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. Harker[1]

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

May 20, 2016

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

Book Description

Title: 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

Chapter 1

Units and Dimensions

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

```
clc;

printf("Example 1.1\n");
// 1 Poise = 1g/cm s = ((1/453.6)lb)/((1/30.48)ft*1s)
be=30.48/453.6*3600; //be->british engineering unit
printf("\n 1 Poise = %.4f lb/ft s",be/3600);
printf("\n          = %.0f lb/ft h",be);

// 1 Poise = 1g/cm s = ((1/1000)kg)/((1/100)m*1s)
si=100/1000; //si->SI units
printf("\n 1 Poise = %.1f kg/m s ",si);
printf("\n          = %.1f N s/m^2 ",si);
```

Scilab code Exa 1.2 Conversion of kW to hp

```
clc;
```

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

```
clc;

printf("Example 2.1\n");
//For 1 kmol of methane

//(a) PV = 1 * RT, where
R=8314;
P=60*10^6;
T=320;
Tc=191;
Pc=4.64*10^6;
printf("\n Given\n R=8314 J/kmol K.\n P=60*10^6 N/m
    ^2\n T=320 K;")
printf("\n Tc=191 K \n Pc=4.64*10^6 N/m^2")
V1=8314*T/P;
printf("\n(a)\n Volume of vessel (ideal gas law) = %
    .4f m^3",V1);


```

```
//(b) In van der Waals equation (2.32), the constants may be taken as:
a=27*R^2*Tc^2/(64*Pc);
b=R*Tc/(8*Pc);
printf("\n(b)\n a = %d (N/m^2)*(m^3)^2/(kmol)^2",a);
printf("\n b = %.4f m^3/kmol",b);
//Thus using equation 2.32:
x=poly([0],'x');
p=roots((60*10^6*x^2+a)*(x-0.0427)-(8314*320*x^2));
printf(" \n Volume of vessel(van der waals eq.) = %.3f m^3",p(1,1));


//(c) Tr=T/Tc  ,Pr=P/Pc
Tr=T/Tc;
printf("\n(c)\n Tr = %.2f",Tr);
Pr=P/Pc;
printf("\n Pr = %.2f",Pr);
//Thus from Figure 2.1,
Z=1.33;
//V = ZnRT/P (from equation 2.31)
V3=Z*R*T/P;
printf("\n Volume of vessel(generalised compressibility-factor chart) = %.4f m^3",V3);
```

Scilab code Exa 2.3 Calculation of reaction force

```
clc;

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

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

Scilab code Exa 2.4 Calculation of resultant force

```
clc;

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

```
clc;

printf("Example 2.5\n");
//From equation 2.68:
// 0.5*((u2)^2-(u1)^2)=g*(z1-z2)+((P1-P2)/rho)
//Suffix 1 to denote conditions in the pipe and
    suffix 2 to denote conditions in the jet
//Symbols have their usual meaning
u1=0;
z1=0;
z2=0;
P1=250*10^3;
P2=0;
rho=1000;//Density of water
printf("\n Suffix 1 to denote conditions in the pipe
     and suffix 2 to denote conditions in the jet")
printf("\n Given:\n u1=%d m/s\n z1= %d m\n z2= %d m\
    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);
g=9.81;
x=poly([0],'x');
u2=roots((0.5*(x)^2)-((P1-P2)/rho));
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

```
clc;

printf("Example 2.6\n");
id=0.5; //internal diameter of pipe
rs=50; //revolution speed
ir=0.15; //internal radius of water
rho=1000; //density of water
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);
omega=2*%pi*rs;
printf("\n\n Angular speed of rotation = %d rad/s",
    omega);
//The wall pressure is given by equation 2.82 as:
wall_pressure=rho*(omega)^2/2*((id/2)^2-ir^2);
printf("\n The wall pressure is = %f N/m^2 \n\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

```
clc;

printf("Example 3.1\n\n");
sap=1.25; //Sulphuric acid pumped
d=25e-3; //Diameter of pipe
l=30; //length of pipe
meu=25e-3; //Viscosity of acid
rho_a=1840; //Density of acid
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);
Re=4*sap/(%pi*meu*d);
printf("\n\n\n Reynolds number , Re=(u*d*rho)/meu =
   4G/(pi*meu*d)= %d",Re);

//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).
//The relative roughness is thus: e/d = 0.002 to 0.02
//From Figure 3.7: R/(rho*u^2) = 0.006 over this range of e/d
u=sap/(rho_a*%pi/4*d^2);
printf("\n Velocity is , u=G/(rho *A) = %.2f m/s",u);

//calculating pressure drop from the energy balance equation and equation 3.19
Dp=rho_a*((0.5+4*0.006*30/0.025)*u^2+9.81*12);
printf("\n Pressure Drop = %.0f N/m^2",Dp);
printf("\n Pressure drop = %.0f kN/m^2",(Dp/10^3));
```

Scilab code Exa 3.2 Maximum allowable water velocity calculation

```
clc;

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

```
e_d=e/(d*1e3);
printf("\n\n phi*Re^2 = %.2f*10^7\n e/d = %.5f",phi_re2*1e-7,e_d);
//From Figure 3.8, for given phi*Re^2 = 1.56 x 10^7 and (e/d) = 0.00026, then:
//Re=7.9*10^4
Re=7.9e4;
u=Re*meu/(rho*d);
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

```
clc;

printf("Example 3.3\n");
Dia_tank=5; //Diameter of the tank
len_pipe=100; //Length of pipe
dia_pipe=225e-3; //Diameter of pipe
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);

//If at time t the liquid level is D m above the bottom of the tank, then
// designating point 1 as the liquid level and point 2 as the pipe outlet,
// and applying the energy balance equation (2.67) for turbulent flow, then:

// The equation becomes (u2^2/2)-D*g+(4*R*len_pipe*u2^2/(rho*u^2*dia_pipe))

//As the level of liquid in the tank changes from D to (D + dD), the quantity
```

```
// of fluid discharged = (pi/4)5^2(−dD) = −19.63dD m
  ^3.
// The time taken for the level to change by an
  amount dD is given by:
//dt=19.63dD/((pi/4)0.225^2 x 4.43D^0.5/([1 + 3552(/
  R/rho*u^2)])^0.5)

// calculating the value of X=R/rho*u^2 as given in
  example (refer to book)
X=.0020;
t=integrate('111.5*(1+(3552*X))^0.5*D^−0.5','D'
  ,0.3,3);
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

```
clc;

printf("Example 3.4\n");

d1=0.3; //diameter of pipe from junction A to D or B
   to D
l1=1.5e3; //length of pipe from junction A to D or B
   to D
d2=0.5; // diameter of pipe from junction D to C
l2=0.75e3; // length of pipe from junction D to C
h_A=10; // height of tank A above C
h_B=h_A+6; // height of tank A above C
rho=870; // density of liquid
Meu_l=0.7e-3; // viscosity of liquid

//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
//respectively ,
//Taking the roughness of mild steel pipe e as 0.00005 m, e/d varies from
//0.0001 to 0.00017. As a first approximation , R/(rho*u^2) may be taken as 0.002
//Then applying the energy balance equation between D and the liquid level in
//each of the tanks gives
//On forming and solving the equations

x=poly([0],'x');
u2=roots(x^4-(7.38*x^2)+13.57);
u1=(u2^2-1.47)^0.5;
u3=(u1+u2)/2.78;
//taking the positive values and which satisfy equation 7
U1=u1(4);
U2=u2(4);
U3=u3(4);
Q=%pi/4*d2^2*U3;
printf("\n The volumetric flow rate = %.2f m^3/s",Q);
```

Scilab code Exa 3.5 Calculation of radius for given condition

```
clc;

printf("Example 3.5\n");

// Ux = Ucl*(y/r)^1/7      equation 3.59 (Prandtl one-seventh power law)
//where UCL is the velocity at the centre line of the pipe, and r is the radius of the pipe.
// Then total flow, Q = 49/60*pi*r^2*Ucl
```

```
    equation 3.62

//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:
//Qc=pi*r^2*Ucl*(105*a^(8/7)-56*a^(15/7))/60
//flow in the core = 0.5 (flow in the whole pipe)

r=50;
a=poly([0],'a');
p=roots((a^8*(105-56*a)^7)-24.5^7);

printf("\n a = %.2f",p(8));
y=p(8)*r;
printf("\n y = %.1f mm",y)
```

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

```
clc;

printf("Example 3.6\n");

Q=7.2;//Water flow rate
d1=40e-3; //initial pipe diameter
d2=50e-3; //diameter of pipe after enlargement
g=9.81;
printf("\n Given\n Water flow rate = %.1f m^3/h\n d1
     = %d mm\n d1 = %d mm",Q,d1*1e3,d2*1e3);

u1=(Q/3600)/(%pi/4*d1^2);//Velocity in 40 mm pipe
u2=(Q/3600)/(%pi/4*d2^2);//Velocity in 50 mm pipe
printf("\n Velocity in 40 mm pipe = %.2f m/s\n
    Velocity in 50 mm pipe = %.2f m/s",u1,u2);
// The head lost is given by equation 3.77 as:
```

```
hf=(u1-u2)^2/(2*g);
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

```
clc;

printf("Example 3.7\n");
Q_h=2.27; // flow rate of water in m^3/h
T=320; //Temperature of water to be pumped
id=40e-3; //internal diameter of pipe
l_h=150; //length of pipe horizontally
l_v=10; //length of pipe vertically
e=0.2e-3;
g=9.81;
rho=1000;
printf("\n Given\n flow rate of water in m^3/h = %.2f 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);

rel_rough=e/id; //Relative roughness
printf("\n\n Relative roughness = %.3f",rel_rough);
meu=0.65e-3; //Viscosity at 320 K
Q_s=Q_h/3600; //flow rate of water in m^3/s
area=%pi/4*id^2; // Area for flow
printf("\n Area for flow = %.2f * 10^-3 m^2",area*1e3);
u=Q_s/area; //Velocity
printf("\n Velocity = %.2f m/s",u);
Re=(id*u*rho)/meu;
printf("\n Reynolds No. = %d",Re);

```

```
//X=R/(rho*u^2)=0.004 (from Figure 3.7)
X=.004;
equi_len=l_h+l_v+(260*id); // Equivalent length of
    pipe
printf("\n Equivalent length of pipe = %.1f m",
    equi_len);
hf=4*X*equi_len*u^2/(id*g);
tot_head=hf+1.5+10; // Total head to be developed
printf("\n Total head to be developed = %.2f m",
    tot_head);
mass_thr=Q_s*rho; //Mass throughput
printf("\n Mass throughput = %.2f kg/s",mass_thr);
power_reqd=(mass_thr*tot_head*g)/0.60;
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

```
clc;
printf("Example 3.8\n");

d=0.15; // diameter of pipe
g=9.81;
printf("\n Given\n Diameter of pipe = %.2f",d);
// From equation 3.20, the head lost due to friction
     is given by:
// hf = 4*phi*l*u^2/(d*g)m water
// The total head loss is:
// h=(u^2/(2*g))+hf+loss in fittings
// From Table 3.2., the losses in the fittings are:
    From Table 3.2., the losses in the fittings are:
//6.6*u^2/(2*g)
//Taking
phi=.0045;
x=poly([0],'x');
```

```
u=roots((7.6+4*phi*(105/.15))*x^2/(2*g)-10);
printf("\n\n Velocity = %.2f m/s",u(1));
rate_dis=u(1)*%pi*d^2/4;
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

```
clc;

printf("Example 3.9\n");

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

```
clc;
```

```

printf("Example 3.10\n");

k=10;
n=0.2;
//Using the power-law model (equation 3.121):
printf("\n Given:\n Consistency coefficient k = %d N.s^n/m^-2",k);
printf("\n Flow behaviour index = %.1f",n);
Ucl=1; // centre line velocity
printf("\n Centre line velocity = %d m/s",Ucl);
l=200; // length of pipe
printf("\n Length of pipe = %d m",l)
r=.02; // radius of pipe
printf("\n Radius of pipe = %.2f m",r);
dux_dy_1=10;
dux_dy_2=50;
Ry_1=k*dux_dy_1^0.2;
Ry_2=k*dux_dy_2^0.2;
//Using the Bingham-plastic model (equation 3.125):
A=[1 10;1 50]
B=[15.85;21.87]
C=inv(A)*B;
Ry=C(1);
Meu_p=C(2);
printf("\n\n Plastic viscosity (Meu_p) = %.3f N s/m^2",C(2));
printf("\n Yeild stress (Ry) = %.2f N s/m^2",C(1));
// Using Equation 3.131
DP=2*k*l*Ucl^n*((n+1)/n)^n*r^(-n-1);
printf("\n Pressure drop (Bingham plastic model)= %.0f kN/m^2",DP/1e3);
// For a Bingham-plastic fluid:
// The centre line velocity is given by equation 3.145:
X=(l*2*Ry)/(r*DP);
Up=(DP*r^2*(2-4*X+2*X^2))/(8*Meu_p*l);
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

```
clc;

printf("Example 3.11\n");
// given:
Meu=0.1; // Viscosity of liquid
printf("\n Given \n Viscosity of liquid = %.1f N s/m
    ^2",Meu);
d=25e-3; // Diameter of pipe
printf("\n Diameter of pipe = %.3f m",d);
l=20; // length of pipe
printf("\n length of pipe = %d m",l);
DP=1e5; // Pressure drop
printf("\n Pressure drop = %d N/m^2",DP);
n=1/3; // flow index of polymer solution
printf("\n flow index = %.2f",n);
dux_dy=1000;
k=Meu;
Meu_a=Meu;
k_poly_sol=Meu_a/(dux_dy)^(n-1);
Ry=10*(dux_dy)^n;
//From equation 3.136:
//For a power-law fluid:
u2=((DP/(4*k_poly_sol*l))^3)*(n*(d^((n+1)/n)))
    /(2*(3*n+1));
printf("\n\n Velocity for polymer solution = %.4f m/
    s",u2);
u1=(DP/(4*k*l))*(d^2)/8
printf("\n Velocity for original solution = %.3f m/s
    ",u1);
ratio=u2/u1;
```

```
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

```
clc;
clear;

printf("\n Example 4.1\n");

d=0.006;//Diameter of the cylinder
Gamma=1.4;
//The critical pressure ratio for discharge through
   the valve
C_r=(2/(Gamma+1))^(Gamma/(Gamma-1));
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.")
//(i) Sonic velocity will occur until the
P_c=101.3/C_r;//pressure at which sonic velocity
   will occur
M=29;//molecular mass of air
//The rate of discharge for cylinder pressures
```

```
    greater than 191.1 kN/m^2 is
//given by equation 4.30: taking mean value for
    Gamma i.e. 1.47
//we get
//G_max=4.23e-8*P1 kg/s
//For cylinder pressures below 191.1 kN/m2, the mass
    flowrate is given by equation 4.20
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]
for i=5:13
    G(i)=4.23e-2*P1a(i);
end
for j=1:4
    G(j)=0.0314*P1a(j)^0.286*((1-0.519*P1a(j)
        ^(-0.286)))^0.5;
end
xset('window',1)
plot(P1a,G);
xtitle('Rate of discharge of air vs Cylider Pressure
    ','Cylinder pressure P1a (MN/m )','Mass flow (kg/
    s)');
xset('window',2);
P2a=[0 1 2 2.65 3 3.5 4 4.5 4.9 4.95 5];
for j=5:11
    G2(j)=0.2548*P2a(j)^0.714*((1-0.631*P2a(j)
        ^0.286))^0.5;
end
for i=1:4
    G2(i)=0.210;
end
plot(P2a,G2);
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

```
clc;

printf("\n Example 4.2\n");

l=30;//Length of the tube
d=150e-3;//Diameter of the tube
P1=0.4e3;//Initial Pressure
P2=0.13e3;//final Pressure
//X=e/d, Relative roughness
//Y=R/(rho*u^2) = 0.004
X=0.003;
Y=0.005;
v1=21.15e1;

G_A=poly([0],'G_A');
f=(G_A^2*log(P1/P2))+((P2^2-P1^2)/(2*P1*v1))+(4*(Y*l
   /d)*G_A^2);
A=roots(f);
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

```
clc;

printf("\n Example 4.3\n");

Q=50; //volumetric flow rate of methane
P=101.3e3;//Given Pressure
T1=288;//Given Temperature
d=0.6;//Diameter of pipeline
l=3e3;//length of the pipe line
```

```
R_R=0.0001;//Relative roughness
P2=170e3;//Pressure at which methane is to be discharged
T2=297;//Temperature at which methane leaves the compressor
M=16;//molecular mass of methane
R=8314;//Gas constant
Meu=1e-5;//Viscosity of methane at 293 K

T=(T1+T2)/2;//Mean temperature
P1_v1=R*T/(M);
//At 288 K and 101.3 kN/m^2
v=P1_v1/P*T1/T;
G=Q/v;//Mass flow rate of methane
A=%pi/4*d^2;//cross sectional area of pipeline
G_A=G/A;
Re=G_A*d/Meu;
//Y=R/(rho*u^2) = 0.0015
Y=0.0015;//(from fig 3.7)
//The upstream pressure is calculated using equation 4.55:
function[y]=pressure(P1)
    y=G_A^(2)*log(P1/P2)+(P2^2-P1^2)/(2*1.5525e5)+4*Y*(l/d)*G_A^2;
    funcprot(0);
endfunction
P1 = 1e5;
z = fsolve(P1,pressure);
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

```
clc;
clear;

printf("\n Example 4.4");

A1=0.007;//cross sectional area of stack pipe
A2=4000e-6;//flow area of ruptured disc
P1=10e6;//Pressure of the gas in the vessel
Gamma=1.4;
M=40;//mean molecular weight of gas

w_c=(2/(Gamma+1))^(Gamma/(Gamma-1));
P_c=P1*w_c;
V1=(22.4/M)*(500/273)*(101.3e3/P1);//Specific volume
    of the gas in the reactor
V=V1*(1/w_c)^(1/Gamma);//Specific volume of gas at
   the throat
u=(Gamma*P_c*V)^0.5;//velocity at the throat
G=u*A2/V;//initial rate of discharge

printf("\n (a)Initial rate of discharge of gas = %.1
   f kg/s",G);
//obtaining the equations as given in book and
   solving for 'w' we get
w=0.0057;//Pressure ratio
P_u=P1*w;
printf("\n (b)The pressure upstream from the
   shockwave = %.0f kN/m^2",P_u*1e-3);
Mach_no=2.23*(w^(-0.29)-1)^0.5;
printf("\n     The mach number is = %.2f",Mach_no);
P_s=56.3*w*(w^(-0.29)-1)*1e6;
printf("\n (c)The pressure downstream from the
   shockwave = %.0f 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

```
clc;
clear;

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

```
f_r_s_m3s=f_r_s/rho_steam; //Flow of Steam
steam_vel=f_r_s_m3s/area; //Steam velocity
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);
meu_steam=0.0113e-3;
meu_water=0.52e-3;
Rel=id*wtr_vel*1000/meu_water;
Reg=id*steam_vel*rho_steam/meu_steam;
printf("\n Reynolds no.(water) = %.2f *10^4",Rel*1e
    -4);
printf("\n Reynolds no.(steam) = %.2f *10^4",Reg*1e
    -4);
// That is, both the gas and liquid are in turbulent
     flow. From the friction chart (Figure 3.7),
    assuming e/d = 0.00015:
// R/(rho*u^2) liq=0.0025     R/(rho*u^2) gas=0.0022
// From equation 3.18:
DPl=4*0.0025*(1000*wtr_vel^2)/id;
DPg=4*0.0022*(rho_steam*steam_vel^2)/id;
X=(DPl/DPg)^0.5;
phi_l=4.35;
phi_g=3.95;
DP_tpf=phi_g^2*DPg;
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

```
clc;
clear;

printf("\n Example 5.2\n");

```

```
M_p_d=0.2e-3; // Mean particle diameter
printf("\n Given:\n Mean particle diameter = %.1f mm",M_p_d*1e3);
f_r_w=0.5; //Flow rate of water
printf("\n Flow rate of water = %.1f kg/s",f_r_w);
id=25e-3; //Diameter of pipe
printf("\n Diameter of pipe = %d mm",id*1e3);
l=100; //length of pipe
printf("\n length of pipe = %d m",l);
t_vel=0.0239; //Terminal velocity of falling sand particles
printf("\n Terminal velocity of falling sand particles = %.4f m/s",t_vel);
//Assuming the mean velocity of the suspension is equal to the water velocity, that is, neglecting slip, then:
Um=f_r_w/(1000*%pi*id^2/4);
printf("\n\n Calculations:\n Mean velocity of suspension = %.2f m/s",Um);
Re=id*Um*1000/0.001;
printf("\n Reynolds no. of water alone = %d",Re);
//Assuming e/d = 0.008, then, from Figure 3.7:
phi=0.0046;
f=0.0092;
//From, equation 3.20, the head loss is:
hf=4*phi*l*Um^2/(9.81*id);
printf("\n Head loss = %.1f m water",hf);
iw=hf/l;
printf("\n Hydraulic gradient = %.3f m water/m",iw);
i=300*1000/(1000*9.81*100);
// Substituting in equation 5.20:
C=(iw/(i-iw)*(1100*9.81*id*(2.6-1)*t_vel)/(Um^2*Um))^-1;
printf("\n C = %.2f",C);
//If G kg/s is the mass flow of sand, then:
G=poly([0],'G');
p=2600^-1*G-0.30*(2600^-1*G+.0005);
printf("\n Mass flow of sand = %.2f kg/s",roots(p));
```

```
printf("")
```

Scilab code Exa 5.3 Calculation of Pressure drop

```
clc;
clear;

printf("\n Example 5.3");

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

```
Ug=flow_air/area;
Us=Ug-(Uo/(0.468+(7.25*(Uo/rho_sand)^0.5)));
printf("\n Air velocity = %.1f m/s",Ug);
printf("\n solids velocity = %.1f m/s",Us);
//Taking
Meu_air=1.7e-5; // viscosity of air
rho_air=1; // Density of air
Re=(d*Ug*rho_air/Meu_air);
printf("\n Reynolds no. of air alone = %d",Re);
phi=0.004;
//Assuming isothermal conditions and incompressible flow, then, in equation 3.18:
DP_air=(4*phi*l/d)*rho_air*Ug^2/2;
printf("\n Pressure drop due to air = %.1f kN/m^2",DP_air*1e-3);
//and in equation 5.38:
DP_x=2805*DP_air/(Uo*Us^2);
printf("\n Pressure drop due to sand particles = %.1f kN/m^2",DP_x*1e-3);
DP=DP_air+DP_x;
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

```
clc;
clear;

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

```
vel_o=flow_o/area_o;//Velocity of water through the orifice
printf("\n Velocity of water through the orifice = %.2f m/s",vel_o);
Re_o=d_o*vel_o*1000/Meu_watr;//Re at the orifice
printf("\n Re at the orifice = %d",Re_o);
//From Figure 6.16, the corresponding value of Cd = 0.61 (diameter ratio = 0.33)
Cd=0.61;
G=flow_o*1e3; //mass flow rate water
//Equation 6.21 may therefore be applied:
ho=poly([0],'ho');
p=G^2-((Cd*area_o*1000)^2*2*9.81*ho);
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

```
clc;
clear;

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

```
printf("\n Coeffecient of discharge = %.2f",Cd);
rho_merc=13550;//Density of mercury
printf("\n Density of mercury = %d kg/m^3",rho_merc)
    ;
rho_watr=1000;//Density of water
printf("\n Density of water = %d kg/m^3",rho_watr);
printf("\n\n Calculations:\n (a)");
area_o=%pi/4*d_o^2;//area of orifice
//The differential pressure is given by:
h_sul=h*(rho_merc-rho_sul)/rho_sul;//
//The mass flow-rate G is given by:
//substituting in equation 6.21 gives the mass
    flowrate as:
G_sul=Cd*area_o*rho_sul*(2*9.81*h_sul)^0.5;
printf("\n The mass flow rate of acid = %.3f kg/s\n
    (b)",G_sul);
DP=rho_sul*9.81*h_sul;
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

```
clc;
clear;

printf("\n Example 6.3\n");
d=150e-3;//Diameter of pipe
printf("\n Given:\n Diameter of pipe = %d mm",d*1e3)
    ;
d_t=50e-3;//Throat diameter
printf("\n Throat diameter = %d mm",d_t*1e3);
hv=121e-3;//Pressure drop over the converging
    section
printf("\n Pressure drop over the converging section
```

```
    = %d mm of water",hv*1e3);
G=2.91; //Mass Flow rate of water
printf("\n Mass Flow rate of water = %.2f kg/s",G);
//From equation 6.32, the mass rate of flow
A1=%pi*d^2/4;
A2=%pi*d_t^2/4;
Cd=G*(A1^2-A2^2)^0.5/(1000*A1*A2*(2*9.81*hv)^0.5);
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

```
clc;
clear;

printf("\n Example 6.4\n");
l=0.3;//length of tube
printf("\n Given:\n length of tube = %.1f m",l);
id_t=25e-3;//Top internal diameter of tube
printf("\n Top internal diameter of tube = %d mm",
    id_t*1e3);
id_b=20e-3;//Bottom internal diameter of tube
printf("\n Bottom internal diameter of tube = %d mm"
    ,id_b*1e3);
d_f=20e-3;//Diameter of float
printf("\n Diameter of float = %d mm",d_f*1e3);
v_f=6e-6;//Volume of float
printf("\n Volume of float = %d cm^3",v_f*1e6);
Cd=0.7;//Coefficient of discharge
printf("\n Coefficient of discharge = %.1f",Cd);
rho=1000;//Density of water
printf("\n Density of water = %d kg/m^3",rho);
rho_f=4800;//Density of float
printf("\n Density of float = %d kg/m^3",rho_f);
```

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

Scilab code Exa 6.5 Calculation of volumetric flow rate of water

```
clc;
clear;

printf("\n Example 6.5\n");

L=0.5;// Length of the weir
printf("\n Given\n Length of the weir = %.1f m",L);
```

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

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

```
clc;
clear;

printf("\n Example 6.6\n");

G=15; //Mass flow rate of organic liquid
printf("\n Given:\n Mass flow rate of organic liquid
     = %d kg/s",G)
L_ow=2;//Length of the weir
printf("\n Length of the weir = %.1f m",L_ow);
rho_l=650;
printf("\n Density of liquid = %d kg/m^3",rho_l);
Q=G/rho_l;
//Use is made of the Francis formula (equation 6.43)
    ,
h_ow=(2/3)*(Q/L_ow)^(2/3);
printf("\n\n Calculations:\n Height of liquid
    flowing over the weir = %.2f mm",h_ow*1e3);
```

Chapter 7

Liquid Mixing

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

```
clc;
clear;

printf("\n Example 7.2\n");

rho_sol=1650;//Density of the solution
printf("\n Given \n Density of the solution = %d kg/
   m^3",rho_sol);
Meu_sol=50e-3;//Viscosity of the solution
printf("\n Viscosity of the solution = %d mN s/m^2",
   Meu_sol*1e3);
Dt=2.28;//Density of the tank
printf("\n Density of the tank = %.2f m",Dt);
D=0.5;//Diameter of the propeller mixer
printf("\n Diameter of the propeller mixer = %.2f m"
   ,D);
H=2.28;//Liquid depth
printf("\n Liquid depth = %.2f m",H);
Za=0.5;//Height of the propeller
printf("\n Height of the propeller = %.1f m",Za);
```

```
N=2;//Rotational speed
//In this problem the geometrical arrangement
   corresponds with the configuration for which the
   curves in Figure 7.6 are applicable.
Re=D^2*N*rho_sol/(Meu_sol);
Fr=N^2*D/9.81;
//From figure 7.6
Np=0.5;
P=Np*rho_sol*N^3*D^5;
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

```
clc;
clear;

printf('Example 7.3\n');

d=0.6;//Tank diameter
N1=4;//Rotor dpeed in Hertz
P1=0.15;//Power consumption
Re1=160000;//Reynold's number
//The correlation of power consumption and Reynolds
   number is given by:
//equation (7.13)
printf("\n For Constant impeller tip speed \n");
D1=d/3;
D2=6*D1;
N2=%pi*N1*D1/(%pi*D2);
printf("\n The new rotor speed = %.2f Hz",N2);
//from eq(1) of the solution
P2=7.32*N2^3*D2^5;
printf("\n The new power required = %.2f kW",P2);
```

```
//For thermal similarity , that is the same
    temperature in both systems:
Re2=Re1*(N2*D2^2/(N1*D1^2));
printf("\n The new reynolds number = %d",Re2);

printf("\n\n For Constant power input per unit
    volume\n");
V2=(%pi/4)*(6*d)^3;
P=V2*0.884;
printf("\n The new power required = %.2f kW",P);
N=(P/(7.32*(6*0.6/3)^5))^(1/3);
printf("\n The new rotor speed = %.2f Hz",N);
Re=Re1*(N*D2^2/(N1*D1^2));
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

```
clc;
clear;
printf("\n Example 8.1\n");

dia_cy=110e-3; //Cylinder diameter
printf("\n Given\n Cylinder diameter = %d mm",dia_cy
    *1e3)
str=230e-3; //stroke
printf("\n Stroke length = %d mm",str*1e3);
l_su=6; //Suction line length
printf("\n Suction line length = %d m",l_su);
d_su=50e-3; //Suction line diameter
printf("\n Suction line diameter = %d mm",d_su*1e3);
lvl_wtr=3; //level of the water in the suction tank
printf("\n level of the water in the suction tank =
    %d m",lvl_wtr);
atm_P=10.36;
printf("\n Atmospheric pressure is equivalent to = %
    .2f m of water",atm_P);
//If the maximum permissible speed of the pump is N
```

```
Hz:
//Angular velocity of the driving mechanism = 2*pi*N radians/s
//Acceleration of piston = 0.5x0.230(2*pi*N)^2*cos(2*pi*N*t) m/s^2
//Maximum acceleration(when t=0) = 4.54*N^2 m/s^2
//Maximum acceleration of the liquid in the suction pipe
//                                      =(.110/.05)^2 x 4.54*N^2 = 21.91*N^2 m/s^2
//Accelerating force acting on the liquid
//                                      = 21.97*N^2*pi/4*(0.050)^2*(6*1000) N
//Pressure drop in suction line due to acceleration
//                                      = 21.97*N^2 *6*1000 N/m^2
//                                      = 1.32x 10^5*N^2 N/m^2
//                                      = 13.44*N^2  m water
//Pressure head at cylinder when separation is about to occur,
x=poly([0],'x');
N=roots(1.20-(10.36-3.0-13.44*x^2));
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

```
clc;
clear;
printf("\n Example 8.2\n");

rho_l=800; //Density of liquid
printf("\n Given\n Density of liquid = %d kg/m^3",rho_l);
```

```
Meu_l=0.5e-3;//Viscosity of liquid
printf("\n Viscosity of liquid = %.1f * 10^-3 N s/m
    ^2",Meu_l*1e3);
Q=0.0004;//Volumetric flow rate
printf("\n Volumetric flow rate = %d m^3/s",Q*1e6);
liq_depth=0.07;
d=25e-3;//Diameter of pipe used
printf("\n Diameter of pipe used = %d",d*1e3);
p_v_r=1e3;//Pressure of vapor in reboiler
printf("\n Pressure of vapor in reboiler = %d kN/m^2
    ",p_v_r*1e-3);
Z=2;//Net Positive Suction Head
printf("\n Net Positive Suction Head = %d m",Z);
A=%pi/4*d^2;//Cross sectional area of pipe
printf("\n\n Calculations:\n Cross sectional area of
     pipe = %.5f m^2",A);
u=Q/A;//Velocity in pipe
printf("\n Velocity in pipe = %.3f m/s",u);
Re=d*u*rho_l/Meu_l;//Reynolds no.
printf("\n Reynolds no. = %d ",Re);
//From Figure 3.7, the friction factor for a smooth
    pipe is:
phi=0.0028;
hf_l=(4*phi*u^2)/(d*9.81);
printf("\n head loss due to friction per unit length
     = %.4f m/m of pipe",hf_l);
//It should be noted that a slightly additional
    height will be required if the kinetic energy at
    the pump inlet cannot be utilised.
//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
l=10;
hf=hf_l*l;
//equation 8.26 becomes:
ho=Z+hf;
printf("\n The minimum height required = %.1f m",ho)
```

```
;
```

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

```
clc;
clear;
printf("\n Example 8.3\n");

Q=0.1;//Flow rate of air suppplied by compressor
printf("\n Given:\n Flow rate of air suppplied by compressor = %.1f m^3/s",Q);
T=273;//Temperature
printf("\n Temperature = %d K",T);
P=101.3e3;//Pressure
printf("\n Pressure = %.1f kN/m^2",P*1e-3);
P2=380e3;//Air compressed to a pressure
printf("\n Air compressed to a pressure = %d kN/m^2",P2*1e-3);
T2=289;//Suction Temperature
printf("\n Suction Temperature = %d K",T2);
l=0.25;//Length of the stroke
printf("\n Length of the stroke = %.2f m",l);
u=4;//Speed
printf("\n Speed = %d Hz",u);
c=4/100;//Cylinder clearance
printf("\n Cylinder clearance = %.2f",c);
Gamma=1.4;
V=Q*T2/(u*T);//Volume per stroke
printf("\n\n Calculations:\n Volume per stroke = %.4f m^3",V);
R=P2/P;//Compression ratio
printf("\n Compression ratio = %.2f",R);
//The swept volume is given by equation 8.42
Vs=V/(1+c-(c*(R)^(1/Gamma)));
```

```
printf("\n The swept volume is = %.4f m^3",Vs);
A=Vs/l;//Cross sectional Area of cylinder
printf("\n Cross sectional Area of cylinder = %.3f m^2",A);
d=(A/%pi*4)^0.5;//Diameter of cylinder
printf("\n Diameter of cylinder = %.2f m",d);
//From equation 8.41, work of compression per cycle
W=P*V*(Gamma/(Gamma-1))*((R)^((Gamma-1)/Gamma)-1);
printf("\n Work of compression per cycle = %.0f J",W);
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 raio of swept volumes in two cylinders

```
clc;
clear;
printf("\n Example 8.4\n");

T=290;//Temperature at which compression takes place
printf("\n Given:\n Temperature at which compression takes place = %d K",T);
P1=101.3e3;//Initial pressure
P2=2065e3;//Final pressure
printf("\n Compressed from a Pressure of %.1f kN/m^2 to %d kN/m^2",P1*1e-3,P2*1e-3);
eta=.85;//Mechanical efficiecy
printf("\n Mechanical efficiecy = %d percent",eta*1e2);
c1=4/100;//Clearance in cylinder 1
printf("\n Clearance in cylinder 1 = %d percent",c1*1e2);
c2=5/100;//Clearance in cylinder 1
printf("\n Clearance in cylinder 2 = %d percent",c2
```

```
*1e2);
R=P2/P1;//Overall compression ratio
printf("\n\n Overall compression ratio = %.1f",R);
V_spe=22.4/28.8*T/273;//Specific volume of air at 290 K
printf("\n Specific volume of air at 290 K = %.3f m^3/kg\n (a)",V_spe);
W=P1*V_spe*2*(1.25/(1.25-1))*(R^.1-1);
//Energy supplied to the compressor, that is the work of compression
W_act=W/0.85;
printf("\n Energy supplied to the compressor, that is the work of compression = %.1f kJ/kg",W_act*1e-3);
printf("\n (b)");
//the work done in isothermal compression of 1 kg of gas
W_it=P1*V_spe*log(R);
//Isothermal efficiency
eta_it=100*W_it/W_act;
printf("\n Isothermal efficiency = %.0f percent",eta_it);
printf("\n (c)");
Gamma=1.4;
//the work done in isentropic compression of 1 kg of gas
W_ie=P1*V_spe*(Gamma/(Gamma-1))*((R)^((Gamma-1)/Gamma)-1);
//Isentropic efficiency
eta_ie=100*W_ie/W_act;
printf("\n Isentropic efficiency = %d percent",eta_ie);
printf("\n (d)")
//From equation 8.47, volume swept out in first cylinder in compression of 1 kg of gas is given by:
Vs1=V_spe/(1+c1-(c1*(R)^(1/(2*2.5))));
//Similarly, the swept volume of the second cylinder
```

```
    is given by:
Vs2=V_spe*(1/R)^0.5/(1+c2-(c2*(R)^(1/(2*2.5))));
ratio=Vs1/Vs2;
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

```
clc;
clear;
printf("\n Example 8.5\n");

Q_l=7.5e-4;
printf("\n Given:\n Volume flow rate of liquid = %.1
   f m^3/s",Q_l*1e4);
rho_l=1200;
printf("\n Density of liquid = %d kg/m^3",rho_l);
h=20;
printf("\n height to which liquid is raised = %d m",
   h);
P=450e3;
printf("\n Air is available at pressure = %d kN/m^2"
   ,P*1e-3);
eta=30/100;
printf("\n Efficiency = %d percent",eta*100);
P_atm=101.3e3;
Gamma=1.4;
G=Q_l*rho_l;//Mass flow of liquid
//Work per unit time done by the pump
W=G*9.81*h;
printf("\n\n Calculations:\n Work per unit time done
    by the pump = %.1f W",W);
//Actual work of expansion of air per unit time
W_act=W/eta;
printf("\n Actual work of expansion of air per unit
```

```
time = %.1f W",W_act);
//Taking the molecular weight of air
M=28.9;
//the specific volume of air at 101.3 kN/m2 and 273
K
va=22.4/M;
//and in equation 8,49:
x=poly([0],'x');
Ga=roots(P_atm*va*x*log(P/P_atm)-W_act);
Q=Ga*va;
printf("\n volume flow rate of air = %.4f m^3/s",Q);
//From equation 8.37
//Power for compression
Power=(P_atm*Q)*(Gamma/(Gamma-1))*((P/P_atm)^((Gamma
-1)/Gamma)-1);
Power_reqd=Power/1000;
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

```
clc;
clear;
printf("\n Example 8.6\n");

P1=101.3e3;
Q_watr=0.01;
printf("\n Given:\n Flow rate of Water = %.2f m^3/s"
,Q_watr);
depth=100;
printf("\n Depth of well = %d m",depth);
d=100e-3;
printf("\n Diameter of pipe = %d mm",d*1e3);
depth_watr=40;
```

```
printf("\n Level of water below water = %d m",
    depth_watr);
Q_air=0.1;
printf("\n Flow rate of Air = %.2f m^3/s",Q_air);
P2=800e3;
Gamma=1.4;
//V1=Q_air;
G_watr=Q_watr*1000;//Mass flow of water
W=G_watr*depth_watr*9.81;
//The energy needed to compress 0. 1 m^3/s of air is
     given by:
E=P1*Q_air*(1.4/0.4)*((P2/P1)^(0.4/1.4)-1);//
    equation 8.37
printf("\n\n Calculations:\n The power required for
    this compression is = %d W",E);
effi=W/E*100;
printf("\n Efficiency = %.1f per cent",effi);
//the mean pressure
P=345e3;
printf("\n The mean pressure = %d kN/m^2",P);
v1=8314*273/(29*P);
printf("\n The specific volume v of air at 273 K and
     given pressure is = %.3f m^3/kg",v1);
v2=8314*273/(29*P1);
printf("\n The specific volume v of air at 273 K and
     101.3 kN/m^2 is = %.3f m^3/kg",v2);
G_air=Q_air/v2; //mass flowrate of the air is:
Q_mean=G_air*v1;//Mean volumetric flowrate of air
Q_tot=Q_watr+Q_mean;//Total volumetric flowrate
A=%pi/4*d^2;//Area of pipe
v_mean=Q_tot/A;
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

```
clc;
clear;
printf("\n Example 8.7\n");

d=40e-3;//Internal Diameter of the pipe
l_p=150; //Lendth of pipe
Q_watr=600e-6;//Flow of water
h1=10; //Vertical Height
h2=2; //head lost across heat exchanger
eta=60/100; //Efficiency of pump

A=%pi/4*d^2; //Area for flow
printf("\n Area for flow = %.4f m^2",A);
u=Q_watr/A; //Velocity
//At 320 K,
Meu=0.65e-3;
rho=1000;
Re=d*u*rho/Meu;
printf("\n Reynolds no. = %d",Re);
phi=0.004; //for a relative roughness of 0.005
l=l_p+h1+(260*d);
printf("\n Equivalent length of pipe = %.1f m",l);
hf=4*phi*l*u^2/(d*9.81);
h_tot=hf+h1+h2;//Total head to be developed
printf("\n Total head to be developed = %.2f m",
    h_tot);
G=Q_watr*rho;// Mass flow of water
P_r=G*h_tot*9.81; //Power Required
P_s=P_r/eta; //Power Supplied
printf("\n Power Required = %.0f W",P_s);
```

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

```
clc;
```

```
clear;
printf("\n Example 8.8\n");

eta=0.50;
Q=[0.0028 0.0039 0.0050 0.0056 0.0059]
h=[23.2 21.3 18.9 15.2 11.0]
plot(Q,h,'o-');
//The head to be developed , h=10+4.12*u^2 m water
//h=10+2.205e5*Q^2
Q1=0.0015:0.0001:0.0060
h1=10+2.205e5*Q1^2;
plot2d(Q1,h1,style=1);
xtitle("Data for Example 8.8","Discharge (Q m^3/s)",
    "Head (m water)");
legend("Pump characteristics","h=10+2.205e5*Q^2");
//showing the intersection point
x1=[0 0.0054];
y1=[16.43 16.43];
x2=[0.0054 0.0054];
y2=[0 16.43];
plot(x1,y1,x2,y2);
Q_r=0.0054;
printf("\n The discharge at the point of
    intersection between\n the purnp characteristic
    equation = %.4f m^3/s",Q_r);
h_r=10+2.205e5*Q_r^2;
printf("\n The total head developed = %.2f m",h_r);
P=Q_r*1000*h_r*9.81/eta;
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

```
clc;
```

```
clear;
printf("\n Example 8.10\n");

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

```
clc;
clear;

printf("\n Example 9.1\n");

M_dot1=20; //rate of mass to be cooled
M_dot2=25; //rate of cooling water
Cp=4.18e3; //Heat capacity
T1=360; //Initial temp.
T2=340; //Final temp.
theta_1=300; //Temperature of cooing water entering
U=2e3; //Overall heat transfer coefficient

Q=M_dot1*Cp*(T1-T2); //Heat load
printf("\n Heat load = %.0f kW",Q*1e-3);
//The cooling water outlet temperature is given by
x=poly([0],'x');
theta_2=roots(Q-(M_dot2*Cp*(x-300)));
printf("\n The cooling water outlet temperature is =
    %.0f K",theta_2);
```

```
printf("\n (a) Counter flow")
//In equation 9.9:
theta_m1=((T1-theta_2)-(T2-theta_1))/(log((T1-theta_2)/(T2-theta_1)));
A1=Q/(U*theta_m1)
printf("\n The surface area required %.2f m^2",A1);
printf("\n (b) Co-current flow")
//In equation 9.9:
theta_m2=((T1-theta_1)-(T2-theta_2))/(log((T1-theta_1)/(T2-theta_2)));
A2=Q/(U*theta_m2)
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

```
clc
clear;

printf("\n Example 9.2\n");
dx=0.5; //Thickness of wall
T1=400; //Temperartue of inner surface
T2=300; //Temperature of outer surface
K=0.7; //Thermal conductivity
A=1; //Area of heat transfer
//From equation 9.12:
Q=K*A*(T1-T2)/dx;
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

```
clc;
clear;

printf("\n Example 9.3\n");

dx1=0.20; //thickness of firebrick
dx2=0.10; //thickness of insulating brick
dx3=0.20; //thickness of building brick
k1=1.4; //Thermal conductivity of firebrick
k2=0.21; //Thermal conductivity of insulating brick
k3=0.7; //Thermal conductivity of building brick
T1=1200; //Temperature at junction 1
T4=330; //Temperature at junction 4

//From equation 9.19:
Q=(T1-T4)/((dx1/k1)+(dx2/k2)+(dx3/k3));
printf("\n Heat loss per unit area = %d W/m^2",Q);
//The ratio (Temperature drop over firebrick)/(Total
    temperature drop)
R=(dx1/k1)/((dx1/k1)+(dx2/k2)+(dx3/k3));
//Temperature drop over firebrick
dT=(T1-T4)*R;
printf("\n Temperature drop over firebrick = %.0f K"
    ,dT);
T2=(T1-dT);
printf("\n The temperature at the firebrick-
    insulating brick interface = %.0f K",T2);
```

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

```
clc;
clear;

printf("\n Example 9.4\n");
```

```
T=295; //initial temperature of surfaces
T2f=375; //Final temperature of far surface
dT1=900; //Temperature of near face raised
//The temperature at any distance x from the near face at time t is given by equation 9.37
//Choosing the temperature scale so that the initial temperature is everywhere zero, then:
R=(T2f-T)/(2*(dT1-T)); //ratio of theta to twice of theta dash

//An approximate solution is obtained by taking the first term only, to give:
//    R=erfc(346*t^-0.5)
//erfc(1.30)=R
//solving above equation
x=poly([0],'x');
t=roots((1.30^2*x)-346^2);
printf("\n Time taken to rise from 295 to 375 K = %.1f h",t/3600);
```

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

```
clc;
clear;

printf("\n Example 9.5\n");

T=295; //initial temperature of surfaces
T2f=375; //Final temperature of far surface
dT1=900; //Temperature of near face raised
DH=4.2e-7; //Thermal diffusivity
//The development of the temperature profile is shown in Figure 9.12
//The problem will be solved by taking relatively
```

```
   large intervals for dx.
//Choosing dx = 50 mm, the construction shown in
   Figure 9.12
dx=50e-3;
//Because the second face is perfectly insulated,
   the temperature gradient must
// be zero at this point.
//It is seen that the temperature is
//less than 375 K after time 23dt and greater than
   375 K after time 25dt
//Thus:
//t=24*dt
//from equation 9.43
dt=dx^2/(2*DH);
t=24*dt;
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

```
clc;
clear;

printf("\n Example 9.6\n");

d=25e-3; //Diameter of copper sphere
l=25e-3; //Side length of a copper cube
h=75; //External heat transfer coefficient
rho_cu=8950; //Density of copper at mean temperature
Cp=0.38e3; //Heat capacity of copper at mean
   temperature
k=385; // Thermal conductivity of copper at mean
   temperature
Tf=923; //Temperature of the furnace
```

```
Ta=368; //Temperature at which they are annealed
t=5*60; // time taken

V_Ae_S=(d/6); //V/Ae tor the sphere
printf("\n V/Ae tor the sphere = %.2f * 10^-3 m",
    V_Ae_S*1e3);
V_Ae_C=(l/6); //V/Ae tor the cube
printf("\n V/Ae tor the cube = %.2f * 10^-3 m",
    V_Ae_C*1e3);
Bi=h*(V_Ae_S)/k;
//The use of a lumped capacity method is therefore
    justified
tao=rho_cu*Cp*V_Ae_S/h;
//Then using equation 9.49
//theta=T
x=poly([0],'x');
T=roots(((x-Ta)/(Tf-Ta))-%e^(-t/tao));
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

```
clc;
clear;

printf("\n Example 9.7\n");

k=2.5; //Thermal conductivity
DH=2e-7;//Thermal diffusivity of the surrounding
    fluid
h=100; //External heat transfer coefficient
To=293; //Initial Temperature
T_dash=373; //Oven Temperture
Tc=353; //temperature throughout the whole of the
```

```
    sheet reaches a minimum
l=10e-3; //thickness of sheet
L=l/2;

//For the given process, the Biot number
Bi=h*L/k;
Bi_1=1/Bi;
lim_val=(T_dash-Tc)/(T_dash-To);
//From Figure 9.17, the Fourier number
Fo=7.7;
t=Fo*L^2/DH
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

```
clc;
clear;

printf("\n Example 9.8\n");

l=5;//Length of the channel of uranium reactor
Q=.25e6; //Heat release from uranium reactor
k=33; //Thermal conductivity of the uranium
Q_m=Q/l; //Heat release rate
//Thus, from equation 9.52:
dT=Q_m/(4*%pi*k);
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

```
clc;
clear;

printf("\n Example 9.9\n");

Cp=2380;//specific heat capacity of nitrobenzene
k=0.15;
Meu=0.70e-3;//Viscosity of nitrobenzene
d_i=15e-3;//internal diameter of tube
d_o=19e-3;//external diameter of the tube
d_s=0.44;//shell diameter
b_s=0.150;//baffle spacing
p=0.025;//pitch
c=0.006;//clearance
//(i)Tube side coefficient
h_i=1000;//based on inside area
h_io=1000*d_i/d_o;//based on outside area
//(ii) Shell side coefficient.
A=d_s*b_s*c/p;//Area for flow
G_s_=4/A;
//Taking Meu/Meu_s=1 in equation 9.91
d_e=4*((25e-3^2-(%pi*d_o^2/4))/(%pi/d_o));
h_o=0.36*k/d_e*(d_e*G_s_/Meu)^0.55*(Cp*Meu/k)^0.33;
//(iii) Overall coefficient
//The logarithmic mean temperature difference is
   given by:
Tm=(((400-345)-(315-305))/log((400-345)/(315-305)));
//The corrected mean temperature difference is
Tm_c=Tm*0.8;
Q=4*Cp*(400-315);
//The surface area of each tube
A_t=0.0598;
U_o=Q/(2*166*5*A_t*Tm_c);
//(iv) Scale resistance.
R_d=(1/U_o)-(1/750)-(1/1000);
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

```
clc;
clear;

printf("\n Example 9.10\n");

G=15;//Mass flow rate of benzene
d_s=1; //Internal diameter of Heat Exchanger
l=5; //Length of tubes
od=19e-3; //Outer diameter of tubes
C=6e-3; //Clearance
l_b=0.25; //Baffle spacing
Meu=.5e-3;
Y=25e-3; //dimension of square pitch
N=19; //no. of Baffles

As=d_s*l_b*C/Y; //Cross-flow area
printf("\n Cross-flow area = %.2f m^2",As);
G_dash_s=G/As; //Mass flow
printf("\n Mass flow = %d kg/m^2 s",G_dash_s);
d_e=4*(Y^2-(%pi*od^2/4))/(%pi*od);//Equivalent
    Diameter
printf("\n Equivalent Diameter = %.4f m",d_e);
Re=G_dash_s*d_e/Meu;
//From Figure 9.29:
f_dash=.280;
rho_b=881;//density of benzene
DPf=f_dash*G_dash_s^2*(N+1)*d_s/(2*rho_b*d_e);
printf("\n The pressure drop over the tube bundle =
    %.0f N/m^2",DPf);
printf("\n\t\t\t\t\t= %.0f m of Benzene",DPf/(rho_b
    *9.81));
```

Scilab code Exa 9.11 Calculation of heat transfer coefficient

```
clc;
clear;

printf("\n Example 9.11\n");

d=0.15; //Diameter of pipe
Ts=400; //Surface temperature
Ta=294; //Air temperture
//Over a wide range of temperature, k^4*(beta*g*rho
    ^2*Cp/(Meu*k)) = 36.0
//For air at a mean temperature i.e. 347 K
k=0.0310; //Thermal conductivity ---Table 6,
    Appendix A1
//X=beta*g*rho^2*Cp/(Meu*k)
X=36/k^4;
//From Equation 9.102:
GrPr=X*(Ts-Ta)*d^3;
//From Table 9.5:
n=0.25;
C_dd=1.32;
//Thus, in Equation 9.104:
h=C_dd*(Ts-Ta)^n*d^(3*n-1);
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

```
clc;
clear;
```

```

printf("\n Example 9.12\n");

lambda=1e-6;//Wavelength
E_l_b=1e9; //Emissive power at given lambda

//From equation 9.108
C2=1.439e-2;
C1=3.742e-16;
T=C2/lambda/log(C1/(E_l_b*lambda^5));
printf("\n The temperature of surface = %d K",T);
//With an error of +2 per cent, the correct value is
    given by:
E_l_b_n=(100-2)*E_l_b/100;
//In equation 9.108:
T_n=C2/lambda/log(C1/(E_l_b_n*lambda^5));
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

```
clc;
clear;

printf("\n Example 9.13\n");

d=10e-3; //Diameter of carbide elements
l=0.5; //Length of carbide elements
Ts=1750; //Maximun surface temperature of carbide
P=500e3; //Thermal power output required
sigma=5.67e-8;

//From equation 9.112, the total emissive power is
    given by:
Eb=sigma*Ts^4;
```

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

Scilab code Exa 9.14 Calculation of emissivity of a grey surface

```
clc;
clear;

printf("\n Example 9.14\n");

A=10; //Area of the surface
P_r=1000e3; //Power radiated
T1=1500; //First Temperature
T2=1600; //Second Temperatue
sigma=5.67e-8;

E=P_r/A; //The emissive Power
printf("\n The emissive Power when T=1500 K = %d kW"
   ,E*1e-3);
//From equation 9,118:
e=E/(sigma*T1^4);
printf("\n Emissivity when T=1500 K = %.3f",e);
E2=e*sigma*T2^4;
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

```
clc;
clear;

printf("\n Example 9.15\n");

A1=2; //Area of rectangle(Surface 1)
A2=%pi*1^2/4; //Area of disc (Surface 2)
T1=1500; //Temperature of Surface 1
T2=750; //Temperature of Surface 2
F12=0.25; //View factor
sigma=5.67e-8;
//From equation 9. 1 26:
F21=A1*F12/A2;
printf("\n View factor, F12 = %.3f",F21);
Q12=sigma*A1*F12*(T1^4-T2^4);
printf("\n The net radiation transfer = %.0f kW",Q12
    *1e-3);
```

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

```
clc;
clear;

printf("\n Example 9.16\n");

printf("\n (a)")
//Using the nomenclature in Figure 9.40 iii;
```

```
X=4;//width of horizontal plate and length vertical
    plate
Y=6;//length of horizontal plate
Z=3;//height of verical plate
W=Y/X;
H=Z/X;
A1=Z*X;//Area of plate 1
A2=X*Y;//Area of plate 2
F12=0.12;
printf("\n View Factor, F12= %.2f",F12);
//From equation 9.126:
F21=A1*F12/A2;
printf("\n View Factor, F21= %.2f",F21);
printf("\n (b)");
//For the two spheres
r1=1; //Diameter of sphere 1
r2=2; //Diameter of sphere 2
F12b=1;
F21b=(r1/r2)^2;
printf("\n View Factor, F21= %.2f",F21b);
F22b=1-F21b;
printf("\n View Factor, F22= %.2f",F22b);
```

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

```
clc;
clear;

printf("\n Example 9.17\n");

ri_u=0.2;// Inner radius of the upper ring
ro_u=0.3;// Outer radius of the upper ring
ri_l=0.3;// Inner radius of the lower ring
ro_l=0.4;// Outer radius of the lower ring
//F23 = ((A12/A2)*(F12_34))-F12_4-((A1/A2)*(F1_34 -
```

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

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

```
clc;
clear;

printf("\n Example 9.18\n");

d=1; //Diameter of plate
r1=0.5;
r4=r1; //Radius of the imaginary disc sealing the
hemisphere
L=r1; //The distance between the plate and the
bottom of the dome

A1=%pi*d^2/4; //Area of the plate
A2=2*%pi*d^2/4; //Area of the underside of the
Hemisphere
A4=%pi*r4^2/4;//Area of an imaginary disc sealing
the hemisphere and parallel
            //to the plate
T1=750;//Temperature of the plate
T2=1200;//Temperature of hemispherical cone
```

```
T3=290;//Temperature of the surroundings
sigma=5.67e-8;
//from equation 9.134, the net radiation to the
    surface of the plate 1 is
//given by:
//Q1=sigma*A2*F21*(T2^4-T1^4)+sigma*A3*F31(T3^4-T1
    ^4)
//using the reciprocity rule:
//Q1=sigma*A1*F12*(T2^4-T1^4)+sigma*A3*F31(T3^4-T1
    ^4)
//All radiation from the disc 1 to the dome 2 is
    intercepted by the imaginary
// disc 4 and hence F\2 = F\4, which may be obtained
     from Figure 9.39ii, with
//i and j representing areas 1 and 4 respectively
R1=r1/L;
R4=r4/L;
S=1+(1+R4^2)/(R1^2);
F14=0.5*(S-(S^2-4*(r4/r1)^2)^0.5);
F12=F14;
//The summation rule states that
//F11+F12+F13=1
//F11=0
F13=1-F12;
Q1=sigma*A1*F12*(T2^4-T1^4)+sigma*A1*F13*(T3^4-T1^4)
    ;
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

```
clc;
clear;

printf("\n Example 9.19\n");
```

```
d=2; //Diameter of the cylinder
h=1; //Depth of insulated cylinder
A1=%pi*d^2/4; //Radiant heater surface
A2=A1; //Under-Surface of the vessel
A_R=%pi*d*h;
T1=1500;
T2=373;
//From Figure 9.40ii, with i = 1, j = 2
r1=1;
r2=1;
L=1;
//The view factor may also be obtained from Figure 9.39ii as follows:
//Using the nomenclature of Figure 9.39
R1=r1/L;
R2=r2/L;
S=1+(1+R2^2)/(R1^2);
F12=0.5*(S-(S^2-4*(r2/r1)^2)^0.5);
sigma=5.67e-8;
//Using the summation rule
//F11=0
F1R=1-F12;;
F2R=F1R;
Q2=(A1*F12+((1/(A1*F1R)+(1/(A2*F2R))))^-1)*sigma*(T1^4-T2^4);
printf("\n The rate of radiant heat transfer to the vessel = %d kW",Q2*1e-3);
//If the surroundings without insulation are surface 3 at
T3=290;
F23=F2R;
//from equation 9.135
Q2_d=sigma*A1*F12*(T1^4-T2^4)+sigma*A2*F23*(T3^4-T2^4);
printf("\n The rate of radiant heat transfer to the vessel\n if the insulation were removed = %.0f kW",Q2_d*1e-3);
red=(Q2-Q2_d)/Q2*100; //Percentage Reduction
```

```
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

```
clc;
clear;

printf("\n Example 9.20\n");

e=0.75; //Emissivity of grey surface
r=1-e; //reflectivity of surface
Ts=400; //Temperature of surface
T_amb=295;
sigma=5.67e-8;
q1=3e3; //Rate of radiation arriving at grey surface
//From equation 9.118
Eb=sigma*Ts^4;
printf("\n Emissive Power = %.0f W/m^2",Eb);
//From equation 9.138
qo=e*Eb+r*q1;
printf("\n Radiosity = %.0f W/m^2",qo);
//From equation 9.140
Q_A=e/r*(Eb-qo);
q=Q_A;
printf("\n The net rate of radiation trasfer = %.0f
    W/m^2",q);
printf("\n where the negative value indicates heat
    transfer to the surface.");
//For convective heat transfer from the surface
qc=-1*q;
hc=qc/(Ts-T_amb);
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

```
clc;
clear;

printf("\n Example 9.21\n");

sigma=5.67e-8;
T=[1000 500 300];//tempertaure of surfaces
//Taking surface 1 as the heater , surface 2 as the
    heated plate and surface 3
//as an imaginary enclosure
A=[1.07 1.07 0.628];//Array of area of surfaces
e=[0.75 0.50 1.0];//Array of emissivity of the
    surfaces
r=[0.250 0.50];// Array of radius of two surfaces
//X is ratio of area to radius(A/r)
//Y = A*e/r
L=0.2; //distance between two discs
for i=1:2
    X(i)=A(i)/r(i);
    Y(i)=A(i)*e(i)/r(i);
    R(i)=r(i)/L;
end

F11=0;
F22=0;
S=1+(1+R(2)^2)/(R(2)^2);
F12=0.5*(S-(S^2-4*(r(2)/(2*r(1)))^2)^0.5);
A1_F11=0;
A2_F22=0;
A1_F12=A(1)*F12;
A1_F13=A(1)-(A(1)*F11+A(2)*F12);
```

```
//for surface 2:
A2_F21=A1_F12;
A2_F23=A1_F13;
//for surface 3:
//By reciprocity rule
A3_F31=A1_F13;
A3_F32=A2_F23;
A3_F33=A(3)-(A3_F31+A3_F32);

//From equation 9.112:
for i=1:3
    E_b(i)=sigma*T(i)^4/1000;
end

//Since surface 3 is a black body
q_o3=E_b(3);
//From equations 9.157 and 9.158:
//we get

function [f]=F(x)
    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);
    f(2)=(A1_F12*x(1))+((A2_F22-A(2)/r(2))*x(2))+E_b
        (2)*A(2)*e(2)/r(2);
    funcprot(0);
endfunction

x=[0 0];
q_o=fsolve(x,F);

//From equation 9.140:
Q1=(A(1)*e(1)/r(1))*(E_b(1)-q_o(1));
Q2=(A(2)*e(2)/r(2))*(E_b(2)-q_o(2));
printf("\n Power input to the heater = %.1f kW",Q1);
printf("\n The rate of heat transfer to the plate =
    %.2f kW",Q2);
printf("\n where the negative sign indicates heat
    transfer to the plate")
```

Scilab code Exa 9.22 Calculation of net radiation to the walls

```
clc;
clear;

printf("\n Example 9.22\n");

d=0.5;//diameter of chamber
l=2;//Length of chamber
e=0.5;//Emissivity
T_s=750;//Temperature at which the chamber is
    maintained
P=150e3;
T_g=1250;
sigma=5.67e-8;

//The partial pressures of carbon dioxide (P_c) and
    of water (P_w) are:
P_c=0.1*P;
P_w=P_c;
//From Table 9.7:
V=%pi/4*d^2*l;//Volume of the chamber
A_s=(2*%pi/4*d^2)+(%pi*d*l);//total surface are of
    chamber

L_e=3.6*(V/A_s);
//FOR WATER VAPOUR
//and from Figure 9.44, e_w = 0.075
//Since P_w*L_e = 0.06 bar m, then from Figure 9.44:
C_w=1.4;
e_w1=C_w*0.075;
//FOR CARBON DIOXIDE
//Since P = 1.5 bar, Pc = 0.15 bar and P_c*L_e =
    0.06 bar m, then, from
```

```
//Figure 9.38:
//and from Figure 9.45, e_c = 0.037
C_c1=1.2;
e_c1=(C_c1*0.037);
A=(P_w+P_c)*L_e;
B=P_c/(P_c+P_w)
//Thus, from Figure 9.45 for T_g > 1203 K, De = 0.001
//and, from equation 9.160:
De=0.001;
e_g=e_w1+0.044-De;

//FOR WATER VAPOUR
//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,
//then, from Figure 9.44:C_w=1.4
e_w2=(0.12*C_w);
//and the absorptivity, from equation 9.163 is:
a_w=e_w2*(T_g/T_s)^0.65;
//FOR CARBON DIOXIDE
//From Figure 9.45 at 750 K, e_c=0.08
//From Figure 9.45 at P=1.5 bar and P_c*L_e*(Ts/Tg)= 0.036 bar m:
C_c2=1.02;
e_c2=(0.08*C_c2);
//and the absorptivity, from equation 9.164 is:
a_c=e_c2*(T_g/T_s)^0.65;
//P_w/(P_c+Pw)=0.5 and (P_c+P_w)*L_e*(T_s/T_g) =(0.036+0.036)=0.072 bar m
//Thus, from Figure 9.46, for Tg=813 K, De=Da <0.01 and this may be neglected
a_g=a_w+a_c;
//If the surrounding surface is black, then:
Q=sigma*A_s*(e_g*T_g^4-a_g*T_s^4);
printf("\n Radiation to the walls if the surface is black = %.1f kW",Q*1e-3);
//For grey walls, the correction factor allowing for multiple reflection of
```

```
//incident radiation is:
C_g=0.5/(1-(1-0.326)*(1-0.5));
Q_w=(Q*C_g);
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

```
clear;
clc;
printf("\t Example 9.23\n");
function[n]=mole(w,m)
    n = w/m;
    funcprot(0);
endfunction

function[p]=partial(n1)
    p = 308*(n1/total);
    funcprot(0);
endfunction

w_steam = 0.57;       //mass flow rate of steam entering in [kg/sec]
w_CO2 = 0.20;         //mass flow rate of CO2 entering in [kg/sec]
m_water = 18;         //molecular mass of water in kg
m_CO2 = 44;           //molecular mass of CO2 in kg
n_steam = mole(w_steam,m_water);      //number of moles in kmol
n_CO2 = mole(w_CO2,m_CO2);            //number of moles in kmol
printf("\n At the entrance there is %.3f kmol steam",n_steam);
printf("\n At the entrance there is %.4f kmol water"
```

```
    ,n_CO2);
total = n_steam + n_CO2;
printf("\n Total number of moles fed to the
    condenser per second = %.4f kmol",total);
p_steam = partial(n_steam);
p_CO2 = partial(n_CO2);
printf("\n At the inlet partial pressure is %d kN/m
    ^2 watre",p_steam);
printf("\n At the inlet there is %d kN/m^2 CO2",
    p_CO2);
printf("\n From the Table 11A in the appendix Dew
    point = %d K",404);
mean_mol = (0.57 + 0.20)/total;      //mean molecular
     weight of the mixture in kg/kmol

outlet_steam = 11.7;              //partial pressure
    of water in kN/m^2
outlet_CO2 = 308 - outlet_steam; //partial pressure
    of water in kN/m^2
printf("\n At the outlet partial pressure is %.1f kN
    /m^2 water",outlet_steam);
printf("\n At the outlet there is %.1f kN/m^2 CO2",
    outlet_CO2);
n_s = n_CO2*outlet_steam/outlet_CO2;
steam_condensed = n_steam - n_s;
printf("\n steam codensed = %.5f kmol",
    steam_condensed);

printf("\n\n For the interval 404 to 401 K");
p_steam_401K = 252.2;    //[kN/m^2]
p_CO2_401K = 308 - 252.2;//[kN/m^2]
steam_remaining = 0.0045*p_steam_401K/p_CO2_401K;
s_c = n_steam - steam_remaining;    //[kmol]
Heat_cond = s_c*18*(2180 + 1.93*(404-401));
    //[kW]
Heat_uncondensed_steam = 0.0203*18*1.93*(404-401);
    //[kW]
Heat_CO2 = 0.020*0.92*(404-401);
```

```
total_heat = Heat_cond + Heat_uncondensed_steam +
    Heat_CO2;
printf("\n Heat of condensation = %d kW",Heat_cond);
printf("\n Heat of uncondensed steam = %.1f kW",
    Heat_uncondensed_steam);
printf("\n Heat from CO2 = %.1f kW",Heat_CO2);
printf("\n Total = %.1f kW",total_heat);

printf("\n\n For other intervals simiarly");
printf("\n Interval(K)          Heat Load(kW)");
printf("\n  404 - 401              %.1f        ",
    total_heat);
printf("\n  401 - 397              %.1f        ",323.5)
    ;
printf("\n  397 - 380              %.1f        ",343.5)
    ;
printf("\n  380 - 339              %.1f        ",220.1)
    ;
printf("\n  339 - 322              %.1f        ",57.9);
printf("\n  total                  %.1f        ",
    total_heat+323.5+343.5+220.1+57.9);
flow_water = 1407.3/(4.187*(319-300));      //[kg/
    sec]
hi = 6.36;        //[kW/m^2 K]Based on flow velocity
    of 1425 kg/m^2 sec
ho = 5.25;        //[kW/m^2 K]Based on outside area
Cp = (0.20*0.92 + 0.57*1.93)/0.77;  //[kJ/kg K]
printf("\n Mean specific heat, Cp = %.3f kJ/kg K",Cp
    );
k_mean = 0.025;    //[kW/m K]
a = 0.0411;        //[m^2]
mass_velocity = (0.20+0.57)/0.0411;      //[kg/m^2
    sec]
printf("\n Mass velocity = %.1f kg/m^2 sec",
    mass_velocity);
hg = 107;          //[W/m^2 K] at Re = 29,800 at
    equivalent diameter = 0.024m
u_pD = 0.62;       //(u/pD)^0.67 = 0.62
```

```
Cpu_k = 1.01;       //(Cp*u/k)^0.67
Psf = (122.6 - 38)/log(122.6/38);
kG = hg*(Cpu_k)/(1000*Cp*Psf*u_pD);
printf("\n\n kG = %.4f",kG);

printf("\n Point   Ts   Tc   UT    UTow   Q       A = Q/(UT)      T     Tow     (Q/T)ow");
printf("\n  1      404   378  309  -       -           -       84.4  -        -       ");
printf("\n  2      401   356  228   268.5  468.4     1.75       88.1  86.3       5.42");
printf("\n  3      397   336  145   186.5  323.5     1.74       88.6  8.4        3.66");
printf("\n  4      380   312  40.6  88.1   343.5     3.89       76.7  82.7       4.15");
printf("\n  5      339   302  5.4   17.5   220.1     12.58      38.1  55.2       4.00");
printf("\n  6      322   300  2.1   3.5    51.9      14.83      22.2  29.6       1.75");

printf("\n Assuming no scale resistance, the overall coefficient = %.3f W/m^ K",1407.3/(34.8*74.2));
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);
printf("\n Actual coeficient = %.3f kW/m^2 K",1407.3/(53.9*74.2));
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 coeffecient

```
clc;
clear;

```

```
printf("\n Example 9.24");

d_v=1;//diameter of the vessel
L=0.3;//diameter of propeller agitator
N=2.5;//rotating speed of propeller agitator
T=310;//Temperature
G=0.5;//circulation speed of cooling water
d_o=25e-3;//outer diameter of stainless steel coil
d=22e-3;//inner diameter of stainless steel coil
d_w=(d_o+d)/2;
d_c=0.8;//diameter of helix
T_m=290;//mean temperature
k1=0.59;
Meu1=1.08e-3;
C_p1=4.18e3;
x_w=1.5e-3;

//From equations 9.202 and 9.203, the inside film coefficient for the water
//is given by:
h_i=(k1/d)*(1+3.5*(d/d_c))*0.023*(d*1315/Meu1)^0.8*(C_p1*Meu1/k1)^0.4;
//The external film coefficient is given by equation 9.204:
C_p2=1.88e3;//Specefic heat capacity
Meu2=6.5e-3;//viscosity
k2=0.40;
rho=1666;
Meu_s=8.6e-3;
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;

k_w=15.9;
R_o=0.0004;
R_i=0.0002;
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;
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

```
clc;
clear;

printf("\n Example 9.25")

C_p=4e3;

//If T K is the temperature of the liquid at time /
   s, then a heat balance on
//the vessel gives:
x=poly([0],'x');
T_max=roots((600*0.5)*(393-x)-(10*6)*(x-293));
printf("\n\n Maximum temperature to which it can be
   heated = %.1f K",T_max)
//solving the equation finally we get

t1=integrate('11111*(1/(376.3-T))','T',293,353);
printf("\n Time taken to heat the liquid from 293 K
   to 353 K = %.0f s",t1);
//The steam is turned off for 7200 s and during this
    time a heat balance gives:
//on solving as given in book we get
T=346.9;
//The time taken to reheat the liquid to 353 K is
   then given by:
t2=integrate('11111*(1/(376.3-T))','T',346.9,353);
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

```
clc;
clear;

printf("\n Example 9.26\n");

//As in Example 9.1, the heat load = 1672 kW
Q=1672;
//With reference to Figure 9.71:
T1=360;
T2=340;
theta1=300;//Temperature of cooling water entering
theta2=316;
X=(theta2-theta1)/(T1-theta1);
Y=(T1-T2)/(theta2-theta1);
//from Figure 9.58
F=0.97;
theta_m=41.9;
//and hence:
A=Q/(2*F*theta_m);//the heat transfer area
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

```
clc;
clear;

printf("\n Example 9.27\n");

//As in Example 9.1, the heat load = 1672 kW
Q=1672;
//With reference to Figure 9.71:
```

```
T1=360;
T2=340;
theta1=300;//Temperature of cooling water entering
theta2=316;
F_theta_m=40.6;//corrected mean temperature
    difference
T=(T1+T2)/2;
d=1.9e-3;//Tube diameter
u=1;//Water velocity
//then, in equation 9,221:
h_i=4.28*(0.00488*T-1)*u^0.8/d^0.2;
//From Table 9.18, an estimate of the shell-side
    film coefficient is:
h_o=(1700+11000)/2000;
//For steel tubes of a wall thickness of 1.6 mm, the
     thermal resistance of the wall, from Table 9.15
    is:
xw_kw=0.025;
//the thermal resistance for treated water, from
    Table 9.16, is 0.26 m2K/kW
Ri=0.26;
Ro=Ri;
U=((1/h_o)+xw_kw+Ri+Ro+(1/h_i))^-1;
A=Q/(F_theta_m*U);
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

```
clear;
clc;

printf('Example 9.28');
//Using Kern's method design Shell n tube heat
    exchanger
```

```

mh = 30;              //[kg/s] Hot fluid flow rate
Thi = 370;           //[K] Hot Fluid Inlet Temperature
Tho = 315;           //[K] Hot Fluid outlet Temperature
Tci = 300;           //[K] Cold Fluid Inlet Temperature
Tco = 315;           //[K] Cold Fluid Outlet Temperature
cpc = 4.18*10^3;         //[J/kg.K] Thermal Conductivity of Cold Fluid
//From table A1.3 at mean temperature 343 K
cph = 2.9*10^3;              //[J/kg.K] Thermal Capacity of Hot fluid

q = mh*cph*(Thi-Tho);         //[kW] Heat load
mc = q/(cpc*(Tco-Tci));       //[kg/s] Flow of cooling water

Tln = ((Thi-Tho)-(Tco-Tci))/(log((Thi-Tho)/(Tco-Tci)));    //[K] Logarithm mean temperature difference

//for one-shell side pass and two-tube side pass Equation 9.213
X = (Thi - Tho)/(Tco-Tci);
Y = (Tco-Tci)/(Thi - Tci);

//From Figure 9.757
F = .85;
//From Table 9.17
U = 500         //[W/m^2.K]
A = q/(F*Tln * U);
//Thus COnvenient tubes to bo used
od = .02         //[m] outer dia
id = .016        //[m] inner dia
l = 4.83         //[m] effective tube length

```

```
s = %pi*od*l;
N = A/s;
//From equation 9.211
db = (1210/.249)^(2.207)^-1*20/1000;              //[m]
//From figure 9.71
dc = .068          //[m] diametric clearance between shell and tubes
ds = db+dc                //[m] Shell dia

//Tube-Side Coefficient
//From equation 9.218
Ac = %pi/4*id^2               //[m^2] Cross sectional area
Ntp = N/2;

Af = N/2*Ac          //[m^2] Tube side flow area
mw = 76.3/Af              //[kg/m^2.s] Mass velocity of water
rho = 995          //[kg/m^3] mas density of water
u = mw/rho                //[m/s] water velocity

//At mean temperature 308 K
vu = .8*10^-3                  //[N.s/m^2] viscosity
k = .59                    //[W/m.K]

Re = id*u*rho/vu;
Pr = cpc*vu/k;
ld = l/id;
//from figure 9.77
jh = 3.7*10^-3
hi = jh*Re*Pr^.3334*.59/id;              //[W/m^2.K]

//Shell-Side Coefficient
//Baffle packing will be taken as 20percent of shell dia
dbf = .20*ds;          //[m] Baffle Dia
tb = 1.25*20*10^-3          //[mm] Tube Pitch
```

```
//From equation 9.226
As = (25-20)/25*10^3*(ds*Ac)                    //[m^2]
Gs = 30/As;                                //[kg/m^2.s]
//From Equation 9.228
de = 1.1*(.025^2-.917*od^2)/od;                   //[m]
//At mean temperature 343 K Butyl Alcohol
rho2 = 780              //[kg/m^3] density
vu2 = .8*10^-3              //[N.s/m^2] viscosity
Cp2 = 3.1*10^3                  //[J/kg.K] Heat
    capacity
k2 = .16                    //[W/m.K]
//Equation 9.229
Re2 = Gs*de/vu2;
Pr2 = Cp2*vu2/k2;
//From figure 9.81
jh2 = 5*10^-3;
//Equation 9.230
hs = jh2*Re2*Pr2^.334*k2/de;

//Overall Coefficient
//from Table 9.16
k3 = 50                    //[W/m.K]          Thermal
    Conductivity
Rw = .00020                //[m^2.K/W]        Scale
    Resistances
Ro = .00015                //[m^2.K/W]        Resistance
    for organic

U = [1/hs + Rw + .5*(od-id)/k3 + Ro*od/id+od/(id*hi)
    ]^-1

//From figure 9.78
jf = 4.5*10^-3;
n = 2;
delP = n*[4*jf*(4.830/id) + 1.25]*(rho*u^2);
u2 = Gs/rho2;
jf2 = 4.6*10^-2;
N2 = 1;
```

```
delP2 = N2*[4*jf2*(4.830/od)*(1005/14.2)]*(rho2*u2^2);

//Increasing the baffle spacing pressure drop is reduced one−fourth
delPs = delP2/4;

ho = hs*(.5)^.8;              // [W/m^2.K]
U2 = [1/ho + Rw + .5*(od-id)/k3 + Ro*od/id+od/(id*hi)]^-1
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)
//END
```

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

```
clc;
clear;

printf("\n Example 9.29\n");

G=1; //Flow rate of organic liquid
Cp=2e3//Heat capacity of organic liquid
T1=350;
T2=330;
theta1=290;
theta2=320;

Q=G*Cp*(T2-T1);//heat load
G_cool=Q/(4187*(theta1-theta2));//flow of water
GCp_hot=(G*Cp);//for organic
GCp_cold=(G_cool*4187);

```

```
//From equation 9.235:
eta=GCp_hot*(T1-T2)/(GCp_cold*(T1-theta1));
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

```
clear;
clc;

printf('Example 9.30');

Tci = 320          //[K] Cold Fluid Initial
    Temperature
Tce = 340          //[K] Cold Fluid Final Temperature
mc = 4              //[kg/s] Flow rate of cold fluid
mh = 8              //[kg/s] Flow rate of hot fluid
Thi = [380 370 360 350]     //[K] Hot fluid initial
    temperature
Cp = 4.18               //[kJ/kg.K] mean heat capacity
U = 1.5          //[W/m^2.K] Overall heat transfer
    coefficient

GCpu= mh*Cp;                //[kW/K]
GCpp= mc*Cp;                //[kW/K]
if(GCpu<GCpp)
    GCpmin = GCpu;          //[kW/K]
    ratio = GCpmin/GCpp;
else
    GCpmin = GCpp;          //[kW/K]
    ratio = GCpmin/GCpu;

//Equation 9.235
n = mc*Cp*(Tce-Tci)*(mc*Cp*(Thi - Tci))^-1;
```

```
//From Figure 9.85b Number of transfer Units
N = [.45 .6 .9 1.7];     //[NTU]
A = N*GCpmin/U;         //Area of required [m^2]

format('v',4)
printf('\n\n    Thi(K)    n       N      A (m^2)');
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)
   ])
//END
```

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

```
clc;
clear;

printf("\n Example 9.31\n");

o_d=10e-3;//outer diameter of the tube
i_d=8.2e-3; //inner diameter of the tube
h=140;//coeffecient of heat transfer between gas and
    copper tube
k=350;//Thermal conductivity of copper tube
L=0.075;

b=%pi*o_d;//perimeter of tube
A=%pi/4*(o_d^2-i_d^2);//cross sectional area of the
   metal
m=((h*b)/(k*A))^0.5;
T_g=((475*cosh(m*L))-365)/(cosh(m*L)-1);
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

```
clc;
clear;

printf("\n Example 9.32\n");

d2=54e-3; //outer diameter of the tube
d1=70e-3; // fin diameter
w=2e-3; //fin thickness
n=230;// number of fins per metre run
T_s=370; //Surface temperature
T=280; //Temperature of surroundings
h=30; //Heat transfer coeffecient between gas and
    fin
k=43; //Thermal conductivity of steel
L=(d1-d2)/2;

theta1=T_s-T;
//Assuming that the height of the fin is small
    compared with its circumference
//and that it may be treated as a straight fin of
    length
l=(%pi/2)*(d1+d2);
b=2*l;//perimeter
A=l*w;//the average area at right-angles to the heat
     flow
m=((h*b)/(k*A))^0.5;
//From equation 9.254, the heat flow is given for
    case (b) as:
Qf=m*k*A*theta1*(%e^(2*m*L)-1)/(1+%e^(2*m*L));
Q=Qf*n;//Heat loss per meter run of tube
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

```
clc;
clear;

printf("\n Example 9.33\n");

d=150e-3;//Internal diameter of tube
d_o=168e-3;//outer diameter of tube
d_w=159e-3;
d_s=268e-3;
d_m=(d_s-d_o)/log(d_s/d_o);//log mean of d_o and d_s
h_i=8500;//The coefficient for condensing steam
    together with that for any scale
k_w=45;
k_l=0.073;
x_l=50e-3;
x_w=9e-3;
DT=444-294;
sigma=5.67e-8;
//The temperature on the outside of the lagging is
    estimated at 314 K and (hr + hc) will be taken as
     10 W/m2 K.
//total thermal resisitance
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;
Q_l=DT/R; //The heat loss per metre of length(from
    eq 9.261)
DT_lagging=((k_l*%pi*d_m/x_l)^-1/R)*DT;
//Taking an emissivity of 0.9, from equation 9.119:

h_r=(0.9*sigma*(310^4-294^4))/(310-294);
C=1.32;
//Substituting in equation 9.105 (putting l =
```

```
    diameter = 0.268 m):
h_c=C*((310-294)/d_s)^0.25;
//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:
Q_l=20*%pi*d_o*150;
printf("\n The heat loss to the air = %.2f kW/m",Q_l
    *1e-3);
```

Scilab code Exa 9.34 Determination of the economic thickness of lagging

```
clc;
clear;
printf('Example 9.34\n');


T1=420;//temperature of steam
k=0.1;//Thermal conductivity
T2=285; //Ambient temperature
h=10;//the coefficient of heat transfer from the
    outside of the lagging to
      //the surroundings
//determining Q/l from equation 9.21 and equating it
     to heat loss from the
//outside of the lagging we get
//(Q/l)=84.82/(log(d_o/0.1)+(0.02/d_o)) W/m
//using various equations we finally get an equation
     in terms of d_o and we
// will solve it by using fsolve
function [f]=F(d_o)
    f(1)=(1/(log(d_o/0.1)+(0.02/d_o))^2)-(2.35*(d_o
       ^3)/(d_o-0.02));
    funcprot(0);
endfunction
d_o=1;
```

```
ans=fsolve(d_o,F);
E_t=(ans-0.1)/2;
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

```
clc;
clear;

printf("\n Example 10.1\n");

x=1e-3; //Thickness of stagnant air film
D=1.8e-5;//Difffusivity of ammonia
R=8314; //Gas constant
T=295; //Temperature
P=101.3e3; //Total Pressure

//If the subscripts 1 and 2 refer to the two sides
    of the stagnant layer and
//the subscripts A and B refer to ammonia and air
    respectively ,
P_A1=.50*P;
P_A2=0;
P_B1=P-P_A1;
P_B2=P-P_A2;
P_BM=(P-P_A1)/log(P/P_A1);
```

```
//Thus, substituting in equation 10.31 gives:
N_A=(-D/(R*T*x))*(P/P_BM)*(P_A2-P_A1);
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

```
clear;
clc;

printf("Example 10.2\n");

th=[0 0 3 7 22 32 46 55 80 106];        //Time in
    hours
tm=[0 26 5 36 16 38 50 25 22 25];       //Time in min

//Conversion to kilo seconds
for i=1:10
    tm(i)=tm(i)*60;
    th(i)=th(i)*3600;
    tim(i)=(tm(i)+th(i))/1000;
end

L=[0 2.5 12.9 23.2 43.9 54.7 67.0 73.8 90.3 104.8];
      //in mm

Lo=L(1);

//Calculations are done as indicated in the
    procedure
//To obtain the values of x and y as below
//For plotting x and t axis of graph
x=L-Lo;

```

```
y(1)=0;
for j=2:10
    y(j)=tim(j)/(L(j)-Lo);
end

plot2d(x,y);
plot(x,y,'+');
xtitle('t/(L-L0) vs (L-L0)','(L-L0) in mm','t/(L-L0)
    in ks/mm^2');

//Calculation of slope
s=(y(4)-y(3))/(x(4)-x(3))*10^3*10^6;
printf("\nSlope is %.2e sec/m^2\n",s);

Vl=22.4;                    //Molar volume in litres
den=1540;                   //Density in kg/m^3
T0=273;
T=321;
vp=37.6;                    //vapour pressure in kPa
P0=101.3;                   //PRessue in kPa
M=154;

Ct=T0/(Vl*T);
Ca=(vp*Ct)/P0;

Cb1=Ct;
Cb2=(P0-vp)*Ct/P0;
Cbm=(Cb1-Cb2)/log(Cb1/Cb2);

//Diffusivity calculation
D=den*Cbm/(2*M*Ca*Ct*s);
printf("\nDiffusivity is %.2e m^2/s\n",D);

//End
```

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

```
clear;
clc;

printf("Example 10.3\n");

P=101.3e3; //pressure of the operating column
T=295; //Temperature of the operating column
P_A=7e3; //partial pressure of ammonia
x=1e-3; //=(y1-y2)Thickness of stationary gas film
D=2.36e-5; //Diffusivity of ammonia

C_A=(1/22.4)*(273/T)*(P_A/P);//=(C_A1-C_A2)
    Concentration of ammonia gas
//X=C_T/C_BM
X=P*log(P/(P-P_A))/(P-(P-P_A));
//From equation 10.33
N_A_=(D/x)*X*(C_A);
printf("\n The transfer rate per unit area = %.2f
    *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

```
clear;
clc;

printf("Example 10.4\n");

Q=3e-6; //Flow rate of water
Meu=1e-3; //Viscosity of water
D=1.5e-9; //diffusivity of carbon dioxide in water
rho=1e3; //Density of water

```

```
//the mean velocity of flow is governed by equation
    3.87 in which sin(phi)is
//put equal to unity for a vertical surface:
s=(Q*1e2*Meu*3/(rho*9.81))^(1/3);//Thickness of film

//A=Ux/Us=0.95;
A=0.95;
y=s*(1-A)^0.5;//The distance below the free surface
//using equation 10.108 and using tables of error
    fuctions
t=(1.305/1.822)^2
Us=rho*9.81*s^2/(2*Meu);//surface velocity
L=Us*t;//The maximum lend=gth of column
printf("\n The maximum length of column = %.2f m",L)
    ;
```

Scilab code Exa 10.5 Calculation of equivalent resisitance and time required

```
clear;
clc;

printf("Example 10.5\n");

N_dot=50; //Initial maas transfer rate
D=1.8e-9; //Diffusivity of gas in liquid phase

C_bg=(1/22.4)*(273/293); //bulk gas concentration
N_C=N_dot*C_bg; //Initial maas transfer rate in
    terms of cocentration
h=N_C/0.04;// Effective Mass transfer coefficient
R=1/h;//Equivalent resistance
printf("\n Equivalent resistance = %.4f s/m",R);
R_l=R*9; //Liquid phase resistance
h_l=1/R_l; //Liquid phase coefficient
```

```
//From equation 10.113 and using liquid phase
    resistance
t=R_l^2/(%pi/D);

printf("\n The required time is = %.2f *10^-11s",t*1
    e11)
```

Scilab code Exa 10.6 Calculating proportion of absorbed carbon dioxide

```
clear;
clc;

printf("Example 10.6\n");

//Diffusivity of CO2 in ethanol
D=4D-9;                  //in m^2/s
t=100;                   //Time in sec

//Solving all the integral as defined in the proces
//as per described in the book
//a useful result is obtained

Cai=poly([0],'x');

y=[0 10^-3];

for i=1:2
    mole(i)=((2*sqrt(D*t/%pi)*exp(-y(i)^2/(4*D*t)))
        -(y(i)*erfc(y(i)/(2*sqrt(D*t)))));
end
ret=(mole(1)-mole(2))/mole(1);

printf("\nProportion retained is %.1f %%\n",ret*100)
    ;
```

```
//End
```

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

```
clc;
clear;

printf("Example 10.8\n")

L=825e-3; //length of the tube
d=15e-3; //diameter of the tube
P_i=7.5e3; //Partial pressure of ammonia at inlet
P_o=2e3; //Partial pressure of ammonia at inlet
A_r=2e-5; //Air rate
P=101.3e3; //Atmospheric pressure

D_F_m=(P_i-P_o)/log(P_i/P_o);//Mean driving force
A_absorbd=A_r*((P_i/(P-P_i))-(P_o/(P-P_o)));
A_w=%pi*d*L;//Wetted surface
K_G=(A_absorbd/(A_w*D_F_m));//Overall transfer
    coefficient
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

```
clear;
clc;

printf("\n Example 10.9\n");

```

```
//The proces is defined by
//     Ca = B1*exp(sqrt((k/D)y)) + B2*exp(-sqrt((k/D)
   y))

//Boundary conditions as
//     Ca=Cai          at y=0
//     Ca=Cai/2        at y=l

//Using above 3 equations, final equation is derived
    as follows

//Assuming
//     ratio = (Na)y=l / (Na)y=0
//     z=l*sqrt(k/D)

z=0.693;

ratio=(exp(sqrt(z))+exp(-sqrt(z))-4)/(2*(1-exp(-sqrt
   (z))-exp(sqrt(z))));

printf("\n The final ratio is %.2f \n",ratio);

//End
```

Scilab code Exa 10.10 Determination of order of chemical reaction

```
clc;
clear;

printf("Example 10.10\n");

//The mass transfer rate (moles/unit area and unit
   time) is given by equation
//10.180, where denoting the original conditions by
   subscript 1 and the
```

```
//conditions at the higher temperature by subscript 2 gives
//N_A2=0.83*N_A1
//Substituting the numerical values gives:
n=2*(log(0.83/(1.35)^0.5)/log(0.8))-1;
printf("\n n = %.2f ",n);
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

```
clear;
clc;
//Coulson and Richardson's Chemical Engineering Volume I
//Chapter 10 Example 11
//Page 630
printf('Example 10.11');
//What factor will the mass transfer rate across interface change

k = 2.5*10^-6              //[s^-1] Rate constant
E = 2.643*10^7             //[J/kmol] Energy of Activation
R = 8314                   //[J/kmol.K] Universal gas contss
D = 10^-9                  //[m^2/s] MOlecular diffuisivity
L = .01                    //[m] Film Thickness

//At T =293K
T = 293                   //[K] temperature
A = k/exp(-E/(R*T));      //[s^-1]
e = exp(-2*L*sqrt(k/D));
N = sqrt(k/D)*(1+e)/(1-e); //Consider relative
```

```
Solubility at 293 K be unity

//At T =313K
T2 = 313                    //[K] temperature
k2 = A*exp(-E/(R*T2));      //[s^-1]
e2 = exp(-2*L*sqrt(k2/D));
N2 = .8*sqrt(k2/D)*(1+e2)/(1-e2); //Consider
    relative Solubility at 313 K be .8 wrt that of
    293K

Nr = N2/N;

printf('\n\nChange in mass transfer rate is given by
    factor %.2f',Nr)
//END
```

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

```
clc;
clear;

printf("\n Example 10.12\n");

k=5e-4;//first order rate constant
D_e=2e-9;//effective diffusivity of reactants in the
    pores of the particles

lambda=(k/D_e)^0.5;
// (i) For the platelet of thickness 8 mm,
L=0.5*(8e-3);
phi=lambda*L;//thiele modulus
//From equation 10.202, the effectiveness factor '
    eta' is given by:
eta=(1/phi)*tanh(phi);
```

```
printf("\n (i) Thiele modulus = %.1f",phi);
printf("\n     The effectiveness factor = %.3f",eta);

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

```
clc;
clear;
printf("\n Example 10.13\n");

D_e=1e-5;//Effective diffusivity for the reactants in the catalyst particle
k=14.4;//first order rate constant
L=2.5e-3;

lambda=(k/D_e)^0.5;
phi=(k/D_e)^0.5*(L);//Thiele modulus
//From equation 10.202, the effectiveness factor,
eta=(1/phi)*tanh(phi);
printf("\n (i) The effectiveness factor = %.3f",eta);
//The concentration profile is given by equation 10.198
```

```
y=1.25e-3;
C_Ai=0.15;
C_A=C_Ai*(cosh(lambda*y)/cosh(lambda*L));
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

```
clear;
clc;

printf("\n Example 10.14\n");

R_r=8.2e-2;//reaction rate when concentration =0.011 kmol/m^3
D_e=7.5e-8;//Effective diffusivity

//Since the value of the first-order rate constant is not given, lambda and
//phi_l cannot be calculated directly. The reaction rate per unit volume of
//catalyst = eta*k*C_Ai (equation 10.217),
//eta=phi_L^-1
//It is assumed that the reactor is operating in this regime and the assumption
//is then checked.Substituting numerical values in equation 10.217:
k=(1.217*R_r/0.011)^2;
phi_L=1.217*(k)^0.5;
eta=phi_L^-1;
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

```
clc;
clear;

printf("\n Example 11.1\n");

u_s=1; //Velocity of water
w=0.6; //Width of plane surface
l=1; //Length of plane surface
A=0.6*1; //Area of the surface
//Taking
Meu=1e-3;//Viscosity of water
rho=1000; //Density of water
//Mean value of S/pw2 from equation 11.41
//X=R/(rho*u^2)
X=0.00214;
F=X*rho*u_s^2*A;
printf("\n Total drag force = %.2f N",F);
```

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

```
clc;
clear;

printf("\n Example 11.2\n");

x=150e-3; //Distance from leading edge where
    thicness is to be found
Meu_o=0.05; //viscosity of oil
rho_o=1000; //Density of oil
u=0.3; //Velocity of flow

Re_x=x*u*rho_o/Meu_o;
//For streamline flow:
//from equation 11.17
del=4.64*x/Re_x^0.5;//thickness of the boundary
    layer
printf("\n The thickness of the boundary layer = %.1
    f mm",del*1e3);
//from equation 11.20
del_star=0.375*del;
printf("\n The displacement thickness = %.1f mm",
    del_star*1e3);
```

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

```
clc;
clear;

printf("\n Example 11.3\n");

D=50e-3; //Diameter of the pipe
```

```
Q=2e-3; //Flow rate of benzene through pipe
rho_b=870; //Density of benzene
Meu_b=0.7e-3; //Viscosity of benzene

G=Q*rho_b; //Mass flow rate of benzene
Re=4*G/(Meu_b*%pi*D); //Reynolds number

//From equation 11.49:
del_b=62*D*Re^(-7/8);
printf("\n The thickness of the laminar sub-layer =
    %.3f mm",del_b*1e3);

area=%pi/4*D^2; //Cross sectional are of pipe
u=G/(rho_b*area); //mean velocity

//From equation 11.47:
u_b=2.49*u*Re^(-1/8);
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

```
clc;
clear;

printf("\n Example 12.1\n");

d=250e-3; //internal diameter of pipe
u=15; //Velocity of air through the pipe
y1=50e-3; //First point where velocity is to be
    found out
y2=5e-3; //Second point where velocity is to be
    found out
rho_air=1.10; //Density of air
Meu_air=20e-6; //Viscosity of air

Re=d*u*rho_air/Meu_air;
//Hence, from Figure 3.7: X=R/(rho*u^2)=0.0018
X=0.0018;
u_s=u/0.817;
u_star=u*X^0.5;
```

```
//At 50 mm from the wall:
y1_r=2*y1/d;// y/r
//Hence, from equation 12.34:
u_x1=u_s+2.5*u_star*log(y1_r);
printf("\n The fluid velocity at 50 mm from the wall
    = %.1f m/s",u_x1);

//At 5 mm from the wall:
y2_r=2*y2/d;// y/r
//Hence, from equation 12.34:
u_x2=u_s+2.5*u_star*log(y2_r);
printf("\n The fluid velocity at 5 mm from the wall
    = %.1f m/s",u_x2);

//The thickness of the laminar sub-layer is given by
    equation 12.54:
del_b=5*d/(Re*X^0.5);
printf("\n The thickness of the laminar sub-layer =
    %.3f mm",del_b*1e3);
```

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

```
clc;
clear;

printf("\n Example 12.2\n");

u=10; //Velocity of air
T=330; //Temperature of air
d=25e-3; //Inner diameter of pipe
T_p=415; //Temperature at which the pipe is
    maintained
DP_l=80; //Drop of static pressure along the pipe
```

```
per unit length

//From equations 12.98 and 3.18:
//we get h=0.05*Cp
//The heat taken up per unit time by the air dQ =0.0052*Cp*dT......(i)
//The heat transferred through the pipe wall is also given by: = 0.039*Cp*(415-T)......(ii)
//Equating (i) & (ii)
//On integrating we get
T_0=415-(85/exp(0.45))
printf("\n The required air Temperature = %d K",T_0);
```

Scilab code Exa 12.3 Estimation of outlet temperature by different methods

```
clc;
clear;

printf("\n Example 12.3\n");

u=3.5; //Velocity of water
d=25e-3; //Diameter of the pipe
l=6; //Length of the pipe
T1=300; //Temperature at enterance
T2=330; //Temperature at exit
rho=1000; //density of water at 310 K
Meu=0.7e-3; //Viscosity of water at 310 K
//Taking the fluid properties at 310 K and assuming that fully developed flow exists
Cp=4.18e3; //heat capapcity
k=0.65; //Thermal conductivity

Re=d*u*rho/Meu;
```

```
Pr=Cp*Meu/k;

printf("\n (a) Reynolds analogy");
h1=0.032*(Re^-0.25)*Cp*rho*u;//....Equation 12.139
printf("\n h = %.2f kW/m^2 K",h1*1e-3);
// on solving we get final equation as
theta_dash1=330-10^(log10(30)-(0.0654*h1*1e-3/2.303)
    );
printf("\n The outlet temperature = %.1f K",
    theta_dash1)

printf("\n\n (b) Taylor Prandtl Equation");
h2=0.032*(Re^-0.25)*(1+2*Re^(-1/8)*(Pr-1))^-1*Cp*rho
    *u;
printf("\n h = %.2f kW/m^2 K",h2*1e-3);
// on solving we get final equation as
theta_dash2=330-10^(log10(30)-(0.0654*h2*1e-3/2.303)
    );//....Equation 12.140
printf("\n The outlet temperature = %.1f K",
    theta_dash2)

printf("\n\n (c) Universal velocity profile equation
    ");
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
printf("\n h = %.2f kW/m^2 K",h3*1e-3);
// on solving we get final equation as
theta_dash3=330-10^(log10(30)-(0.0654*h3*1e-3/2.303)
    );
printf("\n The outlet temperature = %.1f K",
    theta_dash3)

printf("\n\n (d) Nu=0.023*Re^0.8*Pr^0.33");
h4=k/d*0.023*Re^0.8*Pr^0.33;
printf("\n h = %.2f kW/m^2 K",h4*1e-3);
// on solving we get final equation as
theta_dash4=330-10^(log10(30)-(0.0654*h4*1e-3/2.303)
    );
```

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

Scilab code Exa 12.4 Estimation of outlet temperature by different methods

```
clc;
clear;

printf("\n Example 12.4\n");

u=3.5; //Velocity of air
d=25e-3; //Diameter of the pipe
l=6; //Length of the pipe
T1=290; //Temperature at enterance
T2=350; //Temperature at exit
rho=29/22.4*273/310; //density of air at 310 K
Meu=0.018e-3; //Viscosity of air at 310 K
//Taking the physical properties at 310 K and
    assuming that fully developed flow exists
Cp=1.003e3; //heat capapcity
k=0.024; //Thermal conductivity

Re=d*u*rho/Meu;
Pr=Cp*Meu/k;

printf("\n (a) Reynolds analogy");
h1=0.032*(Re^-0.25)*Cp*rho*u;//....Equation 12.139
printf("\n h = %.2f W/m^2 K",h1);
// on solving we get final equation as
theta_dash1=350-10^(log10(60)-(239.88*h1*1e-3/2.303)
    );
printf("\n The outlet temperature = %.1f K",
    theta_dash1)

```

```
printf("\n\n (b) Taylor Prandtl Equation");
h2=0.032*(Re^-0.25)*(1+2*Re^(-1/8)*(Pr-1))^-1*Cp*rho*u;
printf("\n h = %.2f W/m^2 K",h2);
// on solving we get final equation as
theta_dash2=350-10^(log10(60)-(239.88*h2*1e-3/2.303));//....Equation 12.140
printf("\n The outlet temperature = %.1f K",theta_dash2)

printf("\n\n (c) Universal velocity profile equation");
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
printf("\n h = %.2f W/m^2 K",h3);
// on solving we get final equation as
theta_dash3=350-10^(log10(60)-(239.88*h3*1e-3/2.303));
printf("\n The outlet temperature = %.1f K",theta_dash3)

printf("\n\n (d) Nu=0.023*Re^0.8*Pr^0.33");
h4=k/d*0.023*Re^0.8*Pr^0.33;
printf("\n h = %.2f W/m^2 K",h4);
// on solving we get final equation as
theta_dash4=350-10^(log10(60)-(239.88*h4*1e-3/2.303));
printf("\n The outlet temperature = %.1f K",theta_dash4)
```

Chapter 13

Humidification and Water Cooling

Scilab code Exa 13.1 Pressure calculation

```
clc;
clear;

printf("\n Example 13.1\n");

P=101.3e3;
T=297;
R=8314; //gas constant
RH=60; //Relative humidity
p_b1=12.2e3;//Vapor pressure at 297 K
p_b2=6e3; //Vapor pressure at 283 K
M_w=78; //molecular weight of benzene
M_a=28; //Mass of nitrogen

//From the definition of percentage relative
   humidity (RH)
P_w=(p_b1)*(RH/100);
//In the benzene-nitrogen mixture:
m_b=P_w*M_w/(R*T);//mass of benzene
```

```
m_n=(P-P_w)*M_a/(R*T);//mass of nitrogen
H=m_b/m_n; //Humidity at 297 K

//In order to recover 80 per cent of the benzene, the humidity must be reduced to 20 per cent of the initial value
H_o=H*.20;
//Thus in equation 13.2
P_r=p_b2+(p_b2/M_a*M_w)/H_o;
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

```
clc;
clear;

printf("\n Example 13.2\n");

P=101.3e3; //Given pressure
T=300; //Given Temperature
RH=25; //Percentage relative humidity of water
P_wo=3.6e3; //partial pressure of water vapour when air is saturated with vapour
M_w=18; //Molecular weight of water
M_a=29; //Molecular weight of air
R=8314; //gas constant

printf("\n (a)\n The partial pressure of the water vapour in the vessel = ")
P_w=P_wo*(RH/100);
printf("%.1f kN/m^2",P_w*1e-3);

printf("\n (b)");
```

```
m_w=P_w*M_w/(R*T);//mass of water vapour
m_a=(P-P_w)*M_a/(R*T);//mass of water air
Vs_w=1/m_w;//specific volume of water vapour at 0.9 kN/m^2
Vs_a=1/m_a;//specific volume of air at 100.4 kN/m^2
printf("\n Specific volume of water vapour = %.0f m^3/kg",Vs_w);
printf("\n Specific volume of air = %.3f m^3/kg",Vs_a);

H=m_w/m_a;//Humidity
printf("\n (a)\n Humidity of air = %.4f kg water/kg air",H);
H_v=Vs_a;//Humid volume
printf("\n Humid volume = %.3f m^3/kg",H_v);

H_p=(P-P_wo)/(P-P_w)*RH; //Percentage humidity
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

```
clc;
clear;

printf("\n Example 13.3\n");

T=310; //Temperature of moist air
T_w=300; //Wet bulb tempeature
L=2440e3; //Latent heat of vapourisation of water at 300 K
P=105e3; //Given total pressure
P_wo1=3.6e3; //Vapour pressure of water vapour at 300 K
```

```
P_wo2=6.33e3; //Vapour pressure of water vapour at
    310 K
M_w=18; //Molecular weight of water
M_a=29; //Molecular weight of air

H_w=(P_wo1/(P-P_wo1))*(M_w/M_a); //The humidity of
    air saturated at the wet-bulb temperature
//Therefore, taking (h/hD*rho*A) as 1.0 kJ/kg K, in
    equation 13.8:
H=H_w-(1e3/L)*(T-T_w);
printf("\n The humidity of the air = %.3f kg/kg",H);

//In equation 13.2:
x=poly([0],'x');
P_w=roots(H*(P-x)*M_a-M_w*x);
RH=P_w/P_wo2*100;
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

```
clc;
clear;

printf('Example 13.4');

//Refer HUMIDITY ENTHALPY PLOT Figure 13.5 Page 748
    as Humidity Chart
//According the given passes and situatuion
T = [325 301 308 312 315]          //[K]
H = [.005 .015 .022 .027 .032]     //[kg/kg]
//From Humidity Chart on humidifying to 60 percent
    humidity
```

```
Tw = [296 301 305 307]              //[K]

Hin = H(5)-H(1)             //[kg/kg] Increase in Humidity

printf('\n\n (a) The temperature of the material on each tray (in Kelvin)')
disp(Tw);
printf(' Thus the air leaving the system is at %i K and 60 per cent humidity.',T(5));

//From Humitidy Chart at the obtained leaving conditions
v = .893          //[m^3/kg] Specific Volume of dry air
vs = .968         //[m^3/kg] Specific Volume of Saturated air
vh = .937         //[m^3/kg] Humid Volume of air of 60 per cent humidity by Interpolation of Curve in  Humidity Chart
x = 5         //[m^3/s] Amount of moist air leaves the dryer in (b)
m = x/vh          //[kg/s] Mass of air passing through  the dryer
mw = m*Hin        //  [kg/s] Mass of water evaporated

printf('\n\n (b) If 5 m^3/s moist air leaves the dryer, The amount of water removed is %.3f kg/s.',mw)
Tb = 370              //[K] dry Bulb temperature corresponding to humidity of .005 kg/kg and wet-bulb temperature 307 K
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)

//END
```

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

```
clc;
clear;

printf('Example 13.5\n')

G1=1; //flow rate of air at 350 K
PH1=10; //Percentage Humidity at 350 K
G2=5; //flow rate of air at 300 K
PH2=30; //Percentage Humidity at 300 K

//from fig 13.4
H1=0.043; //Humidity at 350 K and 10 percent
    humidity
H2=0.0065; //Humidity at 300 K and 30 percent
    humidity
//Thus, in equation 13.23:
H=((G1*H1)+(G2*H2))/(G1+G2);
printf("\n Humidity of final stream = %.4f kg/kg",H)
    ;

//from fig 13.5
H_1=192e3;//Entahlpy at 350 K and 10 percent
    humidity
H_2=42e3;//Enthalpy at 300 K and 30 percent humidity
x=poly([0],'x');
H_=roots((G1*(x-H_1))-(G2*(H_2-x)));
printf("\n Entahlpy of the resultant stream = %.0f
    kJ/kg",H_*1e-3);

//From Figure 13.5:
//at H_ (Enthalpy)= 67 kJ/kg and H(humidity) =
```

```
0.0125 kg/kg
T=309;
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

```
clc;
clear;

printf('Example 13.6\n')

G_s=0.15; //Mass flow rate of steam
T=400; //Temperature to which steam is superheated
T_a=320;//Tremperature of air
RH_a=20; //Percentage relative humidity of air
G_a=5;//Mass flow rate of air
L=2258e3; //latent heat of steam
Cp=2e3; //Specific heat of superheated steam
//Enthalpy of steam
H_3=4.18*(373-273)+L+Cp*(T-373);
//From Figure .13.5:
//at T=320 K and 20 percent Relative Humidity
H1=0.013;//Humidity
H_1=83e3;//Enthalpy
//By making required constructions we get
H=0.043;
printf("\n Relative humidity of stream= %.3f kg/kg",H);
H_=165e3;
printf("\n Entahlpy of stream = %d kJ/kg",H_*1e-3);
T_s=324;
printf("\n Temperature of stream = %d K",T_s);

```

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