Scilab Textbook Companion for
A Heat Transfer Text Book
by J. H. Lienhard, IV And J. H. Lienhard, V[1]

Created by
Apar Tiwari
B-Tech part 2
Chemical Engineering
IT BHU
College Teacher
Mrs. V.l. Yadav
Cross-Checked by

July 31, 2019

---

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

Book Description

Title: A Heat Transfer Text Book

Author: J. H. Lienhard, IV And J. H. Lienhard, V

Publisher: Phlogiston Press, U.S.A.

Edition: 3

Year: 2008

ISBN: 978097138352

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

Introduction

Scilab code Exa 1.1 `heat flux and heat transfer rate`

```

clear;
clc;

printf("\t Example 1.1\n");

k=35; //Thermal Conductivity, W/m*K
T1=110;// Temperature of front
T2=50; // Temperature of back,C
A=0.4;//area of slab,m^2
x=0.03; //Thickness of slab,m

q=-k*(T2-T1)/(1000*x); //formula for heat flux
printf("\t heat flux is: %.0f KW/m^2\n",q);

Q=q*A; //formula for heat transfer rate
printf("\t heat transfer rate is: %.0f KW\n",Q);

//End
```

Scilab code Exa 1.2 `Temperature Distribution`

```

clear;
clc;
printf("\tExample 1.2\n");
x=poly([0],'x');
k1=372; // Thermal Conductivity of slab,W/m*K
x1=0.003; // Thickness of slab,m
x2=0.002;// Thickness of steel,m
k2=17; // Thermal Conductivity of steel,W/m*K
T1=400; // Temperature on one side,C
T2=100;//Temperature on other side,C

Tcu=roots(x+2*x*(k1/x1)*(x2/k2)-(400-100));

//q=k1*(Tcu/x1)=k2*(Tss/x2);

Tss = Tcu*(k1/x1)*(x2/k2); // formula for temperature gradient in steel  side

Tcul=T1-Tss;
Tcur=T2+Tss;
printf("\t temperature on left copper side is : %.0f C\n",Tcul);
printf("\t Temperature on right copper  side is : %.0f C\n",Tcur);
q=k2*Tss/(1000*x2); // formula for heat conducted
printf("\t heat conducted through the wall is : %.0f W\n",q);
printf("\t our initial approximation was accurate within a few percent.");
//End
```

Scilab code Exa 1.3 `heat transfer coefficient calculation`

```

clear;
clc;
printf(”\t example 1.3\n”);
q1=6000; //Heat flux , W*m^-2
T1=120; // Heater Temperature , C
T2=70; //final Temperature of Heater
q2=2000; // final heat flux
h=q1/(T1-T2);// formula for average heat transfer coefficient
printf(”\t Average Heat transfer coefficient is:%.0f W/(m^2*K)\n”,h);

Tnew=T2 + q2/h; //formula for new Heater temperature
printf(”\t new Heater Temperature is:%.2f C\n”,Tnew);
//End
```

Scilab code Exa 1.4 `response of thermocouple`

```

clear;
clc;
printf(”\t Example 1.4\n”);
h=250; //Heat Transfer Coefficient , W/(m^2*K)
k=45; // Thermal Conductivity , W/(m*K)
c=0.18; //Heat Capacity , kJ/(kg*K)
a=9300; //density , kg/m^3
T1=200; //temperature , C
D=0.001; //diameter of bead ,
```

```
t1 =0:0.1:5;
T=200-180*exp(-t1/((a*c*D*1e3)/(6*h)));
plot(t1,T);
xtitle("Thermocouple response to a hot gas flow","time,t1 sec","temperature,T C");
Bi = h*(0.001/2)/45; //biot no.
printf("The value of Biot no for this thermocouple is %f",Bi);
printf("\n Bi is <0.1 and hence the thermocouple could be considered as a lumped heat capacity system and the assumption taken is valid.\n");
//End
```

Scilab code Exa 1.5 Temperature of thermocouple

```

clear;
clc;

printf("\t Example 1.5\n");
x=poly([0],'x');
T1=293; //Temperature of air around thermocouple, K
T2=373; //Wall temperature, K
h=75; // Average Heat Transfer Coefficient, W/(m^2*K)
s=5.67*10^-8; // stefan Boltzman constant, W/(m^2*K^4)
x=roots(h*(x-T1)+s*(x^4-T2^4));
y=x(4)-273;
printf("\t thermocouple Temperature is : %.1f C\n",y);
//end
```

Scilab code Exa 1.6 `Temperature of thermocouple`

```

clear;
clc;

printf("\t example 1.6\n");
x=poly([0],'x');
e=0.4; //emissivity
T1=293; //Temperature of air around Thermocouple, K
T2=373; // wall Temperature, K
h=75; // Average Heat Transfer Coefficient, W/(m^2*K)
s=5.67*10^-8; // stefan Boltzman constant, W/(m^2*K^4)
x=roots((x-T1)*h+e*s*(x^4-T2^4));
y=x(4)-273;
printf("\t Thermocouple Temperature is : %.1f C\n",y);
//End
```

Chapter 2

Heat conduction concepts and heat transfer coefficient

Scilab code Exa 2.3 `steady flux`

```

clear;
clc;

printf("\t Example 2.3\n");

l=1; // tube length, m
m=0.01; // mass fraction
D12=2.84*10^-5; // diffusivity, m^2/s
a=1.18; // density, kg/m^3

J=a*D12*m/l;
//steady state flux of water from one side to the other,kg/(m^2*s)
printf("\t steady flux of water is %.2e kg/(m^2*s)",J);
//end
```

Scilab code Exa 2.4 `thickness calculation`

```

clear;
clc;

printf("\t Example 2.4\n");

k=18; // thermal conductivity of ressistor , W/(m*K)
A=1;     //area of slab surface , m^2
hc=3000;     //convective heat transfer coefficient ,W/(m^2*K)
//Req=1/A*(2L/k+1/hc) , for contact ressistances to be neglected 2L/18 must be very greater than the 1/3000
printf("thickness of slabs for contact ressistances to be nelected is very greater than 0.003 m. if length is 3 cm, the error is about 10 percent.");
//end
```

Scilab code Exa 2.7 `Critical radius of insulation`

```

clear;
clc;

printf("\t example 2.7\n");

h=20; //convective heat transfer coefficient , W/(m^2*K)
k=0.074; //thermal conductivity , J/(m*K)
```

```
Ro=k/h; // formula for critical thickness of insulation
printf("\t critical thickness of insulation is : %.4f m\n",Ro);
printf("\t insulation will not even start to do any good until ratio of outer radius and inner radius is 2.32 or outer radius is  0.0058 m.")
//End
```

Scilab code Exa 2.8 `Ressistor temperature calculation`

```

clear;
clc;

printf("\t Example 2.8\n");

P=0.1; //dissipating power,W
D=0.0036; //outer diameter of cylinder, m
l=0.01; //length of cylinder, m
T=308; // temperature of air in the cabinet,K
h=13; // convection coefficient, W/(m^2*K)
e=0.9;
A=1.33*10^-4; //area of ressistor's surface, m^2

Tm=(T+323)/2; // ressistor's temperature at 50 K
Hr=4*5.67*10^-8*Tm^3*e; // radiative heat transfer coefficient,W/(m^2*K)


Rteq=1/(A*(Hr+h));
Tres=T+P*Rteq;
//we guessed a ressistor's temperature of 323K in finding Hr,recomputing with this higher temperature,we have Tm=327K and Hr=7.17W/(m^2*K).
```

```
   if we repeat the rest of calculations, we get a
   new value Tres=345.3K, since the use of hr is an
   approximation, we should check its applicability:
    1/4*((345.3-308)/327)^2=0.00325<<1, in this case
   , the approximation is a very good one
Tr=Tres-273.06;
printf("\t temperature of ressistor is : %.2f K\n",
   Tr);
printf("\t since 1/4*(temperature diffference/mean
   temperature)= 1/4*((72.3-35)/327)^2=0.00325<<1,
   in this case, the approximation is a very good
   one.");
//End
```

Scilab code Exa 2.9 `time of cooling of ressistor`

```

clear;
clc;

printf("\t Example 2.9\n");

k=10; // thermal conductivity of ressistor, W/(m*K)
a=2000; //density of ressistor, kg/m^3
l=0.01; //length of cylinder, m
A=1.33*10^-4; //area of ressistor's surface, m^2
T1=308; // temperature of air in the cabinet,K
Cp=700; //heat capacity of ressistor, J/kg/K
Heff=18.44; // the effective heat  transfer
   coefficient of parallel convection and radiation
   process,  W/(m^2*K)
Bi=Heff*(0.0036/2)/k;
T=a*Cp*3.14*l*(0.0036)^2/(4*Heff*A); //since from
   previous example,To=72.3C, we have Tres=T1+(To-T)
   *exp(-t/T),Tres=308+(37.3)*exp(-t/T). 95% of the
```

```
temperature drop has occured when t=T*3=174s.
t=3*T;
printf("\t time for 95 percent cooling of ressistor is : %.0f s\n",t);
//End
```

Scilab code Exa 2.10 `heat transfer coefficient calculation`

```

clear;
clc;

printf("\t Example 2.10\n");

h1=200; // convective heat transfer coefficient , W/(m^2*K)
a=1/60000; // 1/a=l/Kal, l=0.001m, Kal=160 W/(m*K)
h2=5000; //convective heat transfer coefficient during boiling ,W/(m^2*K)

U=1/(1/h1+a+1/h2)+0.40;
printf("\t overall heat transfer coefficient is : %.1f W/(m^2*K)\n",U);
//end
```

Scilab code Exa 2.12 `redesign of siding`

```
 clear;
clc;

printf("\t Example 2.12\n");
Rf=0.0005;   //fouling ressistance ,m^2*K/W
U=5;   //heat transfer coefficient ,W/(m^2*K)
```

```
Ucor=(U*Rf+1)/(U);
printf("\t corrected heat transfer coefficient is : %.2f W/(m^2*K)\n therefore the fouling is entirely irrelevant to domestic heat holds.",Ucor);
//end
```

Scilab code Exa 2.13 `fouling calculation`

```

clear;
clc;

printf("\t Example 2.13\n");
U1=4000; //overall heat transfer coefficient of water cooled steam condenser, W/(m^2*K)
Rf1=0.0006; // lower limit of fouling ressistance of water side, m^2*K/W
Rf2=0.0020; // upper limit of fouling ressistance of water side, m^2*K/W
U2=U1/(U1*Rf1+1);
U3=U1/(U1*Rf2+1);
printf("\t upper  limit of the corrected overall heat transfer coefficient is : %.0f W/(m^2*K)\n",U2);
printf("\t lower limit of corrected overall heattransfer coefficient is : %.0f W/m^2/K, U is reduced from 4000 to between 444 and 1176 W/(m^2*K),fouling is crucial in this case and engineering was in serious error.\n",U3);
//end
```

Chapter 3

Heat Exchanger Design

Scilab code Exa 3.3 `heat transfer coefficient calculation`

```

clear;
clc;

printf("\t Example 3.3\n");

T1=293; // Entering Temperature of Water, K
T2=313; //Exit Temperature of water, K
m=25/60;//Condensation rate of steam, kg/s
T3=333; // Condensation Temperature,K
A=12; //area of exchanger, m^2
h=2358.7*10^3; //latent heat, J/kg

U=(m*h)/(A*((T2-T1)/log((T3-T1)/(T3-T2))))+0.6;
printf("\t Overall heat transfer coefficient is : %.0f W/(m^2*K)\n",U);

Mh=(m*h)/(4174*(T2-T1));
printf("\t required flow of water is : %.2f kg/s\n",Mh);
//End
```

Scilab code Exa 3.4 `heat exchanger area`

```

clear;
clc;

printf("\t Example 3.4\n");

m=5.795; //flow rate of oil , kg/s
T1=454; //Entering Temperature of oil , K
T2=311; //Exit Temperature of oil , K
T3=305; // Entering Temperature of water , K
T4=322; //Exit Temperature of water , K
c=2282; //heat capacity , J/(kg*K)
U=416; //overall heat transfer coefficient , J/(m^2*
   K*s)
F=0.92; // Correction factor for 2 shell and 4 tube-
    pass exchanger ,          since R=(T1-T2)/(T4-T3)
    =8.412 >1, P=(T4-T3)/(T1-T2)=0.114,we can get
    this value of F by using value of P =R*0.114

A=(m*c*(T1-T2))/(U*F*((T1-T4-T2+T3)/log((T1-T4)/(T2-
   T3))));
printf("\t area for heat exchanger is : %.1f m^2\n",
    A);
//End
```

Scilab code Exa 3.5 `heat transfer temperature`

```

clear;
```

```
clc;

printf(”\t Example 3.5\n”);

T1=313; //entering temperature of cold water, K
T2=423; //Entering temperature of hot water, K
Cc=20000; //heat capacity of cold water, W/K
Ch=10000; //heat capacity of hot water, W/K
A=30; //area, m^2
U=500; //overall heat transfer coefficient, w/(m^2*K)
e=0.596; // no. of transfer units(NTU)=(U*A)/Ch=1.5, the effectiveness of heat exchanger e can be found by using this value of NTU

Q=e*Ch*(T2-T1);
Q1=Q/1000
printf(”\t heat transfer is :%.1f KW\n”,Q1);

Texh=T2-Q/Ch;
Tn1=Texh-273;
printf(”\t the exit hot water temperature is:%.2f C\n”,Tn1);

Texc=T1+Q/Cc
Tn2=Texc-273;
printf(”\t the exit cold water temperature is : %.2f C\n”,Tn2);

//End
```

Scilab code Exa 3.6 `area calculation`

```

clear;
```

```
clc;

printf("\t Example 3.6\n");

T1=313; //entering temperature of cold water, K
T2=423; //Entering temperature of hot water, K
T3=363; //Exit temperature of hot water, K
Cc=20000; //heat capacity of cold water, W/K
Ch=10000; //heat capacity of hot water, W/K
U=500; //overall heat transfer coefficient, w/(m^2*K)
T4=T1+(Ch/Cc)*(T2-T3);
e=(T2-T3)/(T2-T1);

NTU=1.15;
A1=Ch*(NTU)/U;  // since NTU=1.15=U*A/Ch, A can be found by using this formula
printf("\t area is : %.2f m^2\n",A1);

//another way to calculate the area is by using log mean diameter method
LMD=(T2-T1-T3+T4)/log(110/20);
A2=Ch*(T2-T3)/(U*LMD);
printf("\t area is : %.2f m^2\n",A2);
printf("\t there is difference of 1 percent in answers which reflects graph reading inaccuracy.");
// we can see that area calulated is same in above 2 methods.
//End
```

Chapter 4

Analysis of Heat Conduction

Scilab code Exa 4.8 `Comparison of tip temperatures`

```

clear;
clc;

printf("\t Example 4.8\n");

d=0.02; //diameter of alluminium rod,m
k=205; //thermal conductivity of rod,W/(m.K)
l=0.08; //length of rod, m
T1=423; //wall temperature, K
T2=299; //air temperatutre, K
h=120; //convective coefficient, W/(m^2*K)

mL=(h*(l^2)/(k*d/4))^0.5;  // formula for mL=((h*
   Perimeter*l^2)/(k*Area))^0.5
Bi=h*l/k
a1=(cosh(0)+(Bi/mL)*sinh(0))/(cosh(mL)+(Bi/mL)*sinh(
   mL));  //formula for temperature difference T-
   Ttip

Ttip1=T2+a1*(T1-T2); // exact tip temperature
```

```
Tt1=Ttip1-273;
printf("\t the exact tip temperature is : %.2f C\n",Tt1);

a2=(cosh(0)+(Bi/mL)*sinh(0))/(cosh(mL)); //if heat transfer from the tip is not considered
Ttip2=T2+a2*(T1-T2);
Tt2=Ttip2-273;
printf("\t approximate tip temperature is : %.2f C\n",Tt2);
printf("\t thus the insulated tip approximation is adequate for the computation in this case.");
//End
```

Scilab code Exa 4.9 `error calculation in heat flux`

```

clear;
clc;

printf("\t Example 4.9\n");

T1=423; //wall temperature, K
d=0.02; //diameter of alluminium rod,m
k=205; //thermal conductivity of rod,W/(m.K)
l=0.08; //length of rod, m
T2=299; //air temperatutre, K
h=120; //convective coefficient, W/(m^2*K)
mL=0.8656;
a=h*d/(2*k);
mr=mL*(d/(2*l)); // by looking at graph of 1-Qact/Q(no temp.depression) vs. mr*tanh(mL), we can find out the value of Troot. 1-Qact./Q(no temp.depression) = 0.05 so heat flow is reduced by 5 percent
```

```

Troot=T1-(T1-T2)*0.05;
Tr=Troot-273;
printf("\t actual temperature of root is : %.1f C , 
   the correction is modest in this \n",Tr);
//end
```

Scilab code Exa 4.10 `Ressistor temperature calculation`

```


clear;
clc;

printf("\t Example 4.10\n");

T1=308; //air temperature , K
Q=0.1; // heat transferred ,W
k=16; //thermal conductivity of wires , W/(m*K)
d=0.00062; //diameter of wire ,m
Heff=23; //convection coefficient , W/(m^2*K)
//the wires act actn as very long fins connected to 
   ressistor hence tanh(mL)=1

R1=1/(k*Heff*3.14^2*d^(3)/4)^0.5;

Req=(1/R1+1/R1+7.17*(1.33*10^-4)+13*(1.33*10^-4))
   ^-1;   //the 2 thermal ressistances are in 
   parallel to the thermal ressistance for natural 
   convection and thermal radiation from the 
   ressistor's surface found in previous eg.

Tres=T1+Q*Req;
Trs=Tres-273;
printf("\t ressistor temperature is : %.2f C or
```

```
   about 10 C lower than before.\n",Trs);
//end
```

Scilab code Exa 4.11 Heat Loss Calulation

```

clear;
clc;

printf("\t Example 4.11\n");

D1=0.03; // outer diameter, m
T1=358; //hot water temperature, K
t1=0.0008; //thickness of fins, m
D2=0.08; // diameter of fins, m
t2=0.02; // spacing between fins, m
h1=20; // convection coefficient, W/(m^2*K)
h2=15; //convection coefficient with fins, W/(m^2*K)

To=295; //surrounding temperature, K

Q=3.14*D1*h1*(T1-To); // if fins are not added.
Q1=199  //heat loss without fins ,W/m
printf("\t heat trnsferred without fins is : %.0f W/
   m\n",Q1);

// we set wall temp.=water temp..since the wall is
   constantly heated by water, we should not have a
   root temp. depression problem after the fins are
   added.hence by looking at the graph, ml(l/
   Perimeter)^0.5=(h*(D2/2-D1/2)/(125*0.025*t1)) =
   0.306, we obtain n(efficiency)=89 percent

Qfin=Q*(t2-t1)/t2 + 0.89*(2*3.14*(D2^(2)/4-D1^(2)/4)
   )*50*h2*(T1-To)+1.14
```

```
printf("\t heat transferred with fins is : %.0f W/m
   or 4.02 times heat loss without fins.\n", Qfin);
//end
```

Chapter 5

Transient and Multidimensional Heat Conduction

Scilab code Exa 5.2 `temperature and heat calculation`

```

clear;
clc;

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

d1=0.1; // diameter of sphere, m
T1=303; // environment temp.,K
T2=278; // fridge temp., K
h=6; //convection coefficient, W/(m^2*K)
k=0.603; //thermal conductivity,W/(m*K)
a=997.6; // density of water, kg/m^3
c=4180; //heat capacity, J/(kg*K)

F=(k/(a*c))*3600/(d1^2)/4;
a1=0.85 // Biot no.=1/2.01 therefore we read from fig. in upper left hand corner
Tcen=a1*(T1-T2)+T2;  // temperature of the center of apple after 1 hour
```

```
Tc=Tcen-273;
printf(”\t temperature after an hour is : %.1f C\n”,Tc);

a2=(283-T2)/(T1-T2);
F1=1.29 //Bi is still 1/2.01, by looking at the graph we can find time.

t=F1*a*c*0.0025/0.603-2;
printf(”\t time to bring the temp equal to 283k is : %.0f s or 6 hr 12 min\n”,t);
//finally we look up at Bi=1/2.01 and fouling factor is 1.29, for spheres heta removal is 43.67 kJ per apple.
x=43.67;   //heat removal for an apple
X=12*x;   //total heat removal,kJ

printf(”\t total energy removal is :%.0f kJ\n”,X);
//end
```

Scilab code Exa 5.3 temperature fluctuation

```

clear;
clc;

printf(”\t Example 5.3\n”);

d1=0.001;; //diameter of nichrome, m
h=30000; //convection coefficient , W/(mˆ2*K)
T1=373; // wire temperature, K

//heat is being generated in proportion to product of voltage and current, if the boiling action removes heat rapidly enough in comparison with
```

```
the heat capacity of the wire,the surface temperature may well vary.

Bi=h*d1/2/13.8; // biot number comes ot to be 1.09 and value of a= w*d1^(2)/4/a1 comes out to be 27.5. by looking at the chart of cylinders, we find that,      (Tmax-Tav)/(Tav-To)=0.04
 TF=0.04; // temperature fluctuation of 4 percent is not serious and experiment is valid.

 printf("\t the temperature fluctuation is : %.2f this fluctuation is probably not serious.it therefore appears that the experiment is valid.\n", TF);
 //end
```

Scilab code Exa 5.4 `Maximum Time Calculation`

```

clear;
clc;

printf("\t Example 5.4\n");

t=0.003; //half thickness of sword, m
a=1.5*10^-5;

Tmax=t^2/(3.64^2*a); //condition for sword to be in semi infinite region
printf("\t maximum time for sword to be in semi infinite region is : %.3f s\n",Tmax);
printf("\t thus the quench would be felt at the centerline of the sword within only 1/20 s. the thermal diffusivity of clay is smaller than that of steel by a factor of about 30, so the quench
```

```
time of coated steel must continue for over 1s before the temperature of the steel is affected at all, if the clay and sword thickness are comparable.")
//end
```

Scilab code Exa 5.5 `Time Calculation`

```

clear;
clc;

printf("\t Example 5.5\n");
h=100; //convective heat transfer coefficient , W/(m^2*K)
k=0.63; // thermal conductivity ,W/(m*K)
//the short exposure to the flame causes only a very superficial heating ,so we consider the finger to be semi-infinite region.it turns out that the burn threshold of human skin ,Tburn is about 65 C. h=100 W/(m^2*K) , we shall assume that the thermal conductivity of human flesh equals that of its major component - water and that the thermal diffusivity is equal to the known value for beef.

// a=0.963, BE=h*x/k=0(since x=0 at the surface)

// b^2=(h^2)*(0.135*10^-6)*t/(k^2)=0.0034*t. On solving error function by trial and error method, we get the value of t=0.33 sec.

w=(1-0.963)*(%pi)^(0.5)/2;

// thus it would require about 1/3 se to bring the
```

```
skin to burn point.

printf("it would require about 1/3 sec to bring the skin to burn point");

//end
```

Scilab code Exa 5.7 `depth calculation`

```

clear;
clc;

printf("\t Example 5.7\n");

//w=2*3.14 rad/yr , a=w*t=0 at present.first we find the depths at which a=0 curve reaches its local extrema.(we pick the a=0 curve because it) gives the highest temperature at t=o.).tan(o-e)=1 so e=3%pi/4, 7%pi/4....and the first minima occurs where e=3%pi/4=2.356.

b=0.139*10^-6; //thermal diffusivity , m^2/s

x=2.356/(2*3.14/(2*b*365*24*3600))^0.5;  //depth of digging of earth to find the temperature wave

printf("\t depth of digging of earth is :%.3f m, if we dug in the earth, we would find it growing older until it reached a maximum coldness at a depth of about 2.8 m.Farther down, it would begin to warm up again, but nt much. in midwinter, the reverse would be true \n",x);

//end
```

Scilab code Exa 5.8 `temperature calculation`

```

clear;
clc;

printf("Example 5.8\n");

l=0.08; //distance between metal walls ,m
k=0.12; //thermal conductivity of insulating
   material , w/(m*K)
l1=0.04; //length of ribs ,m
l2=0.14; //projected legth of wall ,m
T1=40; // temoerature of 1st wall ,C
T2=0; //temperature of wall , C

//by looking at the configuration plot , there are
   approximately 5.6 isothermal increments and 6.15
   flow channels.

Q=2*(6.15/5.6)*k*(T1-T2); //factor of 2 accounts for
    the fact that there are two halves in the
   section.

T=2.1/5.6*(T1-T2); // by simple propotionality

printf("\t temperature in the middle of of wall , 2
   cm from a rib is : %.0f C\n",T);

//end
```

Scilab code Exa 5.9 `Shape Factor Calculation`

```

clear;
clc;

printf("\t Example 5.9\n");

r=3; // radius ratio of one-quarter section of cylinder

S=%pi/(2*log(r)); // shape factor

printf("\t shape factor is : %.2f \n the quarter cylinder will be pictured for the radius ratio of 3, but for the different sizes.in both the cases it will be 1.43",S);
//end
```

Scilab code Exa 5.11 `Thermal Conductivity`

```

clear;
clc;

printf("\t Example 5.11\n");

Q=14; //steady heat transfer ,W
D=0.06; //diameter of heat source ,m
l=0.3;; // length of source below surface ,m
T=308; //temperature of heat source ,K
T1=294; //temperature of surface ,K

k=(Q/(T-T1))*(1-(D/2)/(D*10))/(4*3.14*D/2)+0.025; // thermal conductivity of soil

printf("\t thermal conductivity is : %.3f W/(m*K)\n"
```

```
,k);

//end
```

Scilab code Exa 5.12 `temperature calculation`

```

clear;
clc;

printf("\t Example 5.12\n");

l=0.04; // length of square rod, m
T1=373; // temerature of rod, K
T2=293; // temperature of coolant ,K
h=800; //convective heat transfer coefficient , W/(m^2*K)

a1=0.93; // ratio of temperature difference for Fo1=0.565, Bi1=0.2105, (x/l)1=0
a2=0.91; // ratio of temperature difference for Fo2=0.565, Bi2=0.2105, (x/l)2=0.5
a=a1*a2; //ratio of temperature difference at the axial line of interest

T=(T1-T2)*a+T2; //temperature on a line 1 cm. from one side and 2 cm. from the adjoining side after 10 sec.
Ta=T-273;

printf("\t temperature is : %.2f C\n",Ta);
//end
```

Scilab code Exa 5.13 `Mean Temperature calculation`

```

clear;
clc;

printf("\t Example 5.13\n");

T1=373; // temperature of iron rod,K
T2=293; // temperature of coolant,K

//Biot no., Bi1=Bi2=0.2105,Fo1=Fo2=0.565
a1=0.10;
a2=0.10;

a=a1+a2*(1-a1);

T=(T1-T2)*(1-a)+T2;    //mean temperature,K
Ta=T-273;
printf("\t mean temperature is : %.1f C\n",Ta);
//end
```

Chapter 6

Laminar and Turbulent Boundary Layers

Scilab code Exa 6.2 `boundary layer thickness calculation`

```

clear;
clc;

printf("\t example 6.2\n");

T1=300; //air temperature ,K
v=1.5; //air velocity , m/s
t=0.5; //thickness , m
u=1.853*10^-5; //dynamic viscosity ,kg/(m*s)
v1=1.566*10^-5; // kinematic viscosity , m^2/s

Rex=v*t/v1; //reynolds no. is low enough to permit the use of laminar flow analysis.

b=4.92*t/(Rex^0.5)*100; // bl thickness , cm

//in this case b/x=1.124/50=0.0225 so laminar flow is valid.
```

```

v2=0.8604*(v1*v/t)^(0.5);
//since v2 grows larger as x grows smaller, the condition v2<u is not satisfied very near the leading edge.

printf("\t boundary layer thickness is : %.3f cm\n",b);
//in this case del/thickness is 0.0225.
x=0.8604*(v1*v/t)^0.5;  //velocity,m/s
y=x/t;
printf("\t since velocity grows larger as thickness grows smaller, the condition x<<u is not satisfied very near the leading edge. therefore the BI approximation themselves breakdown.")
//end
```

Scilab code Exa 6.3 `shear Stress and friction coefficient`

```

clear;
clc;

printf("\t Example 6.3\n");

l=0.5; //total length of surface,m
Cf=0.00607; //overall friction coefficient
tw=1.183*(2.25)*Cf/2; // wall shear, kg/(m*s^2)

a=0.5; //ratio of wall shear  at x=l and average wall shear

//tw(x)=twavg where 0.664/(x^0.5)=1.328/(47,)893, x=1/8 m thus the wall shear stress plummets to twavg one fourth of the way from the leading edge
```

```
and drops only to one half  of twavg in the remaining 75 percent plate.x<600*1.566*10^(-5)/1.5=0.0063 m.

// preceding analysis should be good over almost 99 percent of the 0.5 m length of the surface.

printf("\t overall friction coefficient is : %f\n",Cf);
printf("\t wall shear is :%f  kg/(m*s^2)\n",tw);
printf("\t the preceding analysis should be good over almost 99 percent of the 0.5m length of the surface.")
//end
```

Scilab code Exa 6.4 Average Heat Flux

```

clear;
clc;

printf("\t Example 6.4\n");

l=0.06; //length of heater, m
p=15; // pressure of heater, atm
T1=440; //temperature of heater, K
v=2; //free stream velocity,m/s
T2=460; // constant temperature of heater, K

T3=450; //mean temperature of heater, K

q=2*(0.332)*(0.674/l)*(v*l/(1.72*10^-7))^(0.5)*(T2-T1)/1000; // formula for heat flux is q=2*(0.664)*k/l*(Rel^0.5)*(T2-T1)

```

```
printf("\t heat flux is : %.0f kW/m^2\n",q);
//end
```

Scilab code Exa 6.5 Average Heat Transfer coefficient

```

clear;
clc;

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

T1=293; //air temperature ,K
v=15; //air velocity ,m/s
T2=383; // temperature of plate ,K
l=0.5; // length of plate ,m
w=0.5; //width of plate ,m

Pr=0.707; // prandtl no.
Rel=v*l/(0.0000194); //reynplds no.
Nul=0.664*(Rel)^0.5*Pr^(1/3); // nusset no.

h1=367.8*(0.02885)/l; // average convection coefficient , W/(m^2*K)
Q=h1*l^(2)*(T2-T1); // heat transferred ,W

h2=h1/2 // convection coefficient at trailing , W/(m^2*K)
a1=4.92*l/(Rel)^0.5*1000 // hydrodynamic boundary layer ,m

a2=a1/(Pr)^(1/3); //thermal boundary layer ,mm

printf("\t average heat trensfer coefficient is : %.1f W/m^2/K\n",h1);
printf("\t total heat transferred is  %.0f W\n",Q);
```

```
printf("\t convection coefficient at trailing is : %.1fW/(m^2*K)\n",h2);
printf("\t hydrodynamic boundary layer is : %.2f m\n",a1);
printf("\t thermal boundary layer is : %.2f mm\n",a2);

// end
```

Scilab code Exa 6.6 `Average temperature calculation`

```

clear;
clc;

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

T1=288; // air temperature ,K
v=1.8; // air velocity ,m/s
l=0.6; // length of panel , m
Q=420; // power per unit area , m^2
T2=378; // maximum temperature of surface , K

T3=Q*l/(0.0278)/(0.453*(l*v/(1.794*10^-5))^(0.50)*(0.709)^(1/3)); //maximum temperature difference

Twmax=T1+T3;  //Twmax comes out to be 106.5 C, this is very close to 105 C,if 105 is at all conservative , Q = 420 should be safe.

T4=0.453/0.6795*T3; //average temperature difference ,K

Twavg=T1+T4; //average wall temperature ,K
Twa=Twavg-273;
```

```
printf("\t average wall temperature is : %.0f C\n",
   Twa);
//end
```

Scilab code Exa 6.8 `Drag Force calculation`

```

clear;
clc;

printf("\t Example 6.3\n");

v=15; //air velocity ,m/s
T2=383; // temperature of plate ,K
l=0.5; // length of plate ,m
w=0.5; //width of plate ,m

Pr=0.707; // prandtl no.
Rel=v*l/(0.0000194); //reynplds no.
Nul=0.664*(Rel)^0.5*Pr^(1/3); // nusset no.

Cf=2*Nul/(Rel*Pr^(1/3)); //friction coefficient

s=Cf*0.5*1.05*225; //drag shear , kg/(m*s^2)
f=s*0.5^2-0.000024;  //drag force , kg/(m*s^2)

printf("\t drag force on heat transfer surface is :
   %f N or 0.23 oz.\n",f);

//end
```

Scilab code Exa 6.9 `heat transfer coefficient calculation`

```

clear;
clc;

printf("\t Example 6.3\n");

T1=297; // river water temp.,K
T2=283; // ocean water temp., K
n=5; // no. of knots
k=0.5927; // thermal conductivity,W/(m*K)
a=998.8; //density of water, kg/m^3
Cp=4187; // heat capacity, J/kg/K
Pr=7.66;
x=1; //distance from forward edge,m

T3=(T1+T2)/2; // avg. temp.,K
v=1.085*10^-6; // kinematic viscosity, m^2/s

u=2.572; //velocity of knot,m/s

Rex=u/v //reynolds no.
Cf(x)=0.455/(log(0.06*Rex))^2 // friction coefficient

h=k/x*0.032*(Rex)^(0.8)*Pr^(0.43); // heat transfer coefficient,W/(m^2*K)
printf("\t friction coefficient is : %f\n",Cf);
printf("\t convective heat transfer coefficient at a distance of 1 m fom the forward edge is :%.0f W/(m^2*K)\n",h);
h1=a*Cp*u*Cf/2/(1+12.8*(7.66^0.68-1)*(Cf/2)^0.5); //heat transfer coefficient,W/(m^2*K)
printf("\t heat transfer coefficient by another method is :%.0f W/(m^2*K)\n",h1);
printf("\t the two values of h differ by about 18 percent, which is within the uncertainity.");

//end
```

Scilab code Exa 6.10 `Average temperature calculation`

```

clear;
clc;

printf("\t Example 6.3\n");

l=2; // length of plate,m
p=1000; // power density,W/m^2
u=10; // air velocity,m/s
T1=290; // wind tunnel temp.,K
p2=1; // pressure,atm
Re = 400000; // reynolds no.

v=1.578*10^-5; // kinematic viscosity, m^2/s
k=0.02623; // thermal conductivity,W/(m*K)
Pr=0.713; // prandtl no.
Rel=u*l/v; //reynolds no. at 10 m/s

Nul=1845; // nusselt no.

h=Nul*k/l; //convection coefficient,W/(m^2*K)

Tavg=T1+p/h;

printf("\t average temperature of plate is : %.0f K\n",Tavg);
//to take better account of the transition region, we can use churchill eqn.
x=Rel*Pr^(2/3)/(1+(0.0468/Pr)^2/3)^0.5;
x1=1.875*x*Re;
Nul1=0.45+0.6774*x^(0.5)*(1+((x/12500)^3/5/(1+(x1/x)^3.5)^0.4)^0.5);
```

```

H=Nul1*k/l;    //convection coefficient ,W/(m^2*K)
Tw=290+1000/H-77.14;    //average temperature of plate ,K
printf("\t average temperature of plate is :%.0f K , thus in this case, the average heat transfer coefficient is 33 percent higher when the transition regime is included.\n",Tw);
//end
```

Chapter 7

Forced Convection in Configuration Systems

Scilab code Exa 7.1 `depth calculation`

```

clear ;
clc;

printf("\t Example 7.1\n");
D=0.001;     //diameter of tube ,m
T1=293;       // temperature of cold water , K
T2=347;      // temperature of hot water , K
T3=320;      //operating temperature of hot water , K
Q=6000;     //heat flux ,W/m^2
v=0.2 ;    //speed of water ,m/s
k=0.6367;     //thermal conductivity ,W/(m*K)
v1=1.541*10^-7;    // molecular diffusivity , m^2/s
v2=0.556*10^-6;     //molecular diffusivity , m^2/s

Re=D*v/v2;     //reynolds no.

L=D*(54-11/48*Q*D/k)*v*k/(4*Q*v1);     //length that
   is down the tube for water reach to 74 C at its
```

```
hottest point ,m
printf("length that is down the tube for water reach to 74 C at its hottest point is : %.3f m ,while we did not evaluate the thermal entry length here , it may be shown to be much, much less than 1785 diametres.\n",L);
//end
```

Scilab code Exa 7.2 `power input and wall temperature`

```

clear;
clc;

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

T1 = 300; // air temp.,K
T2=313; // final air temp.,K
v=2; // air velocity,m/s
D=0.01; // inner diameter of pipe,m
l=0.2; // length surrounded by heater
Red=v*D/(16.4*10^-6); // reynolds no.
Pr=0.711; // prandtl no.
G=Red*Pr*D/l; // graetz no.

Q=1.159*1004*v*(T2-T1)*(1/80); // power input, W/m^2
printf("\t power input is : %.0f  W/m^2\n",Q);

Tex=T2+Q*D/(5.05*0.0266) // wall temp. at the exit,K
Tex1=Tex-273.1;

printf("\t wall temp. at the exit is: %.1f C\n",Tex1);

//end
```

Scilab code Exa 7.3 `friction factor calculation`

```

clear;
clc;

printf("\t Example 7.3\n");

m=21.5;  //mass flow rate, kg/s

D=0.12;   //diameter of pipe, m
T1=363;   //pipe temperature,K
T2=323;   //bulk temp. of fluid,K
a=977;  //density, kg/m^3
u=m/(a*3.14*(D/2)^2);; //average velocity,m/s
Re=u*D/(4.07*10^-7);  //reynolds no.
Uw=3.1*10^-4;         // wall side viscosity,N*s/m^2
Ub=5.38*10^-4;    //bulk viscosity, N*s/m^2

Pr=2.47;  //prandtl no.
f=1/(1.82/2.303*log(Re)-1.64)^2;      // formula for friction factor for smooth pipes

Nu=(f/8*Re*Pr)/(1.07+12.7*(f/8)^(0.5)*(Pr^(2/3)-1));    //formula for nusselt no.in fully developed flow in smooth pipes

h=Nu*0.661/D    // convective heat transfer coefficient,W/(m^2)/K
h1=8907;   //convective heat transfer coefficient,W/(m^2)/K

//corrected friction factor = friction factor at bulk temp.*K where K=(7-u1/u2)/6 for wall temp.>
```

```
bulk temp.

f1=f*((7-Ub/Uw)/6);     // corrected friction factor
F=0.0122;   //corrected friction factor

printf("\t correlation friction factor. is : %.4f\n",F);
printf("\t convection heat transfer coefficient is : %.0f W/(m^2)/K \n",h1);

// end
```

Scilab code Exa 7.4 `friction factor calculation`

```

clear;
clc;

printf("\t Example 7.3\n");

m=21.5;   //mass flow rate, kg/s
e=260*10^-6;    //wall roughness,m

D=0.12;    //diameter of pipe, m
T1=363;    //pipe temperature,K
T2=323;    //bulk temp. of fluid,K
a=977;   //density, kg/m^3
u=m/(a*3.14*(D/2)^2); //average velocity,m/s
Re=u*D/(4.07*10^-7);   //reynolds no.
Uw=3.1*10^-4;         // wall side viscosity,N*s/m^2
Ub=5.38*10^-4;    //bulk viscosity, N*s/m^2

Pr=2.47;   //prandtl no.

f=1/(1.8/2.303*log(6.9/Re+(e/D/3.7)^1.11))^2;     //
```

```
friction factor from haaland equation.
Re1=Re*e/D*(f/8)^0.5;    //roughness reynols no.

Nu=(f/8)*Re*Pr/(1+(f/8)^0.5*(4.5*Re1^(0.2)*Pr^(0.5)-8.48));    //correlation for local nusselt no.

h=Nu*0.661/D/1000;   //convection heat transfer coefficient, kW/(m^2*K)
printf("\t correlation friction factor is :%.5f\n",f);
printf("\t convection heat transfer coefficient is :%.1f kw/(m^2*K)\n",h);

printf("\t in this case wall roughness causes a factor of 1.8 increase in h and a factor of 2 increase in f and the pumping power.we have omitted the variable properties hre as they were developed for smooth walled pipes.")
//end
```

Scilab code Exa 7.5 `temperature calculation`

```

clear;
clc;

printf("\t Example 7.5\n");

T1 = 293; // air temp.,K
D=0.01; // inner diameter of pipe,m
v=0.7;// air velocity,m/s
T2=333; //pipe wall temp.,K
t=0.25; // distance down the stream
Re=v*D/(1.66*10^-5); // reynolds no.

```

```
// the flow is therefore laminar, to account for the thermal entry region, we compute the graetz no.

Gz=Re*(0.709)*D/t;  // graetz no.
Nu=4.32 // nusselt no., Nu=3.657+(0.0668*Gz^(1/3)/(0.04+Gz^(-2/3)))

h=3.657*(0.0268)/D; // average  convective heat transfer coefficient. W/(m^2*K)

a=1-exp((-h/(1.14*1007*v))*(4*t)/D);   // (Tb-T1)/(T2-T1)=a (suppose)

Tb=a*(T2-T1)+T1;    // temperature 0.25 m farther down stream.
Tb1=Tb-270.6;

printf("\t temperature 0.25 m farther down stream is :%.1f C\n",Tb1)
//end
```

Scilab code Exa 7.6 `change in bulk temperature`

```

clear;
clc;

printf("\t Example 7.6\n");

Tbin=290;      //inlet bulk temp.,K
v=1;           //speed of air, m/s
a=0.09;        //area of steel,m^2
l=15;          //length of duct running outdoors through awarm air,m
To=310;        //temp. of warm air,K
```

```
h=5;            //heat transfer coefficient due to natural convection and thermal radiation.
Dh=0.3;         //hydraulic diameter,m
Re=v*Dh/(1.578*10^-5);    //reynolds no.at Tbin
Pr=0.713;    //prandtl no.

f=1/(1.82/2.303*log(Re)-1.64)^2;      // formula for friction factor for smooth pipes

Nu=(f/8*Re*Pr)/(1.07+12.7*(f/8)^(0.5)*(Pr^(2/3)-1));    //formula for nusselt no.in fully developed flow in smooth pipes


h=Nu*0.02623/Dh;     // convective heat transfer coefficient,W/(m^2)/K
//the remaining problem is to find the bulk temperature change.the thin metal duct wall offers little thermal ressistance, but convection ressistance outside the duct must be considered.

U=(1/4.371+1/5)^-1;     //U=1/Ain*(1/(h*A)in+1/(h*A)out)^-1


Tbout=(To-Tbin)*(1-exp(-U*4*l/(1.217*v*1007*Dh)))+Tbin;     //outlet bulk temp., K
Tbt1=Tbout-273;

printf("\t outside bulk temp. change is : %.1f C\n",Tbt1);
//end
```

Scilab code Exa 7.7 `Air speed calculation`

```
clear;
clc;

printf("\t Example 7.7\n");

D=0.0001; //  diameter of heater, m
T1 = 293; // air temp.,K
T2=313; //heater temp.,K
p=17.8;  //dissipating heat, W/m

h=p/(3.14*D*(T2-T1)); // average  convective heat
    transfer coefficient. W/(m^2*K)
Nu=h*D/0.0264;  //nusselt no., Nu=h*D/thermal
    conductivity

Pr=0.71; //prandtl  no.

Re=((Nu-0.3)*(1+(0.4/Pr)^(2/3))^0.25/(0.62*Pr^(1/3))
    )^2; //reynolds no.

u=1.596*10^(-5)/(D)*Re+0.2;  //air velocity, m/s

printf("\t air velocity is : %.1f m/s\n",u);
printf("\t the data scatter in Red is quite small
    less than 10 percent, it would appear. therefore,
     this method can be used to measure local
    velocities with good accuracy.if the device is
    calliberated, its accuracy is improved further,
    such an air speed indicator is called a hot wire
    anemometer.")

//end
```

Chapter 8

natural Convection in single phase fluids and during film codensation

Scilab code Exa 8.1 `heat transfer coefficient and heat flux calculation`

```

clear;
clc;

printf("\t Example 8.1\n");

T1=313;      //fluid temp.,K
T2=287;      //air temp.,K
H=0.4;       //height of sides,m
Pr=0.711;    //prandtl no.


 b=1/T2;     // b=1/v*d(R*T/p)/dt=1/To
    characterisation constant of thermal expansion
    of solid, K^-1
 RaL=9.8*b*(T1-T2)*H^3/((1.566*10^-5)*(2.203*10^-5))
    ;    //Rayleigh no.
```

```

Nu=0.678*RaL^(0.25)*(Pr/(0.952+Pr))^(1/4);    // nusselt no.
h=Nu*0.02614/H     // average heat transfer coefficient , W/m^2/K

q=h*(T1-T2)        // average heat transfer ,W/m^2
c=3.936*((0.952+Pr)/Pr^2)^(1/4)*(1/(RaL/Pr)^0.25);     //boundary layer thickness.,m
printf("\t  average heat transfer coefficient is : %.2f W/m^2/K\n",h);
printf("\t average heat transfer is : %.1f W/m^2\n",q);
printf("\t boundary layer thickness is : %.3f m\n",c);

printf("\t thus the BL thickness at the end of the plate is only 4 percent of the height , or 1.72 cm thick.this is thicker thsan typical forced convection BL but it is still reasonably thin.")

 //end
```

Scilab code Exa 8.3 `heat transfer coefficient varification`

```

clear;
clc;

printf("\t Example 8.3\n");

T1=323;      //wall temp.,K
T2=293;      //air temp.,K
H=0.3;       //height of wall , m
v1=16.45*10^-6;    // molecular diffusivity , m^2/s
```

```
b=1/T2;     // b=1/v*d(R*T/p)/dt=1/To characterisation constant of thermal expansion of solid , K^-1
v2=2.318*10^-5;     //molecular diffusivity , m^2/s
Pr=0.71;    //prandtl no.

 Ral=9.8*b*(T1-T2)*H^3/((1.566*10^-5)*(2.203*10^-5));     // Rayleigh no.
 Nu=0.678*Ral^(0.25)*(Pr/(0.952+Pr))^(1/4);     // nusselt no.
 h=Nu*0.0267/H      // average heat transfer coefficient , W/m^2/K

 Nu1=0.68+0.67*((Ral)^(1/4)/(1+(0.492/Pr)^(9/16))^(4/9));     //churchill correlation

 h1=Nu1*(0.0267/0.3)-.11;      //average heat transfer coefficient , W/m^2/K


printf("\t correlation average heat transfer coefficient is :%.2f W/m^2/K\n",h1)
printf("\t the prediction is therefore within 5 percent of corelation .we should use the latter result in preference to the theoritical one, although the difference is slight.")
  //end
```

Scilab code Exa 8.4 `Heat flux variation`

```

clear;
clc;

printf("\t Example 8.4\n");
```

```
T1=400;      //hot oil temp.,K
D=0.005;           //diameter of line carrying oil, m
T2=300;     //temp. of air around the tube,K
Tav=350;     //average BI temp.,K

//we evaluate properties at this temp. and write g as ge*(g-level), where ge is g at the earth surface and the g-level is the fraction of ge in the space vehicle.
b=1/T2;     // b=1/v*d(R*T/p)/dt=1/To characterisation constant of thermal expansion of solid, K^-1

v1=2.062*10^-5;    // molecular diffusivity, m^2/s
v2=2.92*10^-5;     //molecular diffusivity, m^2/s
Pr=0.706;    //prandtl no.

g=[10^-6 10^-5 10^-4 10^-2];
i=1;
while(i<5)
Ral=(9.8*b*((T1-T2))*(D^(3))/(v1*v2))*g(i);     //Rayleigh no.
Nu(i)=(0.6+0.387*(Ral/(1+(0.559/Pr)^(9/16))^(16/9))^(1/6))^2;
//Nu(i)=(0.6+0.387*((Ral)/(1+(0.559/Pr)^(9/16))^(16/9))^1/6)^2;     //churchill correlation.
printf("\t Nusselt no. are : %.3f\n",Nu(i));
h(i)=Nu(i)*0.0297/D;     // convective heat transfer coefficient,W/(m^2*K)
printf("\t convective heat transfer coefficient are : %.2fW/(m^2*K)\n",h(i));
Q(i)=%pi*D*h(i)*(T1-T2);     //heat transfer,W/m
printf("\t heat transfer is :%.2fW/m of tube\n",Q(i));
i=i+1;
end
```

```
//end
```

Scilab code Exa 8.5 `Average surface temperature calculation`

```

clear ;
clc;

printf("\t Example 8.5\n");

T2=300;      //air temp.,K
P=15;        //delivered power,W
D=0.17;        //diameter of heater,m
v1=1.566*10^-5;    // molecular diffusivity , m^2/s
b=1/T2;     // b=1/v*d(R*T/p)/dt=1/To characterisation constant of thermal expansion of solid , K^-1
Pr=0.71;    //prandtl no.
v2=2.203*10^-5;     //molecular diffusivity , m^2/s
v3=3.231*10^-5;     //molecular diffusivity at a b except at 365 K., m^2/s
v4=2.277*10^-5;     //molecular diffusivity at a b except at 365 K., m^2/s
k1=0.02614;         //thermal conductivity
k2=0.0314;         //thermal conductivity

//we have no formula for this situation , so the problem calls for some guesswork.following the lead of churchill and chau, we replace RaD with RaD1/NuD in eq.
//(NuD)^(6/5)=0.82*(RaD1)^(1/5)*Pr^0.034

delT=1.18*P/(3.14*D^(2)/4)*(D/k1)/((9.8*b*661*D^(4)/(0.02164*v1*v2))^(1/6)*Pr^(0.028));

```

```
//in the preceding computation, all the properties were evaluated at T2.mow we must return the calculation,reevaluating all properties except b at 365 K.

delTc=1.18*661*(D/k2)/((9.8*b*661*D^(4)/(k2*v3*v4))^(1/6)*(0.99));

TS=T2+delTc;
TS1=TS-271.54

printf("\t average surface temp. is :%.0f K\n",TS1);

printf("\t that is rather hot.obviously, the cooling process is quite ineffective in this case.")
//end
```

Scilab code Exa 8.6 `heat transfer calculation`

```

clear;
clc;

printf("\t Example 8.6\n");

T2=363;         // temp. of strip ,K
T1=373;         //saturated temp.,K
H=0.3;          // height of strip ,m
Pr=1.86;    //prandtl no.
Hfg=2257;     //latent heat. kj/kg
ja=4.211*10/Hfg;    //jakob no.
a1=961.9;         //density of water,kg/m^3
a2=0.6;           //density of air,kg/m^3
k=0.677;          //thermal conductivity ,W/(m*K)

```

```
Hfg1=Hfg*(1+(0.683-0.228/Pr)*ja);    //corrected latent heat,kj/kg

delta=(4*k*(T1-T2)*(2.99*10^(-4))*0.3/(a1*(a1-a2)*9.806*Hfg1*1000))^(0.25)*1000;

Nul=4/3*H/delta;    //average nusselt no.
q=Nul*k*(T1-T2)/H;    // heat flow on an area about half the size of a desktop,W/m^2
Q=q*H;    //overall heat transfer per meter,kW/m

m=Q/(Hfg1);     //mass rate of condensation per meter,kg/(m*s)

printf("\t overall heat transfer per meter is :%.1f kW/m^2\n",Q);
printf("\t  film thickness at the bottom is :%.3f mm\n",delta);
printf("\t mass rate of condensation per meter. is : %.4f kg/(m*s)\n",m);

//end
```

Chapter 9

Heat transfer in boiling and other phase configurations

Scilab code Exa 9.1 `Size estimation`

```

clear;
clc;

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

T2=363;         // temp. of strip ,K
T1=373;         //saturated temp.,K
p=1.013*10^5;                //pressure of water,N/m^2
psat=1.203*10^5;    //saturated pressure at 108 C,N/m
   ^2
psat1=1.769*10^5;    //saturated pressure at 116 C,N/
   m^2
a=57.36*10^-3;          //surface tension , N/mat Tsat
   =108 C
a1=55.78*10^-3;          //surface tension , N/mat Tsat
   =116 C
Rb=2*a/(psat-p)*1000;          //bulk radius at 108 C,
    mm
```

```
Rb1=2*a1/(psat1-p)*1000;       // bulk radius at 116 C, mm

printf("\t bulk radius at 108 C is :%.3f mm\n",Rb);
printf("\t bulk radius at 116 C is :%.4f mm\n",Rb1);

printf("\t this means that the active nucleation sites would be holes with diameters very roughly on the order magnitude of 0.005 mm atleast on the heater .that is within the ransge of roughness of commercially finished surfaces. ")
//end
```

Scilab code Exa 9.2 `surface factor calculation`

```

clear ;
clc;

printf("\t Example 9.2\n");

q=800;      //power delivered per unit area,KW/m^2
T1=373;          //saturated temp.of water, K
delT=22;         // temp. difference,K
Cp=4.22;         //heat capacity of water,kj/(kg*K)
Pr=1.75;         //prandtl no.
a=958;           //desity difference,kg/m^3
s=0.0589;        //surface tension,kg/s^2
Hfg=2257;        //latent heat,kj/kg

//by using rohensow correlation applied data for water boiling on 0.61 mm diameter platinum wire

Csf=(3.1*10^-7*(delT)^3/(q))^(1/3);     //surface correction factor of the heater surface
```

```

printf("\t surface correction factor of the heater surface is : %.3f, this value compares favorably with Csf for a platinum or copper surface under water.\n",Csf);
//end
```

Scilab code Exa 9.3 `steam velocity estimation`

```

clear;
clc;

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

p=1.013*10^5;        //pressure of water,N/m^2
D=0.1;               //inside diameter,m
l=0.04;              //wavelength,m
a=0.0589;            //surface tension,N/m
b=0.577;             //density of gas, kg/m^3

u=(2*%pi*a/(b*l))^(0.5);          //the flow will be helmholtz stable until the steam velocity reaches this value.

printf("\t steam velocity required to destablize the liquid flow is : %.1f m/s ,beyond that, the liquid will form whitecaps and be blown back upward.\n",u);

//end
```

Scilab code Exa 9.4 `maximum spacing calculation`

```

clear;
clc;

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

a=13600;         //desity difference,kg/m^3
s=0.487;        //surface tension,kg/s^2

L=2*%pi*(3^0.5)*(s/(9.8*a))^0.5*100;     //spacing wavelength,cm

printf("\t maximum spacing is : %.1f cm\n",L);
printf("\t actually this spacing would give the maximum rate of collapse.it can be shown that collapse would begin at 1/3^0.5 times this value or at 1.2 cm.")
//end
```

Scilab code Exa 9.5 `peak heat flux calculation`

```

clear;
clc;

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

T1=373;           //saturated temp.of water, K
a=957.6;            //desity difference,kg/m^3
s=0.0589;         //surface tension,kg/s^2
Hfg=2257*1000;        //latent heat,J/kg
a2=0.597;           //density of gas, kg/m^3

Qmax=0.149*a2^0.5*Hfg*(9.8*a*s)^0.25/1000000;

```

```
printf("\t peak heat flux is : %.2f MW/m^2 ,from
   figure it can be shown that qmax =1.16 MW/m^2,
   which is less by only about 8 percent.\n",Qmax);

//end
```

Scilab code Exa 9.6 `maximum heat flux`

```

clear ;
clc;

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

T1=628;          //saturated temp.of water, K
a=13996;           //desity difference,kg/m^3
s=0.418;         //surface tension,kg/s^2
Hfg=2925*1000;        //latent heat,J/kg
a2=4;            //density of mercury, kg/m^3

Qmax=0.149*a2^0.5*Hfg*(9.8*a*s)^0.25/10^7-.015;

printf("\t peak heat flux is : %.3f MW/m^2\n",Qmax);
printf("\t the result is very close to that for
   water,the increase in density and surface tension
    have not been compensated by amuch lower latent
   heat.")

//end
```

Scilab code Exa 9.7 `heat removal rate calculation`

```

```

```
clear ;
clc;

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

T1=373;          //saturated temp.of water, K
a=958 ;          //desity difference,kg/m^3
s=0.0589;        //surface tension,kg/s^2
Hfg=2257*1000;        //latent heat,J/kg
a2=0.597;            //density of gas, kg/m^3
A=400*10^-4;          //area of mettalic body,m^2
V=0.0006;            //volume of body, m^3

Qmax=(0.131*a2^0.5*Hfg*(9.8*a*s)^0.25)*0.9*A/1000 ;        //large rate of energy removal, KW        as the cooling process progresses,it goes through the boiling curve from film boiling,through qmin, up the transitional boiling regime,through qmax and down the3 nucleate boiling curve.

//R=V/A*(9.8*a/s)^0.5       since this value comes out to be 6.0, which is larger than the specified lower bound of about 4.

printf("\t the heat flow is  : %.1f KW\n",Qmax);
//to complete the calculation, it is necessary to check whether or not rate is large enough to justify the use.

R=V/A*(9.8*958/0.0589)^0.5;    //the most rapid rate of heat removal during the quench
printf("\t the most rapid rate of heat removal during the quench is : %.0f , this is larger than the specified lower bound of about 4.\n",R);
//end
```

Scilab code Exa 9.8 `minimum heat flux`

```

clear ;
clc;

printf(”\t Example 9.8\n”);

T1=373;          //saturated temp.of water, K
a=957.6;           //desity difference,kg/mˆ3
s=0.0589;        //surface tension,kg/sˆ2
Hfg=2257*1000;        //latent heat,J/kg
a2=0.597;           //density of gas, kg/mˆ3

Qmin=0.09*a2*Hfg*(9.8*a*s/(959^2))^0.25+2;

printf(”\t peak heat flux is : %.0f W/mˆ2  ,from the figure, we read 20000 W/mˆ2, which is the same, within the accuracy of graph.\n”,Qmin);

//end
```

Scilab code Exa 9.9 `wall temperature calculation`

```

clear;
clc;

printf(”\t Example 9.9\n”);

T1=480;         //bulk temp.of water, K
```

```
m=0.6;          //mass flow rate of saturated water, kg/s
D=0.05;         //diameter of vertical tube,m
p=184000;       //heating rate f tube, W/m^2
A=0.001964;      //area of the pipe,m^2
Pr=0.892;        //prandtl no.
x=0.2;           //quality
a1=9.014;         //density of gas,kg/m^3
a2=856.5         //density of water, kg/m^3
Hfg=1913*1000;     //latent heat,J/kg

G=m/A;         //superficial mass flux
Relo=G*D/(1.297*10^-4);     //reynolds no. for liquid only.
f=1/(1.82/2.303*log(Relo)-1.64)^2;      // formula for friction factor for smooth pipes

Nu=(f/8*Relo*Pr)/(1.07+12.7*(f/8)^(0.5)*(Pr^(2/3)-1));   //formula for nusselt no.in fully developed flow in smooth pipes

hlo=0.659*Nu/D;         //heat transfer coefficient,w/(m^2*K)

Co=((1-x)/x)^0.8*(a1/a2)^0.5;         // Convection no.
Bo=p/(G*Hfg);             // boiling no.

Hfg1=(1-x)^0.8*(0.6683*Co^(-0.2)+1058*Bo^0.7)*hlo;             //heat transfer coefficient for nucleate boiling dominant, w/(m^2*K)
Hfg2=(1-x)^0.8*(1.136*Co^(-0.9)+667.2*Bo^0.7)*hlo;                   //heat transfer coefficient for connective boiling dominant, w/(m^2*K)

//since the second value is larger,we will use it.

Tw=T1+p/Hfg2;             //wall temperature ,K
```

```
Tw1=Tw-273;
printf("\t wall temperature  is : %.0f C\n",Tw1);
//end
```

Chapter 10

thermal radiation heat transfer

Scilab code Exa 10.1 `net heat transfer calculation`

```


clear;
clc;

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

T1=2273;          //temp. of liquid air,K
T2=303;           //temp. of room,K
T3=973;          //temp. of shield,K
D1=0.003;          //diameter of crucible,m
D2=0.05;          //diameter of shield,m
theta1=330;       //surrounding angle of jet, degree
theta2=30          // angle of slit, degree
Fjr=theta2/360;         //fraction of energy of view
   of jet occupied by room
Fjs=theta1/360 ;         //fraction of energy of view
   of jet occupied by shield

Qnjr=%pi*D1*Fjr*5.67*10^-8*(T1^4-T2^4);     //net
   heat transfer from jet to room,W/m
```

```

Qnjs=%pi*D1*Fjs*5.67*10^-8*(T1^4-T3^4);     //net heat transfer from jet to shield ,W/m

//to find the radiation from the inside of the shield to the room, we need Fshield-room.since any radiation passing out of the slit goes to the room,we can find this view factor equating view factors to the room with view factors to the slit.

Fsj=%pi*D1/0.01309*Fjr;     //fraction of energy of view of slit occupied by jet
Fss=1-Fsj;          //fraction of energy of view of slit  occupied by shield.
Fsr=0.01309*Fss/(%pi*D2*Fjs);     //fraction of energy of view of shield occupied by room

Qnsr=%pi*D2*Fjs*5.67*10^-8*Fsr*(T3^4-T2^4);     //net heat transfer from shield to room, W/m

printf("\t  heat transfer from jet to room through the slit is :%.0f W/m\n",Qnjr);

printf("\t  heat transfer from the jet to shield is :%.0f W/m\n",Qnjs);

printf("\t  heat transfer from inside of shield to the room is :%.0f W/m\n",Qnsr);

printf("\t both the jet and the inside of the shield have relatively small view factors to the room, so that comparatively little heat is lost through the silt.");
//end
```

Scilab code Exa 10.2 `net heat transfer calculation`

```

clear ;
clc;

printf("\t Example 10.2\n");
T1=373;         //temp. of shield ,K
T2=1473;         //temp of heater ,K
h=0.2 ;        //height of disc heater ,m
r1=0.05;       //smaller radius of heater ,m
r2=0.1;         //larger radius of heater ,m
R1=r1/h ;         //factors necessary for finding view
    factor
R2=r2/h ;         //factors necessary for finding view
    factor
X=1+(1+R2^2)/R1^2;      //factors necessary for
   finding view factor

Fht=0.5*(X-(X^2-4*(R2^2/R1^2))^0.5);         //view
   factor
Fhs=1-Fht;     //view factor of heater  occupied by
   shield
Qnhs=%pi*r2^2*Fhs*5.67*10^-8*(T2^4-T1^4)/4+1;

printf("\t net heat transfer from the heater to
   shield is : %.0f W\n",Qnhs);
//end
```

Scilab code Exa 10.3 `view factor calculation`

```

```

```
clear;
clc;

printf("\t Example 10.3\n");
h=0.2 ;        //height of disc heater,m
r1=0.05;       //smaller radius of heater,m
r2=0.1;         //larger radius of heater,m
Fhs=0.808;          //view factor of heater occupied by shield

As=%pi*(r1+r2)*(h^2+(r2-r1)^2)^0.5;    //area of frustrum shaped shield,m^2
Ah=%pi/4*r2^2;             //heater area,m^2

Fsh=Ah/As*Fhs;          //view factor of shield occupied by heater

printf("view factor of shield occupied by heater is :%.4f\n",Fsh);
//end
```

Scilab code Exa 10.4 `view factor calculation`

```

clear;
clc;

printf("\t Example 10.4\n");

F1342=0.245;         //view factor of 1and 3 occupied by 2 and 4
F14=0.2;             //view factor of 1 occupied by 4

F12=F1342-F14;         //view factor of 1 occupied by 2
```

```
printf("\t view factor of 1 occupied by 2  is :%.3f\n",F12);
//end
```

Scilab code Exa 10.8 heat gain rate and temperature of the shield

```

clear;
clc;

printf("\t Example 10.8\n");
T1=80;          //temp.of liquid nitrgen ,K
T2=230;          //temp of chamber walls ,K
D1=0.00635;       //outer diameter of steel , m
D2=0.0127;         //diameter of 2nd steel tube , m
e1=0.2 ;           //emissivity 0f steel
x=poly([0],'x');

//the nitrogen coolant will hold the surface of the line at essentially 80 K, since the thermal ressistance of tube wall and int. convection or boiling process are small.

Qgain=%pi*D1*e1*5.67*10^-8*(T2^4-T1^4);    // net heat gain of line per unit length ,W/m
 //with the shield , assuming that the chamber area is large compared to the shielded line.

 Qgain1=%pi*D1*5.67*10^-8*(T2^4-T1^4)/(((1-e1)/e1+1)+D1/D2*(2*(1-e1)/e1+1));        //net heat gain with shield ,W/m

 s=(Qgain-Qgain1)/Qgain*100;     //rate of heat gain reducton in percentage

```

```
x=roots(%pi*D2*e1*5.67*10^-8*(T2^4-x^4)-Qgain1);


printf("\t net heat gain of line per unit length is :%.3f W/m\n",Qgain);
printf("\t rate of heat gain reducton is :%.0f percent \n",s);
printf("\t temp. of the shield is : %.0f C\n",x(4));

//end
```

Scilab code Exa 10.9 `net heat transfer calculation`

```

clear;
clc;

printf("\t Example 10.9\n");
T1=250 ;          //temp.of surrounding,K
l1=1;          //width of strips, m
l2=2.4;          //distance between strips,m
F12=0.2;     //view factor of 1 occupied by 2.

A=[1 -0.14;-1 10] ;     //matrix representation for solving the linear equations, for black surroundings
B=[559.6;3182.5];          //matrix representation for solving the linear equations.

X=inv(A)*B;
A=[1 -0.14;-1 10];     //matrix representation for solving the linear equations, for black surroundings
B=[559.6;3182.5];          //matrix representation for
```

```
solving the linear equations.

X=inv(A)*B;

Qn12=(X(1)-X(2))/(1/(0.9975*F12));         //net heat flow from 1 to 2 for  black surroundings.
//since each strip loses heat to the surrounding, Qnet1, Qnet2 and Qnet1-2 are different.
// three equations will be
//(1451-B1)/2.33 = (B1–B2)/(1/0.2)+(B1–B3)/(1/0.8)......(1)
//(459.B2) = (B2–B1)/(1/0.2)+(B2–B3)/(1/0.8)............(2)
//0=(B3–B1)/(1/0.8)+(B3–B2)/(1/0.8).....................(3)
//solving these equations, we get  the values of B1, B2 and B3.
B1=987.7         //heat flux by surface 1.
B2=657.4         //heat flux by surface 2.
B3=822.6         //heat flux by surface 3.
qn12=(B1-B2)/(1/F12)+(B1-B3)/(1/(1-F12));         //net heat transfer between 1 and 2 if they are connected by an insulated diffuse reflector between the edges on both sides.

printf("net heat transfer between 1 and 2 if the surroundings are black is :%.2f W/m^2\n",Qn12);

printf("net heat transfer between 1 and 2 if they are connected by an insulated diffuse reflector between the edges on both sides is : %.0f W/m^2\n",qn12);

x=poly([0],'x');
 x=roots(5.67*10^-8*(x^4)-822.6);
 printf("\t temperature of the reflector is : %.0f K\n",x(4));
//end
```

Scilab code Exa 10.10 `heat transfer rate calculation`

```

clear ;
clc;

printf("\t Example 10.10\n");
T1=773;           //temp.of two sides of duct ,K
T2=373;           //temperature of the third side ,K
e1=0.5;         //emissivity of stainless steel
e2=0.15;         //emissivity of copper
a=5.67*10^-8;     //stefan constant
f12=0.4;         //view factor of 1 occupied by 2.
f21=0.67;        //view factor of  2 occupied by 1
f13=0.6;         // view factor of 1 occupied by 3
f31=0.75;         //view factor of 3 occupied by 1
f23=0.33;         //view factor of 2 occupied by 3
f32=0.25;         //view factor of 2 occupied by 3

A=[1 (-1+e2)*f12 (e2-1)*f13;(-1*e1*f21) 1 (e1*-1*f23
   );(e1*-1*f31) (e1*-1*f32) 1];    //matrix method
   to solve three equations to find radiosity

B=[e2*a*T2^4;e1*a*T1^4;e1*a*T1^4];     //matrix
   method to solve three equations to find radiosity

X=inv(A)*B;    //solution of above matrix method

Qn1=0.5*e2/(1-e2)*(a*T2^4-X(1));     //net heat
   transfer to the copper base per meter of the
   length of the duct ,W/m
Qn2=Qn1+2.6;
printf("net heat transfer to the copper base per
   meter of length of the duct is : %.1f W/m ,the -
```

```
ve sign indicates that the copper base is gaining heat.\n",Qn2);
//end)
```

Scilab code Exa 10.11 net heat radiation

```


clear;
clc;

printf("\t Example 10.11\n");
T1=1473 ;          //temp.of gas,K
T2=573 ;       //temp of walls,K
D1=0.4;         //diameter of combustor, m
a=5.67*10^-8;     //stefan boltzman coefficient,W/(m^2*K^4)
//we have Lo=D1=0.4m, a total pressure of 1 atm., pco2=0.2 atm. , using figure, we get eg=0.098.
eg=0.098;     //total emittance

ag=(T1/T2)^0.5*(0.074);     //total absorptance
//now we can calculate Qnetgas to wall. for these problems with one wall surrounding one gas, the use of the mean beam length in finding eg and ag accounts for all geometric effects and no view factor is required.

Qngw=%pi*D1*a*(eg*T1^4-ag*T2^4)/1000;     //net heat radiated to the walls,kW/m
printf("\t net heat radiated to the walls is : %.1f KW/m\n",Qngw);
//end
```

Scilab code Exa 10.12 `root temperature calculculation`

```

clear ;
clc;

printf("\t Example 10.12\n");
T1=291;            //temp.of sky,K
T2=308;         //temp of air,K
e1=0.9;              //emissivity 0f black paint
h=8;             //heat transfer coefficient,W/(m^2*K)
P=600 ;          //heat flux,W/m^2

//heat loss from the roof to the inside of the barn will lower the roof temp., since we dont have enough information to evaluate the loss, we can make an upper bound on roof temp. by assuming that no heat is transferred to the interior.

x=poly([0],'x');
x=roots(8*(e1*5.67*10^-8*(x^4-T1^4)+(x-T2)-e1*P));

//for white acrylic paint, by using table, e=0.9 and  absorptivity is 0.26,Troof


T=poly([0],'T');
T=roots(8*(e1*5.67*10^-8*(T^4-T1^4)+(T-T2)-0.26*P));
Tn=T(2)+0.6

printf("\t temp. of the root is :%.1f C or 312 K , the white painted roof is only a few degrees warmer than the air.\n",Tn);
```

```
//end
```

Chapter 11

An Introduction to mass Transfer

Scilab code Exa 11.1 `mol fraction and pressure density calculation`

```

clear;
clc;

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

Mn2=0.7556;          //mass fraction of nitrogen
Mo2=0.2315;     //mass fraction of oxygen
Mar=0.01289;     //mass fraction of argon gas
M1=28.02;     //molar mass of N2,kg/kmol
M2=32;     //molar mass of O2,kg/kmol
M3=39.95 ;    //molar mass of Ar,kg/kmol
Mair=(Mn2/M1+Mo2/M2+Mar/M3)^-1;     //molar mass of air ,kg/kmol

Xo2=Mo2*Mair/M2;     //mole fraction of O2
PO2=Xo2*101325;     //partial pressure of O2,Pa
Co2=PO2/(8314.5*300);     //molar volume of O2,kmol/m^3
```

```
ao2=Co2*M2;     //density of O2,kg/m^3

printf("mole fraction of O2 is :%.4f\n",Xo2);
printf("partial pressure of O2 is :%4e\n",PO2);
printf("molar volume of O2 is :%.5f kmol/m^3\n",Co2);
printf("density of O2 is :%.4f kg/m^3\n",ao2);
//end
```

Scilab code Exa 11.2 `mass and mole flux calculation`

```

clear;
clc;

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

r=0.00241;          //rate of consumption of carbon,kg/(m^2*s)
Mo2=0.2;     //concentration of oxygen at surface s
Mco2=0.052;     //concentration of CO2 at surface s
as=0.29;     //density of surface s,kg/m^3

//since carbon flows through a second imaginary surface u, the mass fluxes are relatedd by    Ncu=-12/32*No2s=12/44*Nco2s
//the minus sign arises because the O2 flow is opposite the C and CO2 flows.in steady state if we apply mass conservation to the control volume between the u and s surface, wee find that the total mass flux entering the u surface equals that leaving the s surface

Ncu=r;     //mass fluxes of carbon in u surface,kg/m^2/s
```

```

No2s=-32/12*Ncu;     //mass flux of O2 in surface s,kg/(m^2*s)
Nco2s=44/12*Ncu;     //mass flux of CO2 in surface s,kg/(m^2*s)
Vo2s=No2s/(Mo2*as);     //mass average speed,m/s
Vco2s=Nco2s/(as);     //mass average speed,m/s

Vs=(Nco2s+No2s)/as;     //effective mass average speed,m/s
j1=0.0584*(Vo2s-Vs)+0.000526;     //diffusional mass flux,kg/(m^2*s)
j2=0.0087+0.00014;     //diffusional mass flux,kg/(m^2*s)
//the diffusional mass fluxes are very nearly equal to the species m ss fluxes. tha is because the mass average speed is much less than species speeds.

N1=Ncu/12;     //mole flux of carbon through the surface s,kmol/(m^2*s)
N2=-N1;          //mole flux of oxygen through the surface s,kmol/(m^2*s)
printf("\t mass flux of O2 through an imaginary surface is :%.5f kg/(m^2*s)\n",j1);
printf("\t mass flux of CO2 through an imaginary surface is :%.5f kg/(m^2*s)\n",j2);

printf("\t mole flux of Co2 through an imaginary surface is :%f kmol/(m^2*s)\n",N1);
printf("\t mole flux of O2through an imaginary surface is :%f kmol/(m^2*s)\n",N2);
printf("\t the two diffusional mole fluxes sum to zero themselves because ther is no convective mole flux for other species to diffuse against. the reader may find the velocity of the interface. that calculation shows the interface to be receding so slowly that the velocities are equal
```

```
   to those that would be seen by a stationary
   observer. ")
//end
```

Scilab code Exa 11.3 `diffusivity calculation`

```


clear ;
clc;

printf("\t Example 11.3\n");
T1=276;          //temp.of air ,K
aa=3.711;     //lennard jones constant or collision
   diameter ,A
ab=2.827;     //lennard jones constant or collision
   diameter ,A
b1=78.6;     //lennard jones constant ,K
b2=59.7;       //lennard jones constant ,K
a=(aa+ab)/2;     //effective molecular diameter for
   collisions of hydrogen and air ,m
b=(b1*b2)^0.5;     //effective potential well depth ,K
c=T1/b;

d=0.8822;     //potential well function
Dab=1.8583*10^-7*T1^1.5/(a^2*d)*(1/2.016+1/28.97)
   ^0.5;     //diffusion coefficient of hydrogen in
   air ,m^2/s

printf("\t diffusion coefficient of hydrogen in air
   is :% -5e m^2/s an experimental value from table
   is 6.34*10^-5 m^2/s,so the prediction is high by
   5 percent.\n",Dab);
//end
```

Scilab code Exa 11.4 `transport properties calculation`

```

clear;
clc;

printf("\t Example 11.4\n");
T1=373.15;            //temp.of tea,K
XN2=0.7808;     //mole fraction of nitrogen
XO2=0.2095;     //mole fraction of oxygen
Xar=0.0093;     //mole fraction of
a=[3.798 3.467 3.542];     //collisin diameter,m
b=[71.4 106.7 93.3];     //lennard jones constant,K
M=[28.02 32 39.95];     //molar masses,kg/kmol
c=[0.9599 1.057 1.021];     //potential well function
d=[1.8*10^-5 2.059*10^-5 2.281*10^-5];     //calculated viscosity,kg/(m*s)
e=[1.8*10^-5 2.07*10^-5 2.29*10^-5 ];     //theoritical viscosity,kg/(m*s)
f=[0.0260 0.02615 0.01787];     //theoritical thermal conducitvity,W/(m*K)
i=1;
while(i<4)
u(i)=2.6693*10^-6*(M(i)*T1)^0.5/((a(i)^2*c(i)));   //viscosity,kg/(m*s)
k(i)=0.083228/((a(i))^2*c(i))*(T1/M(i))^0.5     //thermal conductivity,W/(m*s)

i=i+1;
end
umc=XN2*u(1)/0.9978+XO2*u(2)/1.008+Xar*u(3)/0.9435 ;     //calculated mixture viscosity,kg/(m*s)
umc1=1.857*10^-5;
printf("\t theoritical mixture viscosity is : %-5e
```

```
kg/(m*s)\n”,umc1);
umd=XN2*e(1)/0.9978+XO2*e(2)/1.008+e(3)*Xar/0.9435;      //theoritical mixture viscosity ,kg/(m*s)
printf(”\t calculated mixture viscosity is : %−5e kg/(m*s)\n”,umd);

kmc=XN2*k(1)/0.9978+XO2*k(2)/1.008+Xar*k(3)/0.9435;      //calculated thermal conducitvity ,W/(m*K)
kmc1=0.02623;
printf(”\t theoritical thermal conducitvity is : %f W/(m*K)\n”,kmc1);
kmd=XN2*f(1)/0.9978+XO2*f(2)/1.008+Xar*f(3)/0.9435;      //theoritical thermal conductivity , W/(m*K)
printf(”\t calculated thermal conducitvity is : %.5f W/(m*K)\n”,kmd);
Cp=1006    //mixture diffusivity ,j/(kg*K)
pr=umd*Cp/kmd;    //prandtl no.
printf(”\t prandtl no. is : %.3f\n”,pr);
//end
```

Scilab code Exa 11.5 `mass fraction calculation`

```

clear;
clc;

printf(”\t Example 11.5\n”);
Psat = [0.6113 1.2276 2.3385 4.2461 7.3837 12.35 19.941 31.188 47.39 70.139 101.325];
T = [0.01 10 20 30 40 50 60 70 80 90 100];
i=1;
while i<12
    xw(i)=Psat(i)/101.325;                        //mole fraction of water
    printf(”\n %.4f”,xw(i));
```

```
    mw(i)=(xw(i)*18.02)/(xw(i)*18.02+(1-xw(i))*28.96);//mass fraction of water
    i = i+1;
end
plot(T,mw);
xtitle("Mass fraction of water vapour in air above liquid water surface as a function of surface temperature(1 atm total pressure)","Temperature(degree celsius)","Mass fraction of water vapor");
```

Scilab code Exa 11.6 `mass fraction calculation`

```


clear ;
clc;

printf("\t Example 11.6\n");
T1=263.15;             //temp.of ice ,K

Pv=exp(21.99-6141/(T1));     //vapor pressure ,KPa
xw=Pv/101.325;     //mole fraction of water
mw=xw*18.02/(xw*18.02+(1-xw)*28.96);     //mass fraction
printf("\t mass fraction of watervapor above the surface of ice is :%.5f\n",mw);
//end
```

Scilab code Exa 11.10 `average rate of naphthalene loss`

```


clear;
```

```
clc;

printf("\t Example 11.10\n");
T1=303;           // isothermal temp.,K
v=5;    //air speed,m/s
l=0.05;    //length of naphthalene model that is flat , m
Mnap=128.2;        //molar mass of naphthalene,kg/kmol
D=0.86*10^-5;    //diffusion coefficient of naphthalene in air,m/s

Pv=10^(11.45-3729.3/T1)*133.31;    //vapor pressure, Pa
xn=Pv/101325;    //mole fraction  of naphthalene
mn=xn*Mnap/(xn*Mnap+(1-xn)*28.96);    //mass fraction of naphthalene
mnp=0;    //mass fraction of naphthalene in free stream is 0

Rel=v*l/(1.867*10^-5);        //reynolds no.
Sc=1.867*10^-5/D;    //schimidt no.
Nul=0.664*Rel^0.5*Sc^1/3;    //mass transfer nusselt no.
Gmn=D*Nul*1.166/l;    //gas phase mass transfer coefficient,kg/(m^2*s)
n=Gmn*(mn-mnp)+0.0000071;    //average mass flux,kg/(m^2*s)

printf("\t average rate of loss of naphthalene from a part of model is :%-4e kg/(m^2*s) or 58 g/(m^2*h)\n",n);
printf("\t naphthalene sublimatin can be used to infer heat transfer coefficient by measuring the loss of naphthalene from a model over some length of time.since the schimidt no. of naphthalene is not generally equal to prandtl no. under the conditions of interest, some assumption about the
```

```
dependence of nusselt no. on the prandtl no must usually be introduced.")
//end
```

Scilab code Exa 11.11 average concentration of helium

```

clear;
clc;

printf("\t Example 11.11\n");
T1=300;            //temp. of helium-water tube,K
h=0.4;     //height of vertical wall,m
m=0.087*10^-3;     //flow rate of helium,kg/(m^2*s)
//this is a uniform flux natural convection problem.

Mhes=0.01;     // assuming the value of mass fraction of helium at the wall to be 0.01
Mhef=Mhes/2;     //film composition

af=1.141;     //film density,kg/m^3
as=1.107;     //wall density,kg/m^3
Dha=7.119*10^-5;     //diffusion coefficient,m^2/s
u=1.857*10^5;     //fil,m viscosity at 300K,kg/(m*s)
Sc=(u/af)/Dha;     //schimidt no.
aa=1.177;     //air density,kg/m^3
Ra1=9.8*(aa-as)*m*h^4/(u*af*Dha^2*Mhes);     // Rayleigh no.

Nu=6/5*(Ra1*Sc/(4+9*Sc^0.5+10*Sc))^(1/5);          // approximate nusselt no.
s=m*h/(af*Dha*Nu);     //average concentration of helium at hte wall

//thus we have obtained an average wall
```

```
concentration 14 oercent higher than our initial guess of Mhes.we repeat this calclations with revised values of densities to obtain Mhes = 0.01142

printf(" average conentration of helium at the wall is 0.01142 ,since the result is within  0.5 percent of our second guess, a third iteration is not needed.");
//end
```

Scilab code Exa 11.14 concentration distribution

```

clear ;
clc;

printf("\t Example 11.14\n");
T1=325;             //temp. of helium-water tube,K
l=0.2;      //length of tube,m
x=0.01;        // mole fraction of water
//the vapor pressure of the liquid water is approximately the saturation pressure at the water temp.
p=1.341*10000 ;        //vapor pressure using steam table,Pa
x1=p/101325;     //mole fraction of saturated water
R=8314.472;     //gas constant,J/(kmol*K)
c= 101325/(R*T1);     //mole concentration in tube,kmol/m^3
D12=1.067*10^-4;         //diffusivity ofwater with respect to helium,m^2/s
Nw=c*D12*log(1+(x-x1)/(x1-1))/l ;    //molar evaporation rate,kmol/(m^2*s)

```

```
nw=Nw*18.02;          // mass evaporation rate,kg/(m^2*s)

//S=1+(x1-1)*exp(Nw*y/(c*D12))     //conentration distribution of water-vapor
printf("\t conentration distribution of water-vapor is   : x1(y)=1-0.8677*exp(0.6593*y) where y is expressed in meters.\n")
//end
```

Scilab code Exa 11.15 `rate of evaporation`

```

clear ;
clc;

printf("\t Example 11.15\n");
T1=1473;           //suraface temp.of hot water,K
x=0.05;          //mass fraction of water
Gm=0.017;         //average mass  transfer coefficient ,kg/(m^2*s)
A=0.04;          //suraface area of pan,m^2

//only water vapour passes through the liquid surface, since air is not strongly absorbed into water under normal conditions.

p=38.58*1000;        // saturation pressure of water, kPa
Xwater=p/101325;     //mole fraction of saturated water
Mwater=Xwater*18.02/(Xwater*18.02+(1-Xwater)*28.96);      //mass fraction of saturated water

B=(x-Mwater)/(Mwater-1);     //mass transfer driving
```

```
    force
m=Gm*B*A;              //evaporation rate,kg/s
printf("\t evaporation rate is:%f kg/s or 769 g/hr."
    ,m);
//end
```

Scilab code Exa 11.16 `mass transfer coefficient calculation`

```

clear;
clc;

printf("\t Example 11.16\n");
T1=298;            //temp.of air,K
T2=323.15;          //film temp.,K
x=0.05;          //mass fraction of water at 75 C
Gm=0.017;         //average mass  transfer coefficient
    ,kg/(m^2*s)
A=0.04;          //suraface area of pan,m^2
l=0.2;          //length of pan in flow direction,m
v=5;          //air speed,m/s
m=(x+0.277)/2;     //film composition of water at 50
    C
Mf=26.34;     //mixture molecular weight,kg/kmol
af=101325*Mf/(8314.5*T2);     //film density from
    ideal gas law,kg/m^3
Uf=1.75*10^-5;     //film viscosity,kg/(m*s)
Vf=Uf/af;     //kinematic viscosity,m^2/s
Rel=v*l/Vf;     //reynolds no. comes out to be 56,800
     so the flow is laminar.
B=0.314;      //mass transfer driving force

D=2.96*10^-5;     //diffusivity of water in air,m^2/s
Sc=Vf/D;     //scimidt no.

```

```
Nu=0.664*Rel^0.5*Sc^1/3;    //nussselt no.
Gmw1=Nu*(D*A/l);    //appropriate value of mass transfer gas phase coeffficient of water in air,kg/(m^2*s)
Gmw=Gmw1*(log(1+B)/B)+0.0168;    //mass transfer gas phase coeffficient of water in air,kg/(m^2*s)

printf("mass transfer gas phase coeffficient of water in air is : %.4f kg/(m^2*s)\n In this caes, the blowing factor is 0.870. thus the mild blowing has reduced the mass transfer coefficient by about 13 percent.",Gmw);

//end
```