Efficiency of the pump
15 v_m=1/rho_H2;
16 //Firstly , an approximate value of G is obtained by
    neglecting the kinetic energy of the fluid
17 //Using equation 4.56
18 //phi*Re^2=7.02 * 10 ^8
19 //Taking the roughness of the pipe surface , e as
    0.00005 m
20 //e/d= 0.001 and Re = 5.7 x 10^5 from Figure 3.8
21 //G=.201(approximate value)
22 //From Figure 3.7,
23 phi=0.0024;
24 //Taking the kinetic energy of the fluid into
    account , equation 4.56 may be used:
25 x=poly([0], 'x');
26 G=roots((x/A)^2*log(P1/P2)+(P2-P1)*rho_H2+4*phi*l/d
    *(x/A)^2);
27 printf("\n Mass flow rate = %.2f kg/s",G(1));
28 P=G(1)*P_m*v_m*log(P1/P2)/eta;
29 printf("\n Power required = %.1f kW",P*1e-3);

```

Chapter 9

Heat Transfer

Scilab code Exa 9.1 Calculation of surface area for counter and cocurrent flow in concentric heat exchanger

```
1  clc;
2  clear;
3
4  printf("\n Example 9.1\n");
5
6  M_dot1=20; //rate of mass to be cooled
7  M_dot2=25; //rate of cooling water
8  Cp=4.18e3; //Heat capacity
9  T1=360; //Initial temp.
10 T2=340; //Final temp.
11 theta_1=300; //Temperature of cooling water entering
12 U=2e3; //Overall heat transfer coefficient
13
14 Q=M_dot1*Cp*(T1-T2); //Heat load
15 printf("\n Heat load = %.0f kW",Q*1e-3);
16 //The cooling water outlet temperature is given by
17 x=poly([0], 'x');
18 theta_2=roots(Q-(M_dot2*Cp*(x-300)));
19 printf("\n The cooling water outlet temperature is =
    %.0f K",theta_2);
```

```

20 printf("\n (a) Counter flow")
21 //In equation 9.9:
22 theta_m1=((T1-theta_2)-(T2-theta_1))/(log((T1-
    theta_2)/(T2-theta_1)));
23 A1=Q/(U*theta_m1)
24 printf("\n The surface area required %.2f m^2",A1);
25 printf("\n (b) Co-current flow")
26 //In equation 9.9:
27 theta_m2=((T1-theta_1)-(T2-theta_2))/(log((T1-
    theta_1)/(T2-theta_2)));
28 A2=Q/(U*theta_m2)
29 printf("\n The surface area required %.2f m^2",A2);

```

Scilab code Exa 9.2 Calculation of the heat loss per square meter of the surface

```

1  clc
2  clear;
3
4  printf("\n Example 9.2\n");
5  dx=0.5; //Thickness of wall
6  T1=400; //Temperartue of inner surface
7  T2=300; //Temperature of outer surface
8  K=0.7; //Thermal conductivity
9  A=1; //Area of heat transfer
10 //From equation 9.12:
11 Q=K*A*(T1-T2)/dx;
12 printf("\n The heat loss per square metre of surface
    = %.0f w/m^2",Q);

```

Scilab code Exa 9.3 Estimation of Heat loss and temperature at firebrick insulating brick interface

```

1  clc;
2  clear;
3
4  printf("\n Example 9.3\n");
5
6  dx1=0.20; //thickness of firebrick
7  dx2=0.10; //thickness of insulating brick
8  dx3=0.20; //thickness of building brick
9  k1=1.4; //Thermal conductivity of firebrick
10 k2=0.21; //Thermal conductivity of insulating brick
11 k3=0.7; //Thermal conductivity of building brick
12 T1=1200; //Temperature at junction 1
13 T4=330; //Temperature at junction 4
14
15 //From equation 9.19:
16 Q=(T1-T4)/((dx1/k1)+(dx2/k2)+(dx3/k3));
17 printf("\n Heat loss per unit area = %d W/m^2",Q);
18 //The ratio (Temperature drop over firebrick)/(Total
    temperature drop)
19 R=(dx1/k1)/((dx1/k1)+(dx2/k2)+(dx3/k3));
20 //Temperature drop over firebrick
21 dT=(T1-T4)*R;
22 printf("\n Temperature drop over firebrick = %.0f K"
    ,dT);
23 T2=(T1-dT);
24 printf("\n The temperature at the firebrick-
    insulating brick interface = %.0f K",T2);

```

Scilab code Exa 9.4 Calculation of the time taken for distant face of brick wall to rise from 295 to 375 K

```

1  clc;
2  clear;
3
4  printf("\n Example 9.4\n");

```

```

5 T=295; //initial temperature of surfaces
6 T2f=375; //Final temperature of far surface
7 dT1=900; //Temperature of near face raised
8 //The temperature at any distance x from the near
   face at time t is given by equation 9.37
9 //Choosing the temperature scale so that the initial
   temperature is everywhere zero , then:
10 R=(T2f-T)/(2*(dT1-T)); //ratio of theta to twice of
   theta dash
11
12 //An approximate solution is obtained by taking the
   first term only , to give:
13 // R=erfc(346*t^-0.5)
14 //erfc(1.30)=R
15 //solving above equation
16 x=poly([0], 'x');
17 t=roots((1.30^2*x)-346^2);
18 printf("\n Time taken to rise from 295 to 375 K = %
   .1 f h",t/3600);

```

Scilab code Exa 9.5 Calculation of the time taken for distant face of brick wall to rise from 295 to 375 K using Schmidts method

```

1 clc;
2 clear;
3
4 printf("\n Example 9.5\n");
5
6 T=295; //initial temperature of surfaces
7 T2f=375; //Final temperature of far surface
8 dT1=900; //Temperature of near face raised
9 DH=4.2e-7; //Thermal diffusivity
10 //The development of the temperature profile is
   shown in Figure 9.12
11 //The problem will be solved by taking relatively

```

```

    large intervals for dx.
12 //Choosing dx = 50 mm, the construction shown in
    Figure 9.12
13 dx=50e-3;
14 //Because the second face is perfectly insulated ,
    the temperature gradient must
15 // be zero at this point.
16 //It is seen that the temperature is
17 //less than 375 K after time 23dt and greater than
    375 K after time 25dt
18 //Thus:
19 //t=24*dt
20 //from equation 9.43
21 dt=dx^2/(2*DH);
22 t=24*dt;
23 printf("\n The time taken to rise from 295 to 375 K
    = %.1f h",t/3600);

```

Scilab code Exa 9.6 Calculation of final temperature of a sphere and a cube

```

1  clc;
2  clear;
3
4  printf("\n Example 9.6\n");
5
6  d=25e-3; //Diameter of copper sphere
7  l=25e-3; //Side length of a copper cube
8  h=75; //External heat transfer coefficient
9  rho_cu=8950; //Density of copper at mean temperature
10 Cp=0.38e3; //Heat capacity of copper at mean
    temperature
11 k=385; // Thermal conductivity of copper at mean
    temperature
12 Tf=923; //Temperature of the furnace

```



```

13 Ta=368; //Temperature at which they are annealed
14 t=5*60; // time taken
15
16 V_Ae_S=(d/6); //V/Ae tor the sphere
17 printf("\n V/Ae tor the sphere = %.2f * 10^-3 m",
        V_Ae_S*1e3);
18 V_Ae_C=(l/6); //V/Ae tor the cube
19 printf("\n V/Ae tor the cube = %.2f * 10^-3 m",
        V_Ae_C*1e3);
20 Bi=h*(V_Ae_S)/k;
21 //The use of a lumped capacity method is therefore
    justified
22 tao=rho_cu*Cp*V_Ae_S/h;
23 //Then using equation 9.49
24 //theta=T
25 x=poly([0], 'x');
26 T=roots(((x-Ta)/(Tf-Ta))-%e^(-t/tao));
27 printf("\n Temperature of the sphere and of the cube
        at the end of 5 minutes = %.0f degree C",T-273);

```

Scilab code Exa 9.7 Calculation of minimum time required for heating a sheet

```

1 clc;
2 clear;
3
4 printf("\n Example 9.7\n");
5
6 k=2.5; //Thermal conductivity
7 DH=2e-7; //Thermal diffusivity of the surrounding
    fluid
8 h=100; //External heat transfer coefficient
9 To=293; //Initial Temperature
10 T_dash=373; //Oven Temperture
11 Tc=353; //temperature throughout the whole of the

```

```

        sheet reaches a minimum
12 l=10e-3; //thickness of sheet
13 L=l/2;
14
15 //For the given process , the Biot number
16 Bi=h*L/k;
17 Bi_1=1/Bi;
18 lim_val=(T_dash-Tc)/(T_dash-To);
19 //From Figure 9.17, the Fourier number
20 Fo=7.7;
21 t=Fo*L^2/DH
22 printf("\n The minimum time for which the sheet must
        be heated = %.0f s or %.0f min approx.",t,t/60);

```

Scilab code Exa 9.8 alculation of the temperature difference between the surface and the centre of the uranium element

```

1 clc;
2 clear;
3
4 printf("\n Example 9.8\n");
5
6 l=5; //Length of the channel of uranium reactor
7 Q=.25e6; //Heat release from uranium reactor
8 k=33; //Thermal conductivity of the uranium
9 Q_m=Q/l; //Heat release rate
10 //Thus, from equation 9.52:
11 dT=Q_m/(4*pi*k);
12 printf("\nThe temperature difference between the
        surface and the centre of the uranium element = %
        .0f deg K",dT);

```

Scilab code Exa 9.9 Calculation of value of scale resistance

```

1  clc;
2  clear;
3
4  printf("\n Example 9.9\n");
5
6  Cp=2380; //specific heat capacity of nitrobenzene
7  k=0.15;
8  Meu=0.70e-3; //Viscosity of nitrobenzene
9  d_i=15e-3; //internal diameter of tube
10 d_o=19e-3; //external diameter of the tube
11 d_s=0.44; //shell diameter
12 b_s=0.150; //baffle spacing
13 p=0.025; //pitch
14 c=0.006; //clearance
15 //(i) Tube side coefficient
16 h_i=1000; //based on inside area
17 h_io=1000*d_i/d_o; //based on outside area
18 //(ii) Shell side coefficient.
19 A=d_s*b_s*c/p; //Area for flow
20 G_s_=4/A;
21 //Taking Meu/Meu_s=1 in equation 9.91
22 d_e=4*((25e-3^2-(%pi*d_o^2/4))/(%pi/d_o));
23 h_o=0.36*k/d_e*(d_e*G_s_/Meu)^0.55*(Cp*Meu/k)^0.33;
24 //(iii) Overall coefficient
25 //The logarithmic mean temperature difference is
    given by:
26 Tm=((400-345)-(315-305))/log((400-345)/(315-305));
27 //The corrected mean temperature difference is
28 Tm_c=Tm*0.8;
29 Q=4*Cp*(400-315);
30 //The surface area of each tube
31 A_t=0.0598;
32 U_o=Q/(2*166*5*A_t*Tm_c);
33 //(iv) Scale resistance.
34 R_d=(1/U_o)-(1/750)-(1/1000);
35 printf("\n Value of scale resistance that could be
    allowed = %.5f m^2 K/W",R_d);

```

Scilab code Exa 9.10 Calculation of pressure drop over tube bundle

```
1 clc;
2 clear;
3
4 printf("\n Example 9.10\n");
5
6 G=15; //Mass flow rate of benzene
7 d_s=1; //Internal diameter of Heat Exchanger
8 l=5; //Length of tubes
9 od=19e-3; //Outer diameter of tubes
10 C=6e-3; //Clearance
11 l_b=0.25; //Baffle spacing
12 Meu=.5e-3;
13 Y=25e-3; //dimension of square pitch
14 N=19; //no. of Baffles
15
16 As=d_s*l_b*C/Y; //Cross-flow area
17 printf("\n Cross-flow area = %.2 f m^2",As);
18 G_dash_s=G/As; //Mass flow
19 printf("\n Mass flow = %d kg/m^2 s",G_dash_s);
20 d_e=4*(Y^2-(%pi*od^2/4))/(%pi*od); //Equivalent
    Diameter
21 printf("\n Equivalent Diameter = %.4 f m",d_e);
22 Re=G_dash_s*d_e/Meu;
23 //From Figure 9.29:
24 f_dash=.280;
25 rho_b=881; //density of benzene
26 DPf=f_dash*G_dash_s^2*(N+1)*d_s/(2*rho_b*d_e);
27 printf("\n The pressure drop over the tube bundle =
    %.0 f N/m^2",DPf);
28 printf("\n\t\t\t\t\t= %.0 f m of Benzene",DPf/(rho_b
    *9.81));
```

Scilab code Exa 9.11 Calculation of heat transfer coefficient

```
1  clc;
2  clear;
3
4  printf("\n Example 9.11\n");
5
6  d=0.15; //Diameter of pipe
7  Ts=400; //Surface temperature
8  Ta=294; //Air temperture
9  //Over a wide range of temperature ,  $k^4*(\beta*g*\rho$ 
    $^2*C_p/(Meu*k)) = 36.0$ 
10 //For air at a mean temperature i.e. 347 K
11 k=0.0310; //Thermal conductivity ---Table 6,
   Appendix A1
12 //X= $\beta*g*\rho^2*C_p/(Meu*k)$ 
13 X=36/k^4;
14 //From Equation 9.102:
15 GrPr=X*(Ts-Ta)*d^3;
16 //From Table 9.5:
17 n=0.25;
18 C_dd=1.32;
19 //Thus, in Equation 9.104:
20 h=C_dd*(Ts-Ta)^n*d^(3*n-1);
21 printf("\n The heat transfer coefficient = %.2f W/m
   ^2 K",h);
```

Scilab code Exa 9.12 Calculation of temperature of a surface coated with carbon black

```
1  clc;
2  clear;
```

```

3
4 printf("\n Example 9.12\n");
5
6 lambda=1e-6; //Wavelength
7 E_l_b=1e9; //Emissive power at given lambda
8
9 //From equation 9.108
10 C2=1.439e-2;
11 C1=3.742e-16;
12 T=C2/lambda/log(C1/(E_l_b*lambda^5));
13 printf("\n The temperature of surface = %d K",T);
14 //With an error of +2 per cent, the correct value is
    given by:
15 E_l_b_n=(100-2)*E_l_b/100;
16 //In equation 9.108:
17 T_n=C2/lambda/log(C1/(E_l_b_n*lambda^5));
18 printf("\n The temperature of surface with +2 per
    cent error= %.0f K",T_n);

```

Scilab code Exa 9.13 Calculation of number of heating elements

```

1 clc;
2 clear;
3
4 printf("\n Example 9.13\n");
5
6 d=10e-3; //Diameter of carbide elements
7 l=0.5; //Length of carbide elements
8 Ts=1750; //Maximun surface temperature of carbide
9 P=500e3; //Thermal power output required
10 sigma=5.67e-8;
11
12 //From equation 9.112, the total emissive power is
    given by:
13 Eb=sigma*Ts^4;

```

```

14 printf("\n The total emissive power is = %.2f *10^5
      W/m^2",Eb*1e-5);
15 A=%pi*d*1;
16 printf("\n The area of one element = %.3f *10^-2 m^2
      ",A*1e2);
17 P1=Eb*A; //Power dissipated by one element
18 printf("\n Power dissipated by one element = %.3f
      *10^3 W",P1*1e-3);
19 n=P/P1; //Number of elements required
20 printf("\n Number of elements required = %.0f ",n);

```

Scilab code Exa 9.14 Calculation of emissivity of a grey surface

```

1  clc;
2  clear;
3
4  printf("\n Example 9.14\n");
5
6  A=10; //Area of the surface
7  P_r=1000e3; //Power radiated
8  T1=1500; //First Temperature
9  T2=1600; //Second Temperatue
10 sigma=5.67e-8;
11
12 E=P_r/A; //The emissive Power
13 printf("\n The emissive Power when T=1500 K = %d kW"
      ,E*1e-3);
14 //From equation 9,118:
15 e=E/(sigma*T1^4);
16 printf("\n Emissivity when T=1500 K = %.3f",e);
17 E2=e*sigma*T2^4;
18 printf("\n The Emissive power when T=1600 K = %.1f
      kW",E2*1e-3);

```

Scilab code Exa 9.15 Calculation of the view factor and net radiation transfer

```
1  clc;
2  clear;
3
4  printf("\n Example 9.15\n");
5
6  A1=2; //Area of rectangle(Surface 1)
7  A2=%pi*1^2/4; //Area of disc (Surface 2)
8  T1=1500; //Temperature of Surface 1
9  T2=750; //Temperature of Surface 2
10 F12=0.25; //View factor
11 sigma=5.67e-8;
12 //From equation 9. 1 26:
13 F21=A1*F12/A2;
14 printf("\n View factor , F12 = %.3 f" ,F21);
15 Q12=sigma*A1*F12*(T1^4-T2^4);
16 printf("\n The net radiation transfer = %.0 f kW" ,Q12
    *1e-3);
```

Scilab code Exa 9.16 Calculation of view factor for 3 dimensional geometry using figures

```
1  clc;
2  clear;
3
4  printf("\n Example 9.16\n");
5
6  printf("\n (a)")
7  //Using the nomenclature in Figure 9.40 iii;
```



```

8 X=4; //width of horizontal plate and length vertical
   plate
9 Y=6; //length of horizontal plate
10 Z=3; //height of verical plate
11 W=Y/X;
12 H=Z/X;
13 A1=Z*X; //Area of plate 1
14 A2=X*Y; //Area of plate 2
15 F12=0.12;
16 printf("\n View Factor , F12= %.2 f",F12);
17 //From equation 9.126:
18 F21=A1*F12/A2;
19 printf("\n View Factor , F21= %.2 f",F21);
20 printf("\n (b)");
21 //For the two spheres
22 r1=1; //Diameter of sphere 1
23 r2=2; //Diameter of sphere 2
24 F12b=1;
25 F21b=(r1/r2)^2;
26 printf("\n View Factor , F21= %.2 f",F21b);
27 F22b=1-F21b;
28 printf("\n View Factor , F22= %.2 f",F22b);

```

Scilab code Exa 9.17 Calculation of view factor of two parallel rings

```

1 clc;
2 clear;
3
4 printf("\n Example 9.17\n");
5
6 ri_u=0.2; // Inner radius of the upper ring
7 ro_u=0.3; // Outer radius of the upper ring
8 ri_l=0.3; // Inner radius of the lower ring
9 ro_l=0.4; // Outer radius of the lower ring
10 //F23 = ((A12/A2)*(F12_34))-F12_4 -((A1/A2)*(F1_34 -

```

```

    F14))
11 //Laying out the data in tabular form and obtaining
    F from Figure 9.40 ii , y, then
12 F12_34=0.4;
13 F12_4=0.22;
14 F1_34=0.55;
15 F14=0.30;
16 A12_A2=ro_l^2/(ro_l^2-ri_l^2);
17 A1_A2=ro_u^2/(ro_l^2-ri_l^2);
18 F23=((A12_A2)*(F12_34-F12_4))+((A1_A2)*(F1_34-F14));
19 printf("\n F23 = %.2f",F23);

```

Scilab code Exa 9.18 Calculation of net rate of heat transfer by radiation to the plate

```

1  clc;
2  clear;
3
4  printf("\n Example 9.18\n");
5
6  d=1; //Diameter of plate
7  r1=0.5;
8  r4=r1; //Radius of the imaginary disc sealing the
    hemisphere
9  L=r1; //The distance between the plate and the
    bottom of the dome
10
11 A1=%pi*d^2/4; //Area of the plate
12 A2=2*%pi*d^2/4; //Area of the underside of the
    Hemisphere
13 A4=%pi*r4^2/4; //Area of an imaginary disc sealing
    the hemisphere and parallel
14 //to the plate
15 T1=750; //Temperature of the plate
16 T2=1200; //Temperature of hemispherical cone

```

```

17 T3=290;//Temperature of the surroundings
18 sigma=5.67e-8;
19 //from equation 9.134, the net radiation to the
    surface of the plate 1 is
20 //given by:
21 //Q1=sigma*A2*F21*(T2^4-T1^4)+sigma*A3*F31(T3^4-T1
    ^4)
22 //using the reciprocity rule:
23 //Q1=sigma*A1*F12*(T2^4-T1^4)+sigma*A3*F31(T3^4-T1
    ^4)
24 //All radiation from the disc 1 to the dome 2 is
    intercepted by the imaginary
25 // disc 4 and hence  $F_{12} = F_{41}$ , which may be obtained
    from Figure 9.39 ii, with
26 //i and j representing areas 1 and 4 respectively
27 R1=r1/L;
28 R4=r4/L;
29 S=1+(1+R4^2)/(R1^2);
30 F14=0.5*(S-(S^2-4*(r4/r1)^2)^0.5);
31 F12=F14;
32 //The summation rule states that
33 //F11+F12+F13=1
34 //F11=0
35 F13=1-F12;
36 Q1=sigma*A1*F12*(T2^4-T1^4)+sigma*A1*F13*(T3^4-T1^4)
    ;
37 printf("\n the net rate of heat transfer by
    radiation to the plate = %.1f kW",Q1*1e-3);

```

Scilab code Exa 9.19 Calculation of radiant heat transfer to the vessel

```

1 clc;
2 clear;
3
4 printf("\n Example 9.19\n");

```

```

5 d=2; //Diameter of the cylinder
6 h=1; //Depth of insulated cylinder
7 A1=%pi*d^2/4; //Radiant heater surface
8 A2=A1; //Under-Surface of the vessel
9 A_R=%pi*d*h;
10 T1=1500;
11 T2=373;
12 //From Figure 9.40 ii , with i = 1, j = 2
13 r1=1;
14 r2=1;
15 L=1;
16 //The view factor may also be obtained from Figure
    9.39 ii as follows:
17 //Using the nomenclature of Figure 9.39
18 R1=r1/L;
19 R2=r2/L;
20 S=1+(1+R2^2)/(R1^2);
21 F12=0.5*(S-(S^2-4*(r2/r1)^2)^0.5);
22 sigma=5.67e-8;
23 //Using the summation rule
24 //F11=0
25 F1R=1-F12;;
26 F2R=F1R;
27 Q2=(A1*F12+((1/(A1*F1R)+(1/(A2*F2R))))^-1)*sigma*(T1
    ^4-T2^4);
28 printf("\n The rate of radiant heat transfer to the
    vessel = %d kW",Q2*1e-3);
29 //If the surroundings without insulation are surface
    3 at
30 T3=290;
31 F23=F2R;
32 //from equation 9.135
33 Q2_d=sigma*A1*F12*(T1^4-T2^4)+sigma*A2*F23*(T3^4-T2
    ^4);
34 printf("\n The rate of radiant heat transfer to the
    vessel\n if the insulation were removed = %.0f kW
    ",Q2_d*1e-3);
35 red=(Q2-Q2_d)/Q2*100; //Percentage Reduction

```

```
36 printf("\n\n Reduction percentage = %.0f per cent",
    red);
```

Scilab code Exa 9.20 Calculation of radiosity net rate of heat transfer and coefficient of heat transfer

```
1  clc;
2  clear;
3
4  printf("\n Example 9.20\n");
5
6  e=0.75; //Emissivity of grey surface
7  r=1-e; //reflectivity of surface
8  Ts=400; //Temperature of surface
9  T_amb=295;
10 sigma=5.67e-8;
11 q1=3e3; //Rate of radiation arriving at grey surface
12 //From equation 9.118
13 Eb=sigma*Ts^4;
14 printf("\n Emissive Power = %.0f W/m^2",Eb);
15 //From equation 9.138
16 qo=e*Eb+r*q1;
17 printf("\n Radiosity = %.0f W/m^2",qo);
18 //From equation 9.140
19 Q_A=e/r*(Eb-qo);
20 q=Q_A;
21 printf("\n The net rate of radiation trasfer = %.0f
    W/m^2",q);
22 printf("\n where the negative value indicates heat
    transfer to the surface.");
23 //For convective heat transfer from the surface
24 qc=-1*q;
25 hc=qc/(Ts-T_amb);
26 printf("\n Coefficient of heat transfer = %.1f W/m^2
    K",hc);
```

Scilab code Exa 9.21 Estimation of the electrical input to the heater and the net rate of heat transfer to the plate

```
1  clc;
2  clear;
3
4  printf("\n Example 9.21\n");
5
6  sigma=5.67e-8;
7  T=[1000 500 300]; //tempertaure of surfaces
8  //Taking surface 1 as the heater , surface 2 as the
   heated plate and surface 3
9  //as an imaginary enclosure
10 A=[1.07 1.07 0.628]; //Array of area of surfaces
11 e=[0.75 0.50 1.0]; //Array of emissivity of the
   surfaces
12 r=[0.250 0.50]; // Array of radius of two surfaces
13 //X is ratio of area to radius(A/r)
14 //Y = A*e/r
15 L=0.2; //distance between two discs
16 for i=1:2
17     X(i)=A(i)/r(i);
18     Y(i)=A(i)*e(i)/r(i);
19     R(i)=r(i)/L;
20 end
21
22 F11=0;
23 F22=0;
24 S=1+(1+R(2)^2)/(R(2)^2);
25 F12=0.5*(S-(S^2-4*(r(2)/(2*r(1)))^2)^0.5);
26 A1_F11=0;
27 A2_F22=0;
28 A1_F12=A(1)*F12;
29 A1_F13=A(1)-(A(1)*F11+A(2)*F12);
```

```

30 //for surface 2:
31 A2_F21=A1_F12;
32 A2_F23=A1_F13;
33 //for surface 3:
34 //By reciprocity rule
35 A3_F31=A1_F13;
36 A3_F32=A2_F23;
37 A3_F33=A(3)-(A3_F31+A3_F32);
38
39 //From equation 9.112:
40 for i=1:3
41     E_b(i)=sigma*T(i)^4/1000;
42 end
43
44 //Since surface 3 is a black body
45 q_o3=E_b(3);
46 //From equations 9.157 and 9.158:
47 //we get
48
49 function [f]=F(x)
50     f(1)=(A1_F11-A(1)/r(1))*x(1)+A2_F21*x(2)+A3_F31*
        q_o3+E_b(1)*A(1)*e(1)/r(1);
51     f(2)=(A1_F12*x(1))+((A2_F22-A(2)/r(2))*x(2))+E_b
        (2)*A(2)*e(2)/r(2);
52     funcprot(0);
53 endfunction
54
55 x=[0 0];
56 q_o=fsolve(x,F);
57
58 //From equation 9.140:
59 Q1=(A(1)*e(1)/r(1))*(E_b(1)-q_o(1));
60 Q2=(A(2)*e(2)/r(2))*(E_b(2)-q_o(2));
61 printf("\n Power input to the heater = %.1f kW",Q1);
62 printf("\n The rate of heat transfer to the plate =
        %.2f kW",Q2);
63 printf("\n where the negative sign indicates heat
        transfer to the plate")

```

Scilab code Exa 9.22 Calculation of net radiation to the walls

```
1  clc;
2  clear;
3
4  printf("\n Example 9.22\n");
5
6  d=0.5; //diameter of chamber
7  l=2; //Length of chamber
8  e=0.5; //Emissivity
9  T_s=750; //Temperature at which the chamber is
    maintained
10 P=150e3;
11 T_g=1250;
12 sigma=5.67e-8;
13
14 //The partial pressures of carbon dioxide (P_c) and
    of water (P_w) are:
15 P_c=0.1*P;
16 P_w=P_c;
17 //From Table 9.7:
18 V=%pi/4*d^2*l; //Volume of the chamber
19 A_s=(2*%pi/4*d^2)+(%pi*d*l); //total surface are of
    chamber
20
21 L_e=3.6*(V/A_s);
22 //FOR WATER VAPOUR
23 //and from Figure 9.44, e_w = 0.075
24 //Since P_w*L_e = 0.06 bar m, then from Figure 9.44:
25 C_w=1.4;
26 e_w1=C_w*0.075;
27 //FOR CARBON DIOXIDE
28 //Since P = 1.5 bar, Pc = 0.15 bar and P_c*L_e =
    0.06 bar m, then, from
```



```

29 //Figure 9.38:
30 //and from Figure 9.45, e_c = 0.037
31 C_c1=1.2;
32 e_c1=(C_c1*0.037);
33 A=(P_w+P_c)*L_e;
34 B=P_c/(P_c+P_w)
35 //Thus, from Figure 9.45 for T_g > 1203 K, De =
    0.001
36 //and, from equation 9.160:
37 De=0.001;
38 e_g=e_w1+0.044-De;
39
40 //FOR WATER VAPOUR
41 //Since 0.5*(P_w+P)=0.825 bar and P_w*L_e*(Ts/Tg)=
    P_c*L_e(Ts/Tg)=0.036 bar m,
42 //then, from Figure 9.44:C_w=1.4
43 e_w2=(0.12*C_w);
44 //and the absorptivity, from equation 9.163 is:
45 a_w=e_w2*(T_g/T_s)^0.65;
46 //FOR CARBON DIOXIDE
47 //From Figure 9.45 at 750 K, e_c=0.08
48 //From Figure 9.45 at P=1.5 bar and P_c*L_e*(Ts/Tg)=
    0.036 bar m:
49 C_c2=1.02;
50 e_c2=(0.08*C_c2);
51 //and the absorptivity, from equation 9.164 is:
52 a_c=e_c2*(T_g/T_s)^0.65;
53 //P_w/(P_c+P_w)=0.5 and (P_c+P_w)*L_e*(T_s/T_g)
    =(0.036+0.036)=0.072 bar m
54 //Thus, from Figure 9.46, for T_g=813 K, De=Da <0.01
    and this may be neglected
55 a_g=a_w+a_c;
56 //If the surrounding surface is black, then:
57 Q=sigma*A_s*(e_g*T_g^4-a_g*T_s^4);
58 printf("\n Radiation to the walls if the surface is
    black = %.1f kW",Q*1e-3);
59 //For grey walls, the correction factor allowing for
    multiple reflection of

```

```

60 //incident radiation is:
61 C_g=0.5/(1-(1-0.326)*(1-0.5));
62 Q_w=(Q*C_g);
63 printf("\n Net radiation to the walls = %.1f kW",Q_w
        *1e-3);

```

Scilab code Exa 9.23 Estimation of the overall coefficient of heat transfer and the dirt factor for the condenser

```

1 clear;
2 clc;
3 printf("\t Example 9.23\n");
4 function [n]=mole(w,m)
5     n = w/m;
6     funcprot(0);
7 endfunction
8
9 function [p]=partial(n1)
10    p = 308*(n1/total);
11    funcprot(0);
12 endfunction
13
14 w_steam = 0.57; //mass flow rate of steam
    entering in [kg/sec]
15 w_CO2 = 0.20; //mass flow rate of CO2
    entering in [kg/sec]
16 m_water = 18; //molecular mass of water in kg
17 m_CO2 = 44; //molecular mass of CO2 in kg
18 n_steam = mole(w_steam,m_water); //number of
    moles in kmol
19 n_CO2 = mole(w_CO2,m_CO2); //number of
    moles in kmol
20 printf("\n At the entrance there is %.3f kmol steam"
    ,n_steam);
21 printf("\n At the entrance there is %.4f kmol water"

```

```

    ,n_CO2);
22 total = n_steam + n_CO2;
23 printf("\n Total number of moles fed to the
    condenser per second = %.4f kmol",total);
24 p_steam = partial(n_steam);
25 p_CO2 = partial(n_CO2);
26 printf("\n At the inlet partial pressure is %d kN/m
    ^2 watre",p_steam);
27 printf("\n At the inlet there is %d kN/m^2 CO2",
    p_CO2);
28 printf("\n From the Table 11A in the appendix Dew
    point = %d K" ,404);
29 mean_mol = (0.57 + 0.20)/total; //mean molecular
    weight of the mixture in kg/kmol
30
31 outlet_steam = 11.7; //partial pressure
    of water in kN/m^2
32 outlet_CO2 = 308 - outlet_steam; //partial pressure
    of water in kN/m^2
33 printf("\n At the outlet partial pressure is %.1f kN
    /m^2 water",outlet_steam);
34 printf("\n At the outlet there is %.1f kN/m^2 CO2",
    outlet_CO2);
35 n_s = n_CO2*outlet_steam/outlet_CO2;
36 steam_condensed = n_steam - n_s;
37 printf("\n steam codensed = %.5f kmol",
    steam_condensed);
38
39 printf("\n\n For the interval 404 to 401 K");
40 p_steam_401K = 252.2; // [kN/m^2]
41 p_CO2_401K = 308 - 252.2; // [kN/m^2]
42 steam_remaining = 0.0045*p_steam_401K/p_CO2_401K;
43 s_c = n_steam - steam_remaining; // [kmol]
44 Heat_cond = s_c*18*(2180 + 1.93*(404-401));
    // [kW]
45 Heat_uncondensed_steam = 0.0203*18*1.93*(404-401);
    // [kW]
46 Heat_CO2 = 0.020*0.92*(404-401);

```

```

47 total_heat = Heat_cond + Heat_uncondensed_steam +
    Heat_CO2;
48 printf("\n Heat of condensation = %d kW",Heat_cond);
49 printf("\n Heat of uncondensed steam = %.1f kW",
    Heat_uncondensed_steam);
50 printf("\n Heat from CO2 = %.1f kW",Heat_CO2);
51 printf("\n Total = %.1f kW",total_heat);
52
53 printf("\n\n For other intervals simiarly");
54 printf("\n Interval(K)           Heat Load(kW)");
55 printf("\n 404 - 401           %.1f           ",
    total_heat);
56 printf("\n 401 - 397           %.1f           ",323.5)
    ;
57 printf("\n 397 - 380           %.1f           ",343.5)
    ;
58 printf("\n 380 - 339           %.1f           ",220.1)
    ;
59 printf("\n 339 - 322           %.1f           ",57.9);
60 printf("\n total               %.1f           ",
    total_heat+323.5+343.5+220.1+57.9);
61 flow_water = 1407.3/(4.187*(319-300));           // [kg/
    sec]
62 hi = 6.36;           // [kW/m^2 K] Based on flow velocity
    of 1425 kg/m^2 sec
63 ho = 5.25;           // [kW/m^2 K] Based on outside area
64 Cp = (0.20*0.92 + 0.57*1.93)/0.77;           // [kJ/kg K]
65 printf("\n Mean specific heat , Cp = %.3f kJ/kg K",Cp
    );
66 k_mean = 0.025;           // [kW/m K]
67 a = 0.0411;           // [m^2]
68 mass_velocity = (0.20+0.57)/0.0411;           // [kg/m^2
    sec]
69 printf("\n Mass velocity = %.1f kg/m^2 sec",
    mass_velocity);
70 hg = 107;           // [W/m^2 K] at Re = 29,800 at
    equivalent diameter = 0.024m
71 u_pD = 0.62;           // (u/pD)^0.67 = 0.62

```

```

72 Cpu_k = 1.01;      //(Cp*u/k)^0.67
73 Psf = (122.6 - 38)/log(122.6/38);
74 kG = hg*(Cpu_k)/(1000*Cp*Psf*u_pD);
75 printf("\n\n kG = %.4f",kG);
76
77 printf("\n Point   Ts   Tc   UT   UTow   Q       A = Q/(UT
   )   T   Tow   (Q/T)ow");
78 printf("\n  1     404   378  309   -       -       -
      84.4 -     -     ");
79 printf("\n  2     401   356  228   268.5  468.4   1.75
      88.1 86.3     5.42");
80 printf("\n  3     397   336  145   186.5  323.5   1.74
      88.6 8.4     3.66");
81 printf("\n  4     380   312  40.6  88.1   343.5   3.89
      76.7 82.7     4.15");
82 printf("\n  5     339   302  5.4   17.5   220.1   12.58
      38.1 55.2     4.00");
83 printf("\n  6     322   300  2.1   3.5    51.9    14.83
      22.2 29.6     1.75");
84
85 printf("\n Assuming no scale resistance , the overall
   coefficient = %.3f W/m^2 K",1407.3/(34.8*74.2));
86 printf("\n The available surface area on the outside
   of the tubes = 0.060 m^2 or %.1f m^2"
   ,246*3.65*0.060);
87 printf("\n Actual coefficient = %.3f kW/m^2 K"
   ,1407.3/(53.9*74.2));
88 printf("\n Dirt factor = %.2f m^2 K/kW"
   ,(0.545-0.352)/(0.545*0.352));

```

Scilab code Exa 9.24 Calculation of overall heat transfer coefficient

```

1  clc;
2  clear;
3

```

```

4 printf("\n Example 9.24");
5
6 d_v=1; //diameter of the vessel
7 L=0.3; //diameter of propeller agitator
8 N=2.5; //rotating speed of propeller agitator
9 T=310; //Temperature
10 G=0.5; //circulation speed of cooling water
11 d_o=25e-3; //outer diameter of stainless steel coil
12 d=22e-3; //inner diameter of stainless steel coil
13 d_w=(d_o+d)/2;
14 d_c=0.8; //diameter of helix
15 T_m=290; //mean temperature
16 k1=0.59;
17 Meu1=1.08e-3;
18 C_p1=4.18e3;
19 x_w=1.5e-3;
20
21 //From equations 9.202 and 9.203, the inside film
    coefficient for the water
22 //is given by:
23 h_i=(k1/d)*(1+3.5*(d/d_c))*0.023*(d*1315/Meu1)^0.8*(
    C_p1*Meu1/k1)^0.4;
24 //The external film coefficient is given by equation
    9.204:
25 C_p2=1.88e3; //Specefic heat capacity
26 Meu2=6.5e-3; //viscosity
27 k2=0.40;
28 rho=1666;
29 Meu_s=8.6e-3;
30 h_o=0.87*(C_p2*Meu2/k2)^(1/3)*(L^2*N*rho/Meu2)
    ^0.62*(Meu2/Meu_s)^0.14*k2/d_v;
31
32 k_w=15.9;
33 R_o=0.0004;
34 R_i=0.0002;
35 U_o=((1/h_o)+(x_w*d_o/(k_w*d_w))+(d_o/(h_i*d))+(R_o)
    +(R_i*d_o/d))^-1;
36 printf("\n\n The overall coeffecient of heat

```

```
transfer = %.0f W/m^2.K",U_o);
```

Scilab code Exa 9.25 To calculate the time required fo heating the liquid

```
1  clc;
2  clear;
3
4  printf("\n Example 9.25")
5
6  C_p=4e3;
7
8  //If T K is the temperature of the liquid at time /
   s, then a heat balance on
9  //the vessel gives:
10 x=poly([0], 'x');
11 T_max=roots((600*0.5)*(393-x)-(10*6)*(x-293));
12 printf("\n\n Maximum temperature to which it can be
   heated = %.1f K",T_max)
13 //solving the equation finally we get
14
15 t1=integrate('11111*(1/(376.3-T))','T',293,353);
16 printf("\n Time taken to heat the liquid from 293 K
   to 353 K = %.0f s",t1);
17 //The steam is turned off for 7200 s and during this
   time a heat balance gives:
18 //on solving as given in book we get
19 T=346.9;
20 //The time taken to reheat the liquid to 353 K is
   then given by:
21 t2=integrate('11111*(1/(376.3-T))','T',346.9,353);
22 printf("\n Time taken to reheat the liquid to 353 K
   = %.0f s",t2);
```

Scilab code Exa 9.26 Calculation of the surface area required to effect the given duty using a multipass heat exchanger

```
1  clc;
2  clear;
3
4  printf("\n Example 9.26\n");
5
6  //As in Example 9.1, the heat load = 1672 kW
7  Q=1672;
8  //With reference to Figure 9.71:
9  T1=360;
10 T2=340;
11 theta1=300; //Temperature of cooling water entering
12 theta2=316;
13 X=(theta2-theta1)/(T1-theta1);
14 Y=(T1-T2)/(theta2-theta1);
15 //from Figure 9.58
16 F=0.97;
17 theta_m=41.9;
18 //and hence:
19 A=Q/(2*F*theta_m); //the heat transfer area
20 printf("\n The heat transfer area is = %.1f m^2",A);
```

Scilab code Exa 9.27 Estimation of the heat transfer area required for the system

```
1  clc;
2  clear;
3
4  printf("\n Example 9.27\n");
5
6  //As in Example 9.1, the heat load = 1672 kW
7  Q=1672;
8  //With reference to Figure 9.71:
```



```

9 T1=360;
10 T2=340;
11 theta1=300; //Temperature of cooling water entering
12 theta2=316;
13 F_theta_m=40.6; //corrected mean temperature
    difference
14 T=(T1+T2)/2;
15 d=1.9e-3; //Tube diameter
16 u=1; //Water velocity
17 //then, in equation 9,221:
18 h_i=4.28*(0.00488*T-1)*u^0.8/d^0.2;
19 //From Table 9.18, an estimate of the shell-side
    film coefficient is:
20 h_o=(1700+11000)/2000;
21 //For steel tubes of a wall thickness of 1.6 mm, the
    thermal resistance of the wall, from Table 9.15
    is:
22 xw_kw=0.025;
23 //the thermal resistance for treated water, from
    Table 9.16, is 0.26 m2K/kW
24 Ri=0.26;
25 Ro=Ri;
26 U=((1/h_o)+xw_kw+Ri+Ro+(1/h_i))^-1;
27 A=Q/(F_theta_m*U);
28 printf("\n The heat transfer area = %.1f m^2",A);

```

Scilab code Exa 9.28 Using Kerns method to design Shell n tube heat exchanger

```

1 clear;
2 clc;
3
4 printf('Example 9.28 ');
5 //Using Kern's method design Shell n tube heat
    exchanger

```

```

6
7 mh = 30;           //[kg/s] Hot fluid flow rate
8 Thi = 370;        //[K] Hot Fluid Inlet Temperature
9 Tho = 315;        //[K] Hot Fluid outlet
   Temperature
10 Tci = 300;       //[K] Cold Fluid Inlet
   Temperature
11 Tco = 315;       //[K] Cold Fluid Outlet
   Temperature
12 cpc = 4.18*10^3;  //[J/kg.K] Thermal
   Conductivity of Cold Fluid
13 //From table A1.3 at mean temperature 343 K
14 cph = 2.9*10^3;  //[J/kg.K] Thermal
   Capacity of Hot fluid
15
16 q = mh*cph*(Thi-Tho);      //[kW] Heat load
17 mc = q/(cpc*(Tco-Tci));    //[kg/s] Flow of
   cooling water
18
19 Tln = ((Thi-Tho)-(Tco-Tci))/(log((Thi-Tho)/(Tco-Tci)
   )); // [K] Logarithm mean temperature
   difference
20
21 //for one-shell side pass and two-tube side pass
   Equation 9.213
22 X = (Thi - Tho)/(Tco-Tci);
23 Y = (Tco-Tci)/(Thi - Tci);
24
25 //From Figure 9.757
26 F = .85;
27 //From Table 9.17
28 U = 500           //[W/m^2.K]
29 A = q/(F*Tln * U);
30 //Thus COnvenient tubes to be used
31 od = .02          //[m] outer dia
32 id = .016         //[m] inner dia
33 l = 4.83          //[m] effective tube length
34

```

```

35 s = %pi*od*1;
36 N = A/s;
37 //From equation 9.211
38 db = (1210/.249)^(2.207)^-1*20/1000;           // [m
    ]
39 //From figure 9.71
40 dc = .068           // [m] diametric clearance between
    shell and tubes
41 ds = db+dc         // [m] Shell dia
42
43 //Tube-Side Coefficient
44 //From equation 9.218
45 Ac = %pi/4*id^2     // [m^2] Cross sectional
    area
46 Ntp = N/2;
47
48 Af = N/2*Ac         // [m^2] Tube side flow area
49 mw = 76.3/Af       // [kg/m^2.s] Mass velocity
    of water
50 rho = 995          // [kg/m^3] mas density of water
51 u = mw/rho         // [m/s] water velocity
52
53 //At mean temperature 308 K
54 vu = .8*10^-3      // [N.s/m^2] viscosity
55 k = .59            // [W/m.K]
56
57 Re = id*u*rho/vu;
58 Pr = cpc*vu/k;
59 ld = 1/id;
60 //from figure 9.77
61 jh = 3.7*10^-3
62 hi = jh*Re*Pr^.3334*.59/id;           // [W/m^2.K]
63
64 //Shell-Side Coefficient
65 //Baffle packing will be taken as 20percent of shell
    dia
66 dbf = .20*ds;     // [m] Baffle Dia
67 tb = 1.25*20*10^-3 // [mm] Tube Pitch

```

```

68 //From equation 9.226
69 As = (25-20)/25*10^3*(ds*Ac) // [m^2]
70 Gs = 30/As; // [kg/m^2.s]
71 //From Equation 9.228
72 de = 1.1*(.025^2-.917*od^2)/od; // [m]
73 //At mean temperature 343 K Butyl Alcohol
74 rho2 = 780 // [kg/m^3] density
75 nu2 = .8*10^-3 // [N.s/m^2] viscosity
76 Cp2 = 3.1*10^3 // [J/kg.K] Heat
    capacity
77 k2 = .16 // [W/m.K]
78 //Equation 9.229
79 Re2 = Gs*de/nu2;
80 Pr2 = Cp2*nu2/k2;
81 //From figure 9.81
82 jh2 = 5*10^-3;
83 //Equation 9.230
84 hs = jh2*Re2*Pr2^.334*k2/de;
85
86 //Overall Coefficient
87 //from Table 9.16
88 k3 = 50 // [W/m.K] Thermal
    Conductivity
89 Rw = .00020 // [m^2.K/W] Scale
    Resistances
90 Ro = .00015 // [m^2.K/W] Resistance
    for organic
91
92 U = [1/hs + Rw + .5*(od-id)/k3 + Ro*od/id+od/(id*hi)
    ]^-1
93
94 //From figure 9.78
95 jf = 4.5*10^-3;
96 n = 2;
97 delP = n*[4*jf*(4.830/id) + 1.25]*(rho*u^2);
98 u2 = Gs/rho2;
99 jf2 = 4.6*10^-2;
100 N2 = 1;

```

```

101 delP2 = N2*[4*jf2*(4.830/od)*(1005/14.2)]*(rho2*u2
      ^2);
102
103 //Increasing the baffle spacing pressure drop is
      reduced one-fourth
104 delPs = delP2/4;
105
106 ho = hs*(.5)^.8; // [W/m^2.K]
107 U2 = [1/ho + Rw + .5*(od-id)/k3 + Ro*od/id+od/(id*hi
      )]^-1
108 printf('\n\n Overall Coefficient of %i W/m^2.K \n
      Number of tubes/pass = %i\n Number of tubes
      required = %i ',U2,Ntp,N)
109 //END

```

Scilab code Exa 9.29 Estimation of Effectiveness of the given double pipe heat exchanger

```

1  clc;
2  clear;
3
4  printf("\n Example 9.29\n");
5
6  G=1; //Flow rate of organic liquid
7  Cp=2e3//Heat capacity of organic liquid
8  T1=350;
9  T2=330;
10 theta1=290;
11 theta2=320;
12
13 Q=G*Cp*(T2-T1); //heat load
14 G_cool=Q/(4187*(theta1-theta2)); //flow of water
15 GCp_hot=(G*Cp); //for organic
16 GCp_cold=(G_cool*4187);
17

```

```

18 //From equation 9.235:
19 eta=GCp_hot*(T1-T2)/(GCp_cold*(T1-theta1));
20 printf("\n Effectiveness of the given double pipe
    heat exchanger = %.2f",eta);

```

Scilab code Exa 9.30 Estimate heat transfer surfaces of One shell pass two tube pass heat exchanger

```

1 clear;
2 clc;
3
4 printf('Example 9.30 ');
5
6 Tci = 320           // [K] Cold Fluid Initial
    Temperature
7 Tce = 340           // [K] Cold Fluid Final Temperature
8 mc = 4             // [kg/s] Flow rate of cold fluid
9 mh = 8             // [kg/s] Flow rate of hot fluid
10 Thi = [380 370 360 350] // [K] Hot fluid initial
    temperature
11 Cp = 4.18          // [kJ/kg.K] mean heat capacity
12 U = 1.5            // [W/m^2.K] Overall heat transfer
    coefficient
13
14 GCpu= mh*Cp;       // [kW/K]
15 GCpp= mc*Cp;       // [kW/K]
16 if(GCpu<GCpp)
17     GCpmin = GCpu; // [kW/K]
18     ratio = GCpmin/GCpp;
19 else
20     GCpmin = GCpp; // [kW/K]
21     ratio = GCpmin/GCpu;
22
23 //Equation 9.235
24 n = mc*Cp*(Tce-Tci)*(mc*Cp*(Thi - Tci))^-1;

```

```

25 //From Figure 9.85b Number of transfer Units
26 N = [.45 .6 .9 1.7]; // [NTU]
27 A = N*GCpmin/U; //Area of required [m^2]
28
29 format('v',4)
30 printf('\n\n Thi(K) n N A (m^2)');
31 disp([Thi(4) n(4) N(4) A(4)], [Thi(3) n(3) N(3) A(3)
    ], [Thi(2) n(2) N(2) A(2)], [Thi(1) n(1) N(1) A(1)
    ])
32 //END

```

Scilab code Exa 9.31 Calculation of gas temperature flowing through a copper pipe

```

1 clc;
2 clear;
3
4 printf("\n Example 9.31\n");
5
6 o_d=10e-3; //outer diameter of the tube
7 i_d=8.2e-3; //inner diameter of the tube
8 h=140; //coeffecient of heat transfer between gas and
    copper tube
9 k=350; //Thermal conductivity of copper tube
10 L=0.075;
11
12 b=%pi*o_d; //perimeter of tube
13 A=%pi/4*(o_d^2-i_d^2); //cross sectional area of the
    metal
14 m=((h*b)/(k*A))^0.5;
15 T_g=((475*cosh(m*L))-365)/(cosh(m*L)-1);
16 printf("\n The gas temperature is = %.0f K", T_g)

```

Scilab code Exa 9.32 Determination of the heat loss per metre run of the tube

```
1  clc;
2  clear;
3
4  printf("\n Example 9.32\n");
5
6  d2=54e-3; //outer diameter of the tube
7  d1=70e-3; // fin diameter
8  w=2e-3; //fin thickness
9  n=230; // number of fins per metre run
10 T_s=370; //Surface temperature
11 T=280; //Temperature of surroundings
12 h=30; //Heat transfer coefficient between gas and
    fin
13 k=43; //Thermal conductivity of steel
14 L=(d1-d2)/2;
15
16 theta1=T_s-T;
17 //Assuming that the height of the fin is small
    compared with its circumference
18 //and that it may be treated as a straight fin of
    length
19 l=(%pi/2)*(d1+d2);
20 b=2*l; //perimeter
21 A=l*w; //the average area at right-angles to the heat
    flow
22 m=((h*b)/(k*A))^0.5;
23 //From equation 9.254, the heat flow is given for
    case (b) as:
24 Qf=m*k*A*theta1*(%e^(2*m*L)-1)/(1+%e^(2*m*L));
25 Q=Qf*n; //Heat loss per meter run of tube
26 printf("\n The heat loss per metre run of tube = %.2
    f kW/m",Q*1e-3);
```

Scilab code Exa 9.33 Calculation of heat loss to the air

```
1  clc;
2  clear;
3
4  printf("\n Example 9.33\n");
5
6  d=150e-3; //Internal diameter of tube
7  d_o=168e-3; //outer diameter of tube
8  d_w=159e-3;
9  d_s=268e-3;
10 d_m=(d_s-d_o)/log(d_s/d_o); //log mean of d_o and d_s
11 h_i=8500; //The coefficient for condensing steam
    together with that for any scale
12 k_w=45;
13 k_l=0.073;
14 x_l=50e-3;
15 x_w=9e-3;
16 DT=444-294;
17 sigma=5.67e-8;
18 //The temperature on the outside of the lagging is
    estimated at 314 K and (hr + hc) will be taken as
    10 W/m2 K.
19 //total thermal resisitance
20 R=(h_i*%pi*d)^-1+(10*%pi*d_s)^-1+(k_w*%pi*d_w/x_w)
    ^-1+(k_l*%pi*d_m/x_l)^-1;
21 Q_l=DT/R; //The heat loss per metre of length(from
    eq 9.261)
22 DT_lagging=((k_l*%pi*d_m/x_l)^-1/R)*DT;
23 //Taking an emissivity of 0.9, from equation 9.119:
24
25 h_r=(0.9*sigma*(310^4-294^4))/(310-294);
26 C=1.32;
27 //Substituting in equation 9.105 (putting l =
```

```

    diameter = 0.268 m):
28 h_c=C*((310-294)/d_s)^0.25;
29 //If the pipe were unlagged ,(hc+hr) for DT=150 K
    would be about 20 W/m2 K and the heat loss would
    then be:
30 Q_1=20*%pi*d_o*150;
31 printf("\n The heat loss to the air = %.2f kW/m",Q_1
    *1e-3);

```

Scilab code Exa 9.34 Determination of the economic thickness of lagging

```

1  clc;
2  clear;
3  printf('Example 9.34\n');
4
5
6  T1=420; //temperature of steam
7  k=0.1; //Thermal conductivity
8  T2=285; //Ambient temperature
9  h=10; //the coefficient of heat transfer from the
    outside of the lagging to
10     //the surroundings
11 //determining Q/l from equation 9.21 and equating it
    to heat loss from the
12 //outside of the lagging we get
13 //(Q/l)=84.82/(log(d_o/0.1)+(0.02/d_o)) W/m
14 //using various equations we finally get an equation
    in terms of d_o and we
15 // will solve it by using fsolve
16 function [f]=F(d_o)
17     f(1)=(1/(log(d_o/0.1)+(0.02/d_o))^2)-(2.35*(d_o
        ^3)/(d_o-0.02));
18     funcprot(0);
19 endfunction
20 d_o=1;

```

```
21 ans=fsolve(d_o,F);
22 E_t=(ans-0.1)/2;
23 printf("\n Economic thickness of lagging = %d mm",
        E_t*1e3);
```

Chapter 10

Mass Transfer

Scilab code Exa 10.1 Estimation of the rate of diffusion of ammonia through the layer

```
1  clc;
2  clear;
3
4  printf("\n Example 10.1\n");
5
6  x=1e-3; //Thickness of stagnant air film
7  D=1.8e-5; //Diffusivity of ammonia
8  R=8314; //Gas constant
9  T=295; //Temperature
10 P=101.3e3; //Total Pressure
11
12 //If the subscripts 1 and 2 refer to the two sides
    of the stagnant layer and
13 //the subscripts A and B refer to ammonia and air
    respectively ,
14 P_A1=.50*P;
15 P_A2=0;
16 P_B1=P-P_A1;
17 P_B2=P-P_A2;
18 P_BM=(P-P_A1)/log(P/P_A1);
```

```

19 //Thus, substituting in equation 10.31 gives:
20 N_A=(-D/(R*T*x))*(P/P_BM)*(P_A2-P_A1);
21 printf("\n The rate of diffusion of ammonia through
    the layer = %.2f *10^-4 kmol/m^2*s",N_A*1e4);

```

Scilab code Exa 10.2 Calculation of the diffusivity of carbon tetrachloride vapour in air

```

1 clear;
2 clc;
3
4 printf("Example 10.2\n");
5
6 th=[0 0 3 7 22 32 46 55 80 106]; //Time in
    hours
7 tm=[0 26 5 36 16 38 50 25 22 25]; //Time in min
8
9 //Conversion to kilo seconds
10 for i=1:10
11     tm(i)=tm(i)*60;
12     th(i)=th(i)*3600;
13     tim(i)=(tm(i)+th(i))/1000;
14 end
15
16 L=[0 2.5 12.9 23.2 43.9 54.7 67.0 73.8 90.3 104.8];
    //in mm
17
18 Lo=L(1);
19
20 //Calculations are done as indicated in the
    procedure
21 //To obtain the values of x and y as below
22 //For plotting x and t axis of graph
23 x=L-Lo;
24

```

```

25 y(1)=0;
26 for j=2:10
27     y(j)=tim(j)/(L(j)-Lo);
28 end
29
30 plot2d(x,y);
31 plot(x,y,'+');
32 xtitle('t/(L-L0) vs (L-L0)', '(L-L0) in mm', 't/(L-L0)
    in ks/mm^2');
33
34 // Calculation of slope
35 s=(y(4)-y(3))/(x(4)-x(3))*10^3*10^6;
36 printf("\nSlope is %.2e sec/m^2\n",s);
37
38 V1=22.4; //Molar volume in litres
39 den=1540; //Density in kg/m^3
40 T0=273;
41 T=321;
42 vp=37.6; //vapour pressure in kPa
43 P0=101.3; //PResue in kPa
44 M=154;
45
46 Ct=T0/(V1*T);
47 Ca=(vp*Ct)/P0;
48
49 Cb1=Ct;
50 Cb2=(P0-vp)*Ct/P0;
51 Cbm=(Cb1-Cb2)/log(Cb1/Cb2);
52
53 // Diffusivity calculation
54 D=den*Cbm/(2*M*Ca*Ct*s);
55 printf("\nDiffusivity is %.2e m^2/s\n",D);
56
57 //End

```

Scilab code Exa 10.3 Calculation of the mass transfer rate per unit area

```
1 clear;
2 clc;
3
4 printf("Example 10.3\n");
5
6 P=101.3e3; //pressure of the operating column
7 T=295; //Temperature of the operating column
8 P_A=7e3; //partial pressure of ammonia
9 x=1e-3; //=(y1-y2) Thickness of stationary gas film
10 D=2.36e-5; //Diffusivity of ammonia
11
12 C_A=(1/22.4)*(273/T)*(P_A/P); //=(C_A1-C_A2)
    Concentration of ammonia gas
13 //X=C.T/C.BM
14 X=P*log(P/(P-P_A))/(P-(P-P_A));
15 //From equation 10.33
16 N_A_=(D/x)*X*(C_A);
17 printf("\n The transfer rate per unit area = %.2 f
    *10^-5 kmol/m^2*s",N_A_*1e5)
```

Scilab code Exa 10.4 Calculation of the maximum length of column to which penetration theory can be applied

```
1 clear;
2 clc;
3
4 printf("Example 10.4\n");
5
6 Q=3e-6; //Flow rate of water
7 Meu=1e-3; //Viscosity of water
8 D=1.5e-9; //diffusivity of carbon dioxide in water
9 rho=1e3; //Density of water
10
```

```

11 //the mean velocity of flow is governed by equation
    3.87 in which sin(phi) is
12 //put equal to unity for a vertical surface:
13 s=(Q*1e2*Meu*3/(rho*9.81))^(1/3); //Thickness of film
14
15 //A=Ux/Us=0.95;
16 A=0.95;
17 y=s*(1-A)^0.5; //The distance below the free surface
18 //using equation 10.108 and using tables of error
    fuctions
19 t=(1.305/1.822)^2
20 Us=rho*9.81*s^2/(2*Meu); //surface velocity
21 L=Us*t; //The maximum lend=gth of column
22 printf("\n The maximum length of column = %.2 f m" ,L)
    ;

```

Scilab code Exa 10.5 Calculation of equivalent resistance and time required

```

1 clear;
2 clc;
3
4 printf("Example 10.5\n");
5
6 N_dot=50; //Initial mass transfer rate
7 D=1.8e-9; //Diffusivity of gas in liquid phase
8
9 C_bg=(1/22.4)*(273/293); //bulk gas concentration
10 N_C=N_dot*C_bg; //Initial mass transfer rate in
    terms of concentration
11 h=N_C/0.04; // Effective Mass transfer coefficient
12 R=1/h; //Equivalent resistance
13 printf("\n Equivalent resistance = %.4 f s/m" ,R);
14 R_l=R*9; //Liquid phase resistance
15 h_l=1/R_l; //Liquid phase coefficient

```



```

16
17 //From equation 10.113 and using liquid phase
    resistance
18 t=R_l^2/(%pi/D);
19
20 printf("\n The required time is = %.2f *10^-11s",t*1
    e11)

```

Scilab code Exa 10.6 Calculating proportion of absorbed carbon dioxide

```

1 clear;
2 clc;
3
4 printf("Example 10.6\n");
5
6 //Diffusivity of CO2 in ethanol
7 D=4D-9; //in m^2/s
8 t=100; //Time in sec
9
10 //Solving all the integral as defined in the proces
11 //as per described in the book
12 //a useful result is obtained
13
14 Cai=poly([0], 'x');
15
16 y=[0 10^-3];
17
18 for i=1:2
19     mole(i)=((2*sqrt(D*t/%pi)*exp(-y(i)^2/(4*D*t)))
        -(y(i)*erfc(y(i)/(2*sqrt(D*t))))));
20 end
21 ret=(mole(1)-mole(2))/mole(1);
22
23 printf("\nProportion retained is %.1f %%\n",ret*100)
    ;

```

24
25 //End

Scilab code Exa 10.8 Calculation of overall mass transfer coefficient based on gas phase

```
1 clc;
2 clear;
3
4 printf("Example 10.8\n")
5
6 L=825e-3; //length of the tube
7 d=15e-3; //diameter of the tube
8 P_i=7.5e3; //Partial pressure of ammonia at inlet
9 P_o=2e3; //Partial pressure of ammonia at inlet
10 A_r=2e-5; //Air rate
11 P=101.3e3; //Atmospheric pressure
12
13 D_F_m=(P_i-P_o)/log(P_i/P_o); //Mean driving force
14 A_absorbd=A_r*((P_i/(P-P_i))-(P_o/(P-P_o)));
15 A_w=%pi*d*L; //Wetted surface
16 K_G=(A_absorbd/(A_w*D_F_m)); //Overall transfer
    coefficient
17 printf("\n Overall Transfer coefficient = %.2f *
    10^-9 kmol/[m^2 s (N/m^2)]",K_G*1e9)
```

Scilab code Exa 10.9 Calculation of numerical value of a given ratio

```
1 clear;
2 clc;
3
4 printf("\n Example 10.9\n");
5
```

```

6 //The proces is defined by
7 //      Ca = B1*exp(sqrt((k/D)y)) + B2*exp(-sqrt((k/D)
      y))
8
9 //Boundary conditions as
10 //      Ca=Cai      at y=0
11 //      Ca=Cai/2    at y=1
12
13 //Using above 3 equations , final equation is derived
      as follows
14
15 //Assuming
16 //      ratio = (Na)y=1 / (Na)y=0
17 //      z=l*sqrt(k/D)
18
19 z=0.693;
20
21 ratio=(exp(sqrt(z))+exp(-sqrt(z))-4)/(2*(1-exp(-sqrt
      (z))-exp(sqrt(z))));
22
23 printf("\n The final ratio is %.2f \n",ratio);
24
25 //End

```

Scilab code Exa 10.10 Determination of order of chemical reaction

```

1 clc;
2 clear;
3
4 printf("Example 10.10\n");
5
6 //The mass transfer rate (moles/unit area and unit
      time) is given by equation
7 //10.180, where denoting the original conditions by
      subscript 1 and the

```

```

8 //conditions at the higher temperature by subscript
   2 gives
9 //N_A2=0.83*N_A1
10 //Substituting the numerical values gives:
11 n=2*(log(0.83/(1.35)^0.5)/log(0.8))-1;
12 printf("\n n = %.2f ",n);
13 printf("\n Thus the reaction is of second order");

```

Scilab code Exa 10.11 Calculate By what factor will the mass transfer rate across the interface change

```

1 clear;
2 clc;
3 //Coulson and Richardson's Chemical Engineering
   Volume I
4 //Chapter 10 Example 11
5 //Page 630
6 printf('Example 10.11 ');
7 //What factor will the mass transfer rate across
   interface change
8
9 k = 2.5*10^-6 // [s^-1] Rate constant
10 E = 2.643*10^7 // [J/kmol] Energy of
   Activation
11 R = 8314 // [J/kmol.K] Universal gas
   contss
12 D = 10^-9 // [m^2/s] MOlecular
   diffuisivity
13 L = .01 // [m] Film Thickness
14
15 //At T =293K
16 T = 293 // [K] temperature
17 A = k/exp(-E/(R*T)); // [s^-1]
18 e = exp(-2*L*sqrt(k/D));
19 N = sqrt(k/D)*(1+e)/(1-e); // Consider relative

```

```

        Solubility at 293 K be unity
20
21 //At T =313K
22 T2 = 313           // [K] temperature
23 k2 = A*exp(-E/(R*T2)); // [s^-1]
24 e2 = exp(-2*L*sqrt(k2/D));
25 N2 = .8*sqrt(k2/D)*(1+e2)/(1-e2); //Consider
        relative Solubility at 313 K be .8 wrt that of
        293K
26
27 Nr = N2/N;
28
29 printf('\n\nChange in mass transfer rate is given by
        factor %.2f',Nr)
30 //END

```

Scilab code Exa 10.12 Estimation of the Thiele modulus and the effectiveness factor for a reactor

```

1  clc;
2  clear;
3
4  printf("\n Example 10.12\n");
5
6  k=5e-4; //first order rate constant
7  D_e=2e-9; //effective diffusivity of reactants in the
        pores of the particles
8
9  lambda=(k/D_e)^0.5;
10 // (i) For the platelet of thickness 8 mm,
11 L=0.5*(8e-3);
12 phi=lambda*L; //thiele modulus
13 //From equation 10.202, the effectiveness factor '
        eta' is given by:
14 eta=(1/phi)*tanh(phi);

```

```

15 printf("\n (i) Thiele modulus = %.1f",phi);
16 printf("\n      The effectiveness factor = %.3f",eta)
    ;
17
18 //(ii) For the sphere of diameter 10 mm, r_o = 0.005
    m^-1.
19 r_o=5e-3;
20 phi_o=lambda*r_o;//Thiele modulus
21 //From equation 10.212, the effectiveness factor '
    eta' is given by:
22 eta_o=(3/phi_o)*(coth(phi_o)-(1/phi_o));
23 printf("\n (i) Thiele modulus = %.1f",phi_o);
24 printf("\n      The effectiveness factor = %.3f",
    eta_o);

```

Scilab code Exa 10.13 Calculation of the effectiveness factor and the concentration of reactant at a given position

```

1  clc;
2  clear;
3  printf("\n Example 10.13\n");
4
5  D_e=1e-5;//Effective diffusivity for the reactants
    in the catalyst particle
6  k=14.4;//first order rate constant
7  L=2.5e-3;
8
9  lambda=(k/D_e)^0.5;
10 phi=(k/D_e)^0.5*(L);//Thiele modulus
11 //From equation 10.202, the effectiveness factor ,
12 eta=(1/phi)*tanh(phi);
13 printf("\n (i) The effectiveness factor = %.3f",eta)
    ;
14 //The concentration profile is given by equation
    10.198

```

```

15 y=1.25e-3;
16 C_Ai=0.15;
17 C_A=C_Ai*(cosh(lambda*y)/cosh(lambda*L));
18 printf("\n (ii) The concentration of reactant at a
           position half-way between the centre and the
           outside of the\n\t pellet = %.3f kmol/m^3",C_A);

```

Scilab code Exa 10.14 Calculation of the effectiveness factor

```

1 clear;
2 clc;
3
4 printf("\n Example 10.14\n");
5
6 R_r=8.2e-2;//reaction rate when concentration =0.011
           kmol/m^3
7 D_e=7.5e-8;//Effective diffusivity
8
9 //Since the value of the first-order rate constant
           is not given, lambda and
10 //phi_L cannot be calculated directly. The reaction
           rate per unit volume of
11 //catalyst = eta*k*C_Ai (equation 10.217),
12 //eta=phi_L^-1
13 //It is assumed that the reactor is operating in
           this regime and the assumption
14 //is then checked.Substituting numerical values in
           equation 10.217:
15 k=(1.217*R_r/0.011)^2;
16 phi_L=1.217*(k)^0.5;
17 eta=phi_L^-1;
18 printf("\n Effectiveness factor = %.4f",eta);

```

Chapter 11

The Boundary Layer

Scilab code Exa 11.1 Calculation of total drag force acting on a surface

```
1  clc;
2  clear;
3
4  printf("\n Example 11.1\n");
5
6  u_s=1; //Velocity of water
7  w=0.6; //Width of plane surface
8  l=1; //Length of plane surface
9  A=0.6*1; //Area of the surface
10 //Taking
11 Meu=1e-3; //Viscosity of water
12 rho=1000; //Density of water
13 //Mean value of S/pw2 from equation 11.41
14 //X=R/(rho*u^2)
15 X=0.00214;
16 F=X*rho*u_s^2*A;
17 printf("\n Total drag force = %.2f N",F);
```

Scilab code Exa 11.2 Calculation of thickness and displacement thickness of the boundary layer

```
1  clc;
2  clear;
3
4  printf("\n Example 11.2\n");
5
6  x=150e-3; //Distance from leading edge where
           thickness is to be found
7  Meu_o=0.05; //viscosity of oil
8  rho_o=1000; //Density of oil
9  u=0.3; //Velocity of flow
10
11 Re_x=x*u*rho_o/Meu_o;
12 //For streamline flow:
13 //from equation 11.17
14 del=4.64*x/Re_x^0.5; //thickness of the boundary
           layer
15 printf("\n The thickness of the boundary layer = %.1
           f mm",del*1e3);
16 //from equation 11.20
17 del_star=0.375*del;
18 printf("\n The displacement thickness = %.1 f mm",
           del_star*1e3);
```

Scilab code Exa 11.3 Calculation of the thickness of the laminar sublayer and velocity of the benzene

```
1  clc;
2  clear;
3
4  printf("\n Example 11.3\n");
5
6  D=50e-3; //Diameter of the pipe
```

```

7 Q=2e-3; //Flow rate of benzene through pipe
8 rho_b=870; //Density of benzene
9 Meu_b=0.7e-3; //Viscosity of benzene
10
11 G=Q*rho_b; //Mass flow rate of benzene
12 Re=4*G/(Meu_b*pi*D); //Reynolds number
13
14 //From equation 11.49:
15 del_b=62*D*Re^(-7/8);
16 printf("\n The thickness of the laminar sub-layer =
    %.3f mm",del_b*1e3);
17
18 area=%pi/4*D^2; //Cross sectional are of pipe
19 u=G/(rho_b*area); //mean velocity
20
21 //From equation 11.47:
22 u_b=2.49*u*Re^(-1/8);
23 printf("\n The velocity of the benzene at the edge
    of the laminar sub-layer = %.3f m/s",u_b);

```

Chapter 12

Momentum Heat and Mass Transfer

Scilab code Exa 12.1 Calculation of thickness of laminar sub layer

```
1  clc;
2  clear;
3
4  printf("\n Example 12.1\n");
5
6  d=250e-3; //internal diameter of pipe
7  u=15; //Velocity of air through the pipe
8  y1=50e-3; //First point where velocity is to be
   found out
9  y2=5e-3; //Second point where velocity is to be
   found out
10 rho_air=1.10; //Density of air
11 Meu_air=20e-6; //Viscosity of air
12
13 Re=d*u*rho_air/Meu_air;
14 //Hence, from Figure 3.7:  $X=R/(\rho*u^2)=0.0018$ 
15 X=0.0018;
16 u_s=u/0.817;
17 u_star=u*X^0.5;
```

```

18
19 //At 50 mm from the wall:
20 y1_r=2*y1/d;// y/r
21 //Hence, from equation 12.34:
22 u_x1=u_s+2.5*u_star*log(y1_r);
23 printf("\n The fluid velocity at 50 mm from the wall
      = %.1 f m/s",u_x1);
24
25 //At 5 mm from the wall:
26 y2_r=2*y2/d;// y/r
27 //Hence, from equation 12.34:
28 u_x2=u_s+2.5*u_star*log(y2_r);
29 printf("\n The fluid velocity at 5 mm from the wall
      = %.1 f m/s",u_x2);
30
31 //The thickness of the laminar sub-layer is given by
      equation 12.54:
32 del_b=5*d/(Re*X^0.5);
33 printf("\n The thickness of the laminar sub-layer =
      %.3 f mm",del_b*1e3);

```

Scilab code Exa 12.2 Estimation of the air temperature at a given point along the pipe

```

1  clc;
2  clear;
3
4  printf("\n Example 12.2\n");
5
6  u=10; //Velocity of air
7  T=330; //Temperature of air
8  d=25e-3; //Inner diameter of pipe
9  T_p=415; //Temperature at which the pipe is
      maintained
10 DP_l=80; //Drop of static pressure along the pipe

```

```

        per unit length
11
12 //From equations 12.98 and 3.18:
13 //we get  $h=0.05 \cdot C_p$ 
14 //The heat taken up per unit time by the air  $dQ$ 
        = $0.0052 \cdot C_p \cdot dT$ .....( i)
15 //The heat transferred through the pipe wall is also
        given by: =  $0.039 \cdot C_p \cdot (415 - T)$  .....( ii)
16 //Equating (i) & (ii)
17 //On integrating we get
18  $T_0=415 - (85 / \exp(0.45))$ 
19 printf(" \n The required air Temperature = %d K", T_0)
        ;

```

Scilab code Exa 12.3 Estimation of outlet temperature by different methods

```

1 clc;
2 clear;
3
4 printf(" \n Example 12.3 \n");
5
6  $u=3.5$ ; //Velocity of water
7  $d=25e-3$ ; //Diameter of the pipe
8  $l=6$ ; //Length of the pipe
9  $T1=300$ ; //Temperature at entrance
10  $T2=330$ ; //Temperature at exit
11  $\rho=1000$ ; //density of water at 310 K
12  $\mu=0.7e-3$ ; //Viscosity of water at 310 K
13 //Taking the fluid properties at 310 K and assuming
        that fully developed flow exists
14  $C_p=4.18e3$ ; //heat capacity
15  $k=0.65$ ; //Thermal conductivity
16
17  $Re=d \cdot u \cdot \rho / \mu$ ;

```

```

18 Pr=Cp*Meu/k;
19
20 printf("\n (a) Reynolds analogy");
21 h1=0.032*(Re^-0.25)*Cp*rho*u;//... Equation 12.139
22 printf("\n h = %.2f kW/m^2 K",h1*1e-3);
23 // on solving we get final equation as
24 theta_dash1=330-10^(log10(30)-(0.0654*h1*1e-3/2.303)
    );
25 printf("\n The outlet temperature = %.1f K",
    theta_dash1)
26
27 printf("\n\n (b) Taylor Prandtl Equation");
28 h2=0.032*(Re^-0.25)*(1+2*Re^(-1/8)*(Pr-1))^-1*Cp*rho
    *u;
29 printf("\n h = %.2f kW/m^2 K",h2*1e-3);
30 // on solving we get final equation as
31 theta_dash2=330-10^(log10(30)-(0.0654*h2*1e-3/2.303)
    );//... Equation 12.140
32 printf("\n The outlet temperature = %.1f K",
    theta_dash2)
33
34 printf("\n\n (c) Universal velocity profile equation
    ");
35 h3=0.032*(Re^-0.25)*(1+0.82*Re^(-1/8)*((Pr-1)+log
    (0.83*Pr+0.17)))^-1*Cp*rho*u;//... equation 12.141
36 printf("\n h = %.2f kW/m^2 K",h3*1e-3);
37 // on solving we get final equation as
38 theta_dash3=330-10^(log10(30)-(0.0654*h3*1e-3/2.303)
    );
39 printf("\n The outlet temperature = %.1f K",
    theta_dash3)
40
41 printf("\n\n (d) Nu=0.023*Re^0.8*Pr^0.33");
42 h4=k/d*0.023*Re^0.8*Pr^0.33;
43 printf("\n h = %.2f kW/m^2 K",h4*1e-3);
44 // on solving we get final equation as
45 theta_dash4=330-10^(log10(30)-(0.0654*h4*1e-3/2.303)
    );

```

```
46 printf("\n The outlet temperature = %.1f K",  
    theta_dash4)
```

Scilab code Exa 12.4 Estimation of outlet temperature by different methods

```
1  clc;  
2  clear;  
3  
4  printf("\n Example 12.4\n");  
5  
6  u=3.5; //Velocity of air  
7  d=25e-3; //Diameter of the pipe  
8  l=6; //Length of the pipe  
9  T1=290; //Temperature at entrance  
10 T2=350; //Temperature at exit  
11 rho=29/22.4*273/310; //density of air at 310 K  
12 Meu=0.018e-3; //Viscosity of air at 310 K  
13 //Taking the physical properties at 310 K and  
    assuming that fully developed flow exists  
14 Cp=1.003e3; //heat capacity  
15 k=0.024; //Thermal conductivity  
16  
17 Re=d*u*rho/Meu;  
18 Pr=Cp*Meu/k;  
19  
20 printf("\n (a) Reynolds analogy");  
21 h1=0.032*(Re^-0.25)*Cp*rho*u; //.... Equation 12.139  
22 printf("\n h = %.2f W/m^2 K",h1);  
23 // on solving we get final equation as  
24 theta_dash1=350-10^(log10(60)-(239.88*h1*1e-3/2.303)  
    );  
25 printf("\n The outlet temperature = %.1f K",  
    theta_dash1)  
26
```

```

27 printf("\n\n (b) Taylor Prandtl Equation");
28 h2=0.032*(Re^-0.25)*(1+2*Re^(-1/8)*(Pr-1))^-1*Cp*rho
    *u;
29 printf("\n h = %.2f W/m^2 K",h2);
30 // on solving we get final equation as
31 theta_dash2=350-10^(log10(60)-(239.88*h2*1e-3/2.303)
    );//....Equation 12.140
32 printf("\n The outlet temperature = %.1f K",
    theta_dash2)
33
34 printf("\n\n (c) Universal velocity profile equation
    ");
35 h3=0.032*(Re^-0.25)*(1+0.82*Re^(-1/8)*((Pr-1)+log
    (0.83*Pr+0.17)))^-1*Cp*rho*u;//...equation 12.141
36 printf("\n h = %.2f W/m^2 K",h3);
37 // on solving we get final equation as
38 theta_dash3=350-10^(log10(60)-(239.88*h3*1e-3/2.303)
    );
39 printf("\n The outlet temperature = %.1f K",
    theta_dash3)
40
41 printf("\n\n (d) Nu=0.023*Re^0.8*Pr^0.33");
42 h4=k/d*0.023*Re^0.8*Pr^0.33;
43 printf("\n h = %.2f W/m^2 K",h4);
44 // on solving we get final equation as
45 theta_dash4=350-10^(log10(60)-(239.88*h4*1e-3/2.303)
    );
46 printf("\n The outlet temperature = %.1f K",
    theta_dash4)

```

Chapter 13

Humidification and Water Cooling

Scilab code Exa 13.1 Pressure calculation

```
1  clc;
2  clear;
3
4  printf("\n Example 13.1\n");
5
6  P=101.3e3;
7  T=297;
8  R=8314; //gas constant
9  RH=60; //Relative humidity
10 p_b1=12.2e3; //Vapor pressure at 297 K
11 p_b2=6e3; //Vapor pressure at 283 K
12 M_w=78; //molecular weight of benzene
13 M_a=28; //Mass of nitrogen
14
15 //From the definition of percentage relative
    humidity (RH)
16 P_w=(p_b1)*(RH/100);
17 //In the benzene -nitrogen mixture:
18 m_b=P_w*M_w/(R*T); //mass of benzene
```

```

19 m_n=(P-P_w)*M_a/(R*T); //mass of nitrogen
20 H=m_b/m_n; //Humidity at 297 K
21
22 //In order to recover 80 per cent of the benzene,
    the humidity must be reduced to 20 per cent of
    the initial value
23 H_o=H*.20;
24 //Thus in equation 13.2
25 P_r=p_b2+(p_b2/M_a*M_w)/H_o;
26 printf("\n The required pressure is = %.0f kN/m^2",
    P_r*1e-3);

```

Scilab code Exa 13.2 Calculation of partial pressure specific volumes humidity humid volume and percentage humidity

```

1  clc;
2  clear;
3
4  printf("\n Example 13.2\n");
5
6  P=101.3e3; //Given pressure
7  T=300; //Given Temperature
8  RH=25; //Percentage relative humidity of water
9  P_wo=3.6e3; //partial pressure of water vapour when
    air is saturated with vapour
10 M_w=18; //Molecular weight of water
11 M_a=29; //Molecular weight of air
12 R=8314; //gas constant
13
14 printf("\n (a)\n The partial pressure of the water
    vapour in the vessel = ")
15 P_w=P_wo*(RH/100);
16 printf("%.1f kN/m^2",P_w*1e-3);
17
18 printf("\n (b)");

```

```

19 m_w=P_w*M_w/(R*T); //mass of water vapour
20 m_a=(P-P_w)*M_a/(R*T); //mass of water air
21 Vs_w=1/m_w; //specific volume of water vapour at 0.9
    kN/m^2
22 Vs_a=1/m_a; //specific volume of air at 100.4 kN/m^2
23 printf("\n Specific volume of water vapour = %.0f m
    ^3/kg",Vs_w);
24 printf("\n Specific volume of air = %.3f m^3/kg",
    Vs_a);
25
26 H=m_w/m_a; //Humidity
27 printf("\n (a)\n Humidity of air = %.4f kg water/kg
    air",H);
28 H_v=Vs_a; //Humid volume
29 printf("\n Humid volume = %.3f m^3/kg",H_v);
30
31 H_p=(P-P_wo)/(P-P_w)*RH; //Percentage humidity
32 printf("\n (d)\n Percentage humidity = %.1f per cent
    ",H_p)

```

Scilab code Exa 13.3 Estimation of the humidity of the air and the percentage relative humidity

```

1  clc;
2  clear;
3
4  printf("\n Example 13.3\n");
5
6  T=310; //Temperature of moist air
7  T_w=300; //Wet bulb tempeature
8  L=2440e3; //Latent heat of vapourisation of water at
    300 K
9  P=105e3; //Given total pressure
10 P_wo1=3.6e3; //Vapour pressure of water vapour at
    300 K

```

```

11 P_wo2=6.33e3; //Vapour pressure of water vapour at
    310 K
12 M_w=18; //Molecular weight of water
13 M_a=29; //Molecular weight of air
14
15 H_w=(P_wo1/(P-P_wo1))*(M_w/M_a); //The humidity of
    air saturated at the wet-bulb temperature
16 //Therefore, taking (h/hD*rho*A) as 1.0 kJ/kg K, in
    equation 13.8:
17 H=H_w-(1e3/L)*(T-T_w);
18 printf("\n The humidity of the air = %.3f kg/kg",H);
19
20 //In equation 13.2:
21 x=poly([0], 'x');
22 P_w=roots(H*(P-x)*M_a-M_w*x);
23 RH=P_w/P_wo2*100;
24 printf("\n The percentage relative humidity (RH)= %
    .1f per cent",RH);

```

Scilab code Exa 13.4 Determination of the temperature of the material on each tray the amount of water removed and the temperature to which the inlet air would have to be raised

```

1  clc;
2  clear;
3
4  printf('Example 13.4 ');
5
6  //Refer HUMIDITY ENTHALPY PLOT Figure 13.5 Page 748
    as Humidity Chart
7  //According the given passes and situatuion
8  T = [325 301 308 312 315]          //[K]
9  H = [.005 .015 .022 .027 .032]    //[kg/kg]
10 //From Humidity Chart on humidifying to 60 percent
    humidity

```

```

11 Tw = [296 301 305 307]           //[K]
12
13 Hin = H(5)-H(1)                 //[kg/kg] Increase in
    Humidity
14
15 printf('\n\n (a) The temperature of the material on
    each tray (in Kelvin)')
16 disp(Tw);
17 printf(' Thus the air leaving the system is at %i K
    and 60 per cent humidity. ',T(5));
18
19 //From Humidity Chart at the obtained leaving
    conditions
20 v = .893                        //[m^3/kg] Specific Volume of dry
    air
21 vs = .968                       //[m^3/kg] Specific Volume of
    Saturated air
22 vh = .937                       //[m^3/kg] Humid Volume of air of
    60 per cent humidity by Interpolation of Curve in
    Humidity Chart
23 x = 5                          //[m^3/s] Amount of moist air leaves
    the dryer in (b)
24 m = x/vh                        //[kg/s] Mass of air passing through
    the dryer
25 mw = m*Hin                      //[kg/s] Mass of water evaporated
26
27 printf('\n\n (b) If 5 m^3/s moist air leaves the
    dryer, The amount of water removed is %.3f kg/s. ',
    ,mw)
28 Tb = 370                        //[K] dry Bulb temperature
    corresponding to humidity of .005 kg/kg and wet-
    bulb temperature 307 K
29 printf('\n\n (c) The Temperature to which the inlet
    air would have to be raised to carry out the
    drying in single stage is %i K. ',Tb)
30
31 //END

```

Scilab code Exa 13.5 Calculation of enthalpy humidity and temperature of resultant stream

```

1  clc;
2  clear;
3
4  printf('Example 13.5\n')
5
6  G1=1; //flow rate of air at 350 K
7  PH1=10; //Percentage Humidity at 350 K
8  G2=5; //flow rate of air at 300 K
9  PH2=30; //Percentage Humidity at 300 K
10
11 //from fig 13.4
12 H1=0.043; //Humidity at 350 K and 10 percent
    humidity
13 H2=0.0065; //Humidity at 300 K and 30 percent
    humidity
14 //Thus, in equation 13.23:
15 H=((G1*H1)+(G2*H2))/(G1+G2);
16 printf(" \n Humidity of final stream = %.4f kg/kg" ,H)
    ;
17
18 //from fig 13.5
19 H_1=192e3; //Entahlpy at 350 K and 10 percent
    humidity
20 H_2=42e3; //Enthalpy at 300 K and 30 percent humidity
21 x=poly([0], 'x');
22 H_=roots((G1*(x-H_1))-(G2*(H_2-x)));
23 printf(" \n Entahlpy of the resultant stream = %.0f
    kJ/kg" ,H_*1e-3);
24
25 //From Figure 13.5:
26 //at H_ (Enthalpy)= 67 kJ/kg and H(humidity) =

```

```

    0.0125 kg/kg
27 T=309;
28 printf("\n Temperature of the resultant stream = %d
    K",T);

```

Scilab code Exa 13.6 Calculation of temperature enthalpy and relative humidity of mixed stream

```

1  clc;
2  clear;
3
4  printf('Example 13.6\n')
5
6  G_s=0.15; //Mass flow rate of steam
7  T=400; //Temperature to which steam is superheated
8  T_a=320; //Temperature of air
9  RH_a=20; //Percentage relative humidity of air
10 G_a=5; //Mass flow rate of air
11 L=2258e3; //latent heat of steam
12 Cp=2e3; //Specific heat of superheated steam
13 //Enthalpy of steam
14 H_3=4.18*(373-273)+L+Cp*(T-373);
15 //From Figure .13.5:
16 //at T=320 K and 20 percent Relative Humidity
17 H1=0.013; //Humidity
18 H_1=83e3; //Enthalpy
19 //By making required constructions we get
20 H=0.043;
21 printf("\n Relative humidity of stream= %.3f kg/kg",
    H);
22 H_=165e3;
23 printf("\n Entahlpy of stream = %d kJ/kg",H_*1e-3);
24 T_s=324;
25 printf("\n Temperature of stream = %d K",T_s);
26

```

```
27 printf("\n\n When exit temperature = 330 K");
28 //from chart and equation 13.28
29 G_case2=0.41;
30 printf("\n The required flow of steam = %.2f kg/s",
        G_case2);
31 printf("\n humidity of this mixture = %.3f kg/kg"
        ,0.094);
```
