Scilab Textbook Companion for
Process Heat Transfer
by D. Q. Kern[1]

Created by
Byragoni Spandana
Physics
Electrical Engineering
IIT-B
College Teacher
None
Cross-Checked by
Lavitha

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: Process Heat Transfer

Author: D. Q. Kern

Publisher: Tata McGraw-Hill, NY

Edition: 1

Year: 1950

ISBN: 0-07-085353-3

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 2

Conduction

Scilab code Exa 2.1 `Value of Q`

```
clc
//page 13
printf("\t Example 2.1 \n");
printf("\t approximate values are mentioned in the book \n");
Tavg=900; // average temperature of the wall,F
k=0.15; // Thermal conductivity at 932 F,Btu/(hr)(ft^2)(F/ft)
T1=1500; // hot side temperature,F
T2=300; // cold side temperature,F
A=192; // surface area,ft^2
L=0.5; // thickness,ft
Q=(k)*(A)*(T1-T2)/L; // formula for heat,Btu/hr
printf("\t heat is : %.2e Btu/hr \n",Q);
//end
```

Scilab code Exa 2.2 `Temperature increase`

```
//Page 14
clc
printf(”\t Example 2.2 \n”);
printf(”\t approximate values are mentioned in the book \n”);
La=8/12; // Thickness of firebrick wall,ft
Lb=4/12; // Thickness of insulating brick wall,ft
Lc=6/12; // Thickness of building brick wall,ft
Ka=0.68; // themal conductivity of firebrick,Btu/(hr)*(ft^2)*(F/ft)
Kb=0.15; // themal conductivity of insulating brick,Btu/(hr)*(ft^2)*(F/ft)
Kc=0.40; // themal conductivity of building brick,Btu/(hr)*(ft^2)*(F/ft)
A=1; // surface area,ft^2
Ta=1600; // temperature of inner wall,F
Tb=125; // temperature of outer wall.F
Ra=La/(Ka)*(A); // formula for resistance,(hr)*(F)/Btu
printf(”\t resistance offered by firebrick : %.2f (hr)*(F)/Btu \n”,Ra);
Rb=Lb/(Kb)*(A); // formula for resistance,(hr)*(F)/Btu
printf(”\t resistance offered by insulating brick : %.2f (hr)*(F)/Btu \n”,Rb);
Rc=Lc/(Kc)*(A); // formula for resistance,(hr)*(F)/Btu
printf(”\t resistance offered by buildingbrick : %.2f (hr)*(F)/Btu \n”,Rc);
R=Ra+Rb+Rc; // total resistance offered by three walls,(hr)*(F)/Btu
printf(”\t total resistance offered by three walls : %.2f (hr)*(F)/Btu \n”,R);
Q=(1600-125)/4.45; // using formula for heat loss/ft^2,Btu/hr
printf(”\t heat loss/ft^2 : %.0f Btu/hr \n”,Q);
// T1,T2 are temperatures at interface of firebrick and insulating brick, and insulating brick and
```

```
building brick respectively ,F
delta=(Q)*(Ra); // formula for temperature difference ,F
printf("\t delta is : %.0f F \n",delta);
T1=Ta-((Q)*(Ra)); // temperature at interface of firebrick and insulating brick ,F
printf("\t temperature at interface of firebrick and insulating brick :%.0f F \n",T1);
deltb=Q*(Rb);
printf("\t deltb is : %.0f F \n",deltb);
T2=T1-((Q)*(Rb)); //temperature at interface of insulating brick and building brick ,F
printf("\t temperature at interface of insulating brick and building brick :%.0f F \n",T2);
//end
```

Scilab code Exa 2.3 `Value of Q`

```
clc
//page 15
printf("\t example 2.3 \n");
printf("\t approximate values are mentioned in the book \n");
Lair=0.25/12; // thickness of air film ,ft
Kair=0.0265; // thermal conductivity of air at 572F, Btu/(hr)*(ft^2)(F/ft)
A=1; // surface area ,ft^2
Rair=Lair/(Kair*(A)); // resistance offered by air film , (hr)(F)/Btu
printf("\t resistance offered by air film %.2f (hr)(F)/Btu \n",Rair);
R=4.45; // resistance from previous example 2.2,(hr)(F)/Btu
Rt=(R)+Rair; // total resistance ,(hr)(F)/Btu
printf("\t total resistance %.2f (hr)(F)/Btu \n",Rt)
```

```
;
Ta=1600; // temperature of inner wall,F
Tb=125; // temperature of outer wall,F
Q=(1600-125)/Rt; // heat loss, Btu/hr
printf("\t heat loss %.2f Btu/hr \n",Q);
```

Scilab code Exa 2.4 `Heat flow through pipe wall`

```
//page 16
clc
printf("\t example 2.4 \n");
printf("\t approximate values are mentioned in the book \n");
k=0.63; // thermal conductivity of pipe, Btu/(hr)*(ft^2)*(F/ft)
Do=6; // in
Di=5; // in
Ti=200; // inner side temperature,F
To=175; // outer side temperature,F
q=(2*(3.14)*(k)*(Ti-To))/(2.3*log10(Do/Di)); // formula for heat flow,Btu/(hr)*(ft)
printf("\t heat flow is : %.0f Btu/(hr)*(ft) \n",q);
// caculation mistake in book
// end
```

Scilab code Exa 2.5 `Temperature`

```
clc
//page 19
printf("\t example 2.5 \n");
printf("\t approximate values are mentioned in the book \n");
```

```
t1=150; // assume temperature of outer surface of rockwool,F
ta=70; // temperature of surrounding air,F
ha=2.23; // surface coefficient,Btu/(hr)*(ft^2)*(F)
q=(3.14)*(300-70)/(((2.3/(2*0.033))*log10(3.375/2.375))+(1/((2.23)*(3.375/12)))); // using formula for heat loss,Btu/(hr)*(lin ft),calculation mistake
printf("\t heat loss for linear foot is : %.1f Btu/(hr)*(lin ft) \n",q);
printf("\t Check between ts and t1, since delt/R = deltc/Rc \n");
t1=300-(((104.8)*((2.3)*(log10(3.375/2.375))))/((2)*(3.14)*(.033))); // using eq 2.31,F
printf("\t t1 is : %.1f F \n",t1);
t1=125; // assume temperature of outer surface of rockwool,F
ha=2.10; // surface coefficient,Btu/(hr)*(ft^2)*(F)
q=((3.14)*(300-70))/(((2.3/(2*0.033))*log10(3.375/2.375))+(1/((2.10)*(3.375/12)))); // using formula for heat loss,Btu/(hr)*(lin ft)
printf("\t heat loss for linear foot is : %.1f Btu/(hr)*(lin ft) \n",q);
printf("\t Check between ts and t1, since delt/R = deltc/Rc \n");
t1=300-(((103)*((2.3)*(log10(3.375/2.375))))/((2)*(3.14)*(.033))); // using eq 2.31,F
printf("\t t1 is : %.1f F \n",t1);
// end
```

Chapter 4

Radiation

Scilab code Exa 4.1 `Radiation between large 2 planes`

```
clc
//page 75
printf("\t example 4.1 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=1000+460; // R
T2=800+460; // R
Q=((0.173)*((14.6)^4-(12.6)^4)); // using eq.4.24,
   Btu/(hr)*(ft^2)
printf("\t heat removed from colder wall per unit
   area is : %.0f Btu/(hr)*(ft^2) \n",Q);
// end
```

Scilab code Exa 4.2 `Radiation between Planes with Different Emissivities`

```
printf("\t example 4.2 \n");
printf("\t approximate values are mentioned in the
   book \n");
```

```
T1=1000+460; // R
T2=800+460; // R
e1=0.6; // emissivity of hotter wall
e2=0.8; // emissivity of colder wall
Q=(((0.173)/((1/0.6)+(1/0.8)-1))*((14.6)^4-(12.6)^4)
   ); // using eq.4.26,heat loss per unit area,Btu/(
   hr)*(ft^2)
printf("\t heat removed from colder wall per unit
   area is : %.0f Btu/(hr)*(ft^2) \n",Q);
printf("\t For perfect black bodies the value was
   3500 Btu/(hr)(ft^2) \n");
// end
```

Scilab code Exa 4.3 `Calculation of Radiation from a Pipe`

```
clc
//page 78
printf("\t example 4.3 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=125+460; // R
T2=70+460; // R
e=0.9; // emissivity,using table 4.1B
A=(%pi)*(3.375/12)*(1); // area,ft^2/lin ft
printf("\t area is : %.2f ft^2/lin ft \n",A);
Q=(0.9)*(0.88)*(0.173)*((T1/100)^4-(T2/100)^4); //
   heat loss using eq.4.32,Btu/(hr)*(lin ft)
printf("\t heat loss  is : %.1f Btu/(hr)*(lin ft) \n
   ",Q);
hr=(Q)/((A)*(T1-T2)); // fictitious film coefficient
   ,using eq 4.33,Btu/(hr)(ft^2)(F)
printf("\t fictitious film coefficient is : %.2f Btu
   /(hr)(ft^2)(F) \n",hr);
//end
```

Scilab code Exa 4.4 `Radiation from a Pipe to a Duct`

```
clc
//page 82
printf(”\t example 4.4 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=300+460; // R
T2=75+460; //R
A1=0.622; // area from table 11 in the appendix A,ft^2/lin ft
A2=4*(1*1); // surface area of duct,ft^2/lin ft
e1=0.79; // emissivity of oxidized steel from table 4.1
e2=0.276; // emissivity of oxidized zinc from table 4.1
printf(”\t surface area of pipe is : %.3f ft^2/lin ft \n”,A1);
printf(”\t surface area of duct is : %.0f ft^2/lin ft \n”,A2);
printf(”\t The surface of the pipe is not negligible by comparison with that of the duct, and(f) of Table 4.2 applies most nearly \n”);
Fa=1; // from table 4.2
Fe=((1)/((1/e1)+((A1/A2)*((1/e2)-1)))); // from table 4.2
printf(”\t Fe is : %.2f \n”,Fe);
Q=(0.173*10^-8)*(Fa)*(Fe)*(A1)*((T1)^4-(T2)^4); // heat loss due to radiation,Btu/(hr)*(lin ft)
printf(”\t heat loss due to radiation is : %.0f Btu/(hr)*(lin ft) \n”,Q);
// end
```

Chapter 5

Temperature

Scilab code Exa 5.1 `Calculation of the LMTD`

```
printf(”\t example 5.1 \n”);
T1=300; // hot fluid inlet temperature ,F
T2=200; // hot fluid outlet temperature ,F
t1=100; // cold fluid inlet temperature ,F
t2=150; // cold fluid outlet temperature ,F
printf(”\t for counter current flow \n”);
delt1=T1-t2; //F
delt2=T2-t1; // F
printf(”\t delt1 is : %.0f F \n”,delt1);
printf(”\t delt2 is : %.0f F \n”,delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf(”\t LMTD is :%.1f F \n”,LMTD);
printf(”\t for parallel flow \n”);
delt1=T1-t1; // F
delt2=T2-t2; // F
printf(”\t delt1 is : %.0f F \n”,delt1);
printf(”\t delt2 is : %.0f F \n”,delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf(”\t LMTD is :%.0f F \n”,LMTD);
//end
```

Scilab code Exa 5.2 `Calculation of the LMTD with Equal Outlet Temperatures`

```
printf("\t example 5.2 \n");
T1=300; // hot fluid inlet temperature,F
T2=200; // hot fluid outlet temperature,F
t1=150; // cold fluid inlet temperature,F
t2=200; // cold fluid outlet temperature,F
printf("\t for counter current flow \n");
delt1=T1-t2; //F
delt2=T2-t1; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
printf("\t for parallel flow \n");
delt1=T1-t1; // F
delt2=T2-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
if(delt2==0);
    printf("\t denominator becomes infinity so LMTD becomes Zero \n");
    printf("\t LMTD is Zero \n");
else
    LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
    end
//end
```

Scilab code Exa 5.3 `Calculation of the LMTD`

```
printf(”\t example 5.3 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=300; // hot fluid inlet temperature,F
T2=200; // hot fluid outlet temperature,F
t1=100; // cold fluid inlet temperature,F
t2=275; // cold fluid outlet temperature,F
printf(”\t for counter current flow \n”);
deltc=T2-t1; //F
delth=T1-t2; // F
printf(”\t delth is : %.0f F \n”,delth);
printf(”\t deltc is : %.0f F \n”,deltc);
LMTD=((delth-deltc)/((2.3)*(log10(delth/deltc))));
printf(”\t LMTD is :%.1f F \n”,LMTD);
//end
```

Scilab code Exa 5.4 `Calculation of the LMTD with One Isothermal Fluid`

```
printf(”\t example 5.4 \n”);
printf(”\t process is isothermal with hot fluid so temperature of hot fluid remains constant \n”);
T1=300; // hot fluid inlet temperature,F
T2=300; // hot fluid outlet temperature,F
t1=100; // cold fluid inlet temperature,F
t2=275; // cold fluid outlet temperature,F
printf(”\t for counter current flow \n”);
delt1=T1-t2; //F
delt2=T2-t1; // F
printf(”\t delt1 is : %.0f F \n”,delt1);
printf(”\t delt2 is : %.0f F \n”,delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf(”\t LMTD is :%.0f F \n”,LMTD);
printf(”\t for parallel flow \n”);
delt1=T1-t1; // F
delt2=T2-t2; // F
```

```
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
if(delt2==0);
    printf("\t denominator becomes infinity so LMTD
       becomes Zero \n");
    printf("\t LMTD is Zero \n");
else
    LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1)))
       );
printf("\t LMTD is :%.0f F \n",LMTD);
    end
printf("\t these are identical \n");
//end
```

Scilab code Exa 5.5 `Calculation of point`

```
printf("\t example 5.5 \n");
printf("\t approximate values are mentioned in the
   book \n");
printf("\t for inlet \n");
t1=99.1; // temperature of inlet ,F
t2=129.2; // temperature of outlet ,F
c=.478; // Btu/(hr)*(ft)*(F/ft)
mu=2.95*2.42; // lb/(ft)(hr)
k=0.078; // Btu/(hr)*(ft)*(F/ft)
G=854000; // mass velocity ,lb/(ft^2)(hr)
D=0.622/12; // diameter ,ft
Re=((D)*((G)/(mu)))^(0.9);
printf("\t Re is : %.2e \n",Re);
Pr=((c)*(mu)/k)^(1/3); // prandtl number raised to
   power 1/3
printf("\t Pr is : %.2f \n",Pr);
Nu=0.0115*(Re)*(Pr); // formula for nusselt number
printf("\t nusselt number is : %.0f \n",Nu);
hi=((k)*(Nu)/(D)); // heat transfer coefficient
```

```
printf("\t heat transfer coefficient is : %.0f \n",hi); // caculation mistake in book
printf("\t for outlet \n");
c=.495; // Btu/(hr)*(ft)*(F/ft)
mu=2.20*2.42; // lb/(ft)(hr)
k=0.078; // Btu/(hr)*(ft)*(F/ft)
G=854000; // mass velocity,lb/(ft^2)(hr)
D=0.622/12; // diameter,ft
Re=((D)*((G)/(mu)))^(.9); // reynolds number raised to poer 0.9, calculation mistake in book
printf("\t Re is : %.2e \n",Re);
Pr=((c)*(mu)/k)^(1/3); // prandtl number raised to power 1/3
printf("\t Pr is : %.2f \n",Pr);
Nu=0.0115*(Re)*(Pr); // formula for nusselt number
printf("\t nusselt number is : %.0f \n",Nu);
hi=((k)*(Nu)/(D)); // heat transfer coefficient
printf("\t heat transfer coefficient is : %.0f \n",hi); // caculation mistake in book
//end
```

Scilab code Exa 5.6 Calculation of the Caloric Temperature

```
printf("\t example 5.6 \n");
printf("\t approximate values are mentioned in the book \n");
T1=300; // hot fluid inlet temperature,F
T2=200; // hot fluid outlet temperature,F
t1=80; // cold fluid inlet temperature,F
t2=120; // cold fluid outlet temperature,F
printf("\t for counter current flow \n");
delT=T1-T2; // temperature difference for crude oil,F
printf("\t temperature difference for crude oil is : %.0f F \n",delT);
```

```
Kc=0.68; // from fig.17
delt=t2-t1; // temperature difference for gasoline ,F
printf("\t temperature difference for gasoline is : %.0f F \n",delt);
Kc<=0.10; // from fig.17
printf("\t The larger value of K. correspQnds to the controlling heat transfer coefficient which is assumed to establish the variation of U with temperature \n");
deltc=T2-t1; //F
delth=T1-t2; // F
printf("\t deltc is : %.0f F \n",deltc);
printf("\t delth is : %.0f F \n",delth);
A=((deltc)/(delth));
printf("\t ratio of two local temperature difference is : %.3f \n",A);
Fc=0.425; // from fig.17
Tc=((T2)+((Fc)*(T1-T2))); // caloric temperature of hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.1f F \n",Tc);
tc=((t1)+((Fc)*(t2-t1))); // caloric temperature of cold fluid ,F
printf("\t caloric temperature of cold fluid is : %.0f F \n",tc);
// end
```

Chapter 6

CounterflowDouble pipe exchangers

Scilab code Exa 6.1 `Double Pipe Benzene Toluene Exchanger`

```
printf("\t example 6.1 \n");
printf("\t approximate values are mentioned in the book \n");
T1=160; // inlet hot fluid ,F
T2=100; // outlet hot fluid ,F
t1=80; // inlet cold fluid ,F
t2=120; // outlet cold fluid ,F
w=9820; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for benzene \n");
tav=((t1+t2)/2); // F
printf("\t average temperature of benzene is : %.0f F \n",tav);
c=0.425; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for benzene is : %.2e Btu/hr \n",Q);
printf("\t for toulene \n");
Tav=((T1+T2)/2); //F
```

```
printf("\t average temperature of toulene is : %.0f F \n",Tav);
c=0.44; // Btu/(lb)*(F)
W=((Q)/((c)*(T1-T2))); // lb/hr
printf("\t W is :%.2e lb/hr \n",W);
printf("\t 2.LMTD \n");
printf("\t for counter current flow \n");
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
printf("\t 3.caloric temperatures \n");
printf("\t both streams will show that neither is viscous at the cold terminal (the viscosities less than 1 centipoise) and the temperature ranges and temperature difference are moderate. The coefficients may accordingly be evaluated from properties at the arithmetic mean, and the value of (mu/muw)^0.14 may be assumed equal to 1.0 \n");
tav=((t1+t2)/2); // F
printf("\t average temperature of benzene is : %.0f F \n",tav);
Tav=((T1+T2)/2); //F
printf("\t average temperature of toulene is : %.0f F \n",Tav);
printf("\t hot fluid:annulus,toulene \n");
D1=0.138; // ft
D2=0.1725; // ft
aa=((%pi)*(D2^2-D1^2)/4); // flow area,ft^2
printf("\t flow area is : %.5f ft^2 \n",aa);
De=(D2^2-D1^2)/D1; // equiv diameter,ft
printf("\t equiv diameter is : %.4f ft \n",De);
Ga=(W/aa); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Ga);
```

```
mu1=0.41*2.42; // at 130 F,lb/(ft)*(hr)
Rea=((De)*(Ga)/mu1); // reynolds number
printf(”\t reynolds number is : %.1e \n”,Rea);
jH=167; // from fig.24
c=0.44; // Btu/(lb)*(F),at 130F
k=0.085; // Btu/(hr)*(ft^2)*(F/ft), from table 4
Pr=((c)*(mu1)/k)^(1/3); // prandelt number raised to power 1/3
printf(”\t Pr is : %.3f \n”,Pr);
ho=((jH)*(k/De)*(Pr)*(1^0.14)); // using eq.6.15b, Btu/(hr)*(ft^2)*(F)
printf(”\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n”,ho);
printf(”\t cold fluid:inner pipe,benzene \n”);
D=0.115; // ft
ap=((%pi)*(D^2)/4); // flow area, ft^2
printf(”\t flow area is : %.4f ft^2 \n”,ap);
Gp=(w/ap); // mass velocity,lb/(hr)*(ft^2)
printf(”\t mass velocity is : %.2e lb/(hr)*(ft^2) \n”,Gp);
mu2=0.5*2.42; // at 130 F,lb/(ft)*(hr)
Rep=((D)*(Gp)/mu2); // reynolds number
printf(”\t reynolds number is : %.2e \n”,Rep);
jH=236; // from fig.24
c=0.425; // Btu/(lb)*(F),at 130F
k=0.091; // Btu/(hr)*(ft^2)*(F/ft), from table 4
Pr=((c)*(mu2)/k)^(1/3); // prandelt number raised to power 1/3
printf(”\t Pr is : %.3f \n”,Pr);
hi=((jH)*(k/D)*(Pr)*(1^0.14)); // using eq.6.15a,Btu/(hr)*(ft^2)*(F)
printf(”\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n”,hi);
ID=1.38; // ft
OD=1.66; //ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf(”\t Correct hi to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n”,hio);
```

```
Uc=((hio)*(ho)/(hio+ho)); // clean overall
   coefficient ,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(
   hr)*(ft^2)*(F) \n",Uc);
Rd=0.002; // required by problem ,(hr)*(ft^2)*(F)/Btu
UD=((Uc)/((1)+(Uc*Rd))); // design overall
   coefficient ,Btu/(hr)*(ft^2)*(F)
printf("\t design overall coefficient is : %.0f Btu
   /(hr)*(ft^2)*(F) \n",UD);
A=((Q)/((UD)*(LMTD))); // required surface ,ft^2
printf("\t required surface is : %.1f ft^2 \n",A);
A1=0.435; // From Table 11 for 1(1/4)in IPS standard
    pipe there are 0.435 ft2 of external surface per
    foot length ,ft^2
L=(A/A1); // required length;lin ft
printf("\t required length is : %.0f lin ft \n",L);
printf("\t This may be fulfilled by connecting three
    20-ft hairpins in series \n");
A2=120*0.435; // actual surface supplied ,ft^2
printf("\t actual surface supplied is : %.1f ft^2 \n
   ",A2);
UD=((Q)/((A2)*(LMTD)));
printf("\t actual design overall coefficient is : %
   .0f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \
   n",Rd);
printf("\t pressure drop  for annulus \n");
De1=(D2-D1); //ft
printf("\t De1 is : %.4f ft \n",De1);
Rea1=((De1)*(Ga)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Rea1);
f=(0.0035)+((0.264)/(Rea1^0.42)); // friction factor
   , using eq.3.47b
printf("\t friction factor is : %.4f \n",f);
s=0.87;
row=62.5*0.87; // from table 6
delFa=((4*f*(Ga^2)*L)/(2*4.18*(10^8)*(row^2)*(De1)))
```

```
; // ft
printf(”\t delFa is : %.1f ft \n”,delFa);
V=((Ga)/(3600*row)); //fps
printf(”\t V is : %.2f fps \n”,V);
Fl=((3*(V^2))/(2*32.2)); //ft
printf(”\t Fl is : %.1f ft \n”,Fl);
delPa=((delFa+Fl)*(row)/144); // psi
printf(”\t delPa is : %.1f psi \n”,delPa);
printf(”\t allowable delPa is 10 psi \n”);
printf(”\t pressure drop  for inner pipe \n”);
f=(0.0035)+((0.264)/(Rep^0.42)); // friction factor, using eq.3.47b
printf(”\t friction factor is : %.4f \n”,f);
s=0.88;
row=62.5*0.88; // from table 6
delFp=((4*f*(Gp^2)*L)/(2*4.18*(10^8)*(row^2)*(D))); // ft
printf(”\t delFp is : %.1f ft \n”,delFp);
delPp=((delFp)*(row)/144); // psi
printf(”\t delPp is : %.1f psi \n”,delPp);
printf(”\t allowable delPp is 10 psi \n”);
//end
```

Scilab code Exa 6.2 Calculation of the True Temperature Difference

```
printf(”\t example 6.2 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=300; // inlet hot fluid,F
T2=200; // outlet hot fluid,F
t1=190; // inlet cold fluid,F
t2=220; // outlet cold fluid,F
n=6; // number of parallel streams
P=((T2-t1)/(T1-t1));
printf(”\t P is : %.3f \n”,P);
```

```
R=((T1-T2)/((n)*(t2-t1)));
printf("\t R is : %.3f \n",R);
gama=((1-P)/((2.3)*((n*R)/(R-1))*log10(((R-1)/R)*(1/P)^(1/n)+(1/R)))); // using eq.6.35a
printf("\t gama is : %.3f \n",gama);
delt=(gama*(T1-t1)); // true temperature difference , F
printf("\ true temperature difference is : %.1f F \n",delt);
//end
```

Scilab code Exa 6.3 `Double Pipe Lube Oil Crude Oil Exchanger`

```
printf("\t example 6.3 \n");
printf("\t approximate values are mentioned in the book \n");
T1=450; // inlet hot fluid ,F
T2=350; // outlet hot fluid ,F
t1=300; // inlet cold fluid ,F
t2=310; // outlet cold fluid ,F
W=6900; // lb/hr
w=72500; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for lube oil \n");
c=0.62; // Btu/(lb)*(F)
Q=((W)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for lube oil is : %.2e Btu/hr \n",Q);
printf("\t for crude oil \n");
c=0.585; // Btu/(lb)*(F)
Q1=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for crude oil is : %.2e Btu/hr \n",Q1); // calculation mistake in book
delt1=T2-t1; //F
```

```
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
A=((delt1)/(delt2));
printf("\t ratio of two local temperature difference is : %.3f \n",A);
Fc=0.395; // from fig.17
Tc=((T2)+((Fc)*(T1-T2))); // caloric temperature of hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.1f F \n",Tc);
tc=((t1)+((Fc)*(t2-t1))); // caloric temperature of cold fluid ,F
printf("\t caloric temperature of cold fluid is : %.0f F \n",tc);
printf("\t hot fluid:annulus,lube oil \n");
D1=0.199; // ft
D2=0.256; // ft
aa=((%pi)*(D2^2-D1^2)/4); // flow area,ft^2
printf("\t flow area is : %.4f ft^2 \n",aa);
De=(D2^2-D1^2)/D1; // equiv diameter,ft
printf("\t equiv diameter is : %.2f ft \n",De);
Ga=(W/aa); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Ga);
mu1=3*2.42; // at 389.5F,lb/(ft)*(hr), from fig.14
Rea=((De)*(Ga)/mu1); // reynolds number
printf("\t reynolds number is : %.0e \n",Rea);
jH=20.5; // from fig.24
c=0.615; // Btu/(lb)*(F),at 130F
k=0.067; // Btu/(hr)*(ft^2)*(F/ft), from table 4
Pr=((c)*(mu1)/k)^(1/3); // prandelt number raised to power 1/3
printf("\t Pr is : %.3f \n",Pr);
Ho=((jH)*(k/De)*(Pr)); // H0=(h0/phya),using eq.6.15,Btu/(hr)*(ft^2)*(F)
```

```
printf("\t individual heat transfer coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Ho);
printf("\t cold fluid:inner pipe,crude oil \n");
D=0.172; // ft
ap=((%pi)*(D^2)/4); // flow area, ft^2
printf("\t flow area is : %.4f ft^2 \n",ap);
Gp=(w/(2*ap)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gp);
mu2=0.83*2.42; // at 304 F,lb/(ft)*(hr)
Rep=((D)*(Gp)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Rep);
jH=320; // from fig.24
c=0.585; // Btu/(lb)*(F),at 304F,from fig.4
k=0.073; // Btu/(hr)*(ft^2)*(F/ft), from fig.1
Pr=((c)*(mu2)/k)^(1/3); // prandelt number raised to power 1/3
printf("\t Pr is : %.3f \n",Pr);
Hi=((jH)*(k/D)*(Pr)*(1^0.14)); //Hi=(hi/phyp),using eq.6.15a,Btu/(hr)*(ft^2)*(F)
printf("\t Hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",Hi);
ID=2.067; // ft
OD=2.38; //ft
Hio=((Hi)*(ID/OD)); //Hio=(hio/phyp), using eq.6.5
printf("\t Correct Hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",Hio);
muw=0.77*2.42; // lb/(ft)*(hr), from fig.14
phyp=(mu2/muw)^0.14;
printf("\t phyp is : %.0f \n",phyp); // from fig.24
hio=(Hio)*(1); // from eq.6.37
printf("\t Correct hio to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
tw=(tc)+(((Ho)/(Hio+Ho))*(Tc-tc)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
muw=6.6*2.42; // lb/(ft)*(hr), from fig.14
phya=(mu1/muw)^0.14;
printf("\t phya is : %.1f \n",phya); // from fig.24
ho=(Ho)*(phya); // from eq.6.36
```

```
printf(”\t Correct h0 to the surface at the OD is : %.1f Btu/(hr)*(ft^2)*(F) \n”,ho);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient ,Btu/(hr)*(ft^2)*(F)
printf(”\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n”,Uc);
Rd=0.006; // required by problem,(hr)*(ft^2)*(F)/Btu
UD=((Uc)/((1)+(Uc*Rd))); // design overall coefficient ,Btu/(hr)*(ft^2)*(F)
printf(”\t design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n”,UD);
A=((Q)/((UD)*(LMTD))); // required surface,ft^2
printf(”\t required surface is : %.0f ft^2 \n”,A);
A1=0.622; // From Table 11,ft^2
Lr=(A/A1); // required length;lin ft
printf(”\t required length is : %.0f lin ft \n”,Lr);
printf(”\t Since two parallel streams are employed, use eight 20 ft hairpins or 320 lin. feet \n”);
L=320;
A2=320*0.622; // actual surface supplied,ft^2
printf(”\t actual surface supplied is : %.1f ft^2 \n”,A2);
UD=((Q)/((A2)*(LMTD)));
printf(”\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n”,UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf(”\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n”,Rd);
printf(”\t pressure drop  for annulus \n”);
De1=.058; //ft
printf(”\t De1 is : %.3f ft \n”,De1);
Rea1=((De1)*(Ga)/7.25); // reynolds number
printf(”\t reynolds number is : %.2e \n”,Rea1);
f=(0.0035)+((0.264)/(2680^0.42)); // friction factor , using eq.3.47b
printf(”\t friction factor is : %.4f \n”,f);
s=0.775;
row=62.5*0.775; // from fig 6
```

```
delFa=((4*f*(Ga^2)*L)/(2*4.18*(10^8)*(row^2)*(De1))); // ft
printf("\t delFa is : %.1f ft \n",delFa);
V=((Ga)/(3600*row)); //fps
printf("\t V is : %.1f fps \n",V);
delFl=((8*(V^2))/(2*32.2)); //ft
printf("\t delFl is : %.2f ft \n",delFl);
delPa=((delFa+delFl)*(row)/144); // psi
printf("\t delPa is : %.1f psi \n",delPa);
printf("\t allowable delPa is 10 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=(0.0035)+((0.264)/(Rep^0.42)); // friction factor, using eq.3.47b
printf("\t friction factor is : %.5f \n",f);
s=0.76;
row=62.5*0.76; // from table 6
Lp=160;
delFp=((4*f*(Gp^2)*Lp)/(2*4.18*(10^8)*(row^2)*(D))); // ft
printf("\t delFp is : %.1f ft \n",delFp);
delPp=((delFp)*(row)/144); // psi
printf("\t delPp is : %.1f psi \n",delPp);
printf("\t allowable delPp is 10 psi \n");
// end
```

Chapter 7

Parallel Counterflow Shell and Tube Excahngers

Scilab code Exa 7.1 `shell side equivalent`

```
printf(”\t example 7.1 \n”);
PT=1; // square pitch,in
do=0.75; // outer diameter,in
de=((4*(PT^2-(3.14*do^2/4)))/(3.14*do));
printf(”\t equivalent diameter is : %.2f in \n”,de);
De=(de/12); // ft
printf(”\t De is : %.3f in \n”,De);
//end
```

Scilab code Exa 7.2 `Calculation of FT for Fluids with Equal Ranges`

```
printf(”\t example 7.2 \n”);
printf(”\t approximate values are mentioned in the book \n”);
printf(”\t considering 50F approach \n”);
T1=350; //F
```

```
T2=250; //F
t2=T2-50; // formula for approach,f
printf("\t t2 is : %.0f F \n",t2);
printf("\t fluids are with equal ranges,so \n");
t1=t2-(T1-T2); // F
printf("\t t1 is : %.0f F \n",t1);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.0f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.2f \n",S);
printf("\t FT is 0.925 \n"); // from fig 18
printf("\t considering 0F approach \n");
T1=300; //F
T2=200; //F
t2=T2-0; // formula for approach,f
printf("\t t2 is : %.0f F \n",t2);
printf("\t fluids are with equal ranges,so \n");
t1=t2-(T1-T2); // F
printf("\t t1 is : %.0f F \n",t1);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.0f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.2f \n",S);
printf("\t FT is 0.80 \n"); // from fig 18
printf("\t considering 20F cross \n");
T1=280; //F
T2=180; //F
t2=T2+20; // formula for approach,f
printf("\t t2 is : %.0f F \n",t2);
printf("\t fluids are with equal ranges,so \n");
t1=t2-(T1-T2); // F
printf("\t t1 is : %.0f F \n",t1);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.0f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.3f \n",S);
printf("\t FT is 0.64 \n"); // from fig 18
//end
```

Scilab code Exa 7.3 `Calculation of a Kerosene Crude Oil Exchanger`

```
printf("\t example 7.3 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=390; // inlet hot fluid ,F
T2=200; // outlet hot fluid ,F
t1=100; // inlet cold fluid ,F
t2=170; // outlet cold fluid ,F
W=43800; // lb/hr
w=149000; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for kerosene \n");
c=0.605; // Btu/(lb)*(F)
Q1=((W)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for kerosene is : %.1
   e Btu/hr \n",Q1); // calculation mistake in
   problem
printf("\t for crude oil \n");
c=0.49; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for mid continent
   crude is : %.1e Btu/hr \n",Q); // calculation
   mistake in problem
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.2f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.3f \n",S);
```

```
printf("\t FT is 0.905 \n"); // from fig 18
delt=(0.905*LMTD); // F
printf("\t delt is : %.0f F \n",delt);
X=((delt1)/(delt2));
printf("\t ratio of two local temperature difference is : %.3f \n",X);
Fc=0.42; // from fig.17
Kc=0.20; // crude oil controlling
Tc=((T2)+((Fc)*(T1-T2))); // caloric temperature of hot fluid,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+((Fc)*(t2-t1))); // caloric temperature of cold fluid,F
printf("\t caloric temperature of cold fluid is : %.0f F \n",tc);
printf("\t hot fluid:shell side,kerosene \n");
ID=21.25; // in
C=0.25; // clearance
B=5; // baffle spacing,in
PT=1.25;
as=((ID*C*B)/(144*PT)); // flow area,ft^2
printf("\t flow area is : %.4f ft^2 \n",as);
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=0.40*2.42; // at 280F,lb/(ft)*(hr), from fig.14
De=0.99/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
jH=93; // from fig.28
c=0.59; // Btu/(lb)*(F),at 280F,from fig.4
k=0.0765; // Btu/(hr)*(ft^2)*(F/ft), from fig.1
Pr=((c)*(mu1)/k)^(1/3); // prandelt number raised to power 1/3
printf("\t Pr is : %.3f \n",Pr);
Ho=((jH)*(k/De)*(Pr)); // H0=(h0/phya),using eq.6.15,Btu/(hr)*(ft^2)*(F)
```

```
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Ho);
printf("\t cold fluid:inner tube side,crude oil \n");
D=0.0675; // ft
Nt=158;
n=4; // number of passes
L=16; //ft
at1=0.515; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu2=3.6*2.42; // at 129F,lb/(ft)*(hr)
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
jH=31; // from fig.24
c=0.49; // Btu/(lb)*(F),at 304F,from fig.4
k=0.077; // Btu/(hr)*(ft^2)*(F/ft), from fig.1
Pr=((c)*(mu2)/k)^(1/3); // prandelt number raised to power 1/3
printf("\t Pr is : %.3f \n",Pr);
Hi=((jH)*(k/D)*(Pr)*(1^0.14)); //Hi=(hi/phyp),using eq.6.15a,Btu/(hr)*(ft^2)*(F)
printf("\t Hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",Hi);
ID=0.81; // ft
OD=1; //ft
Hio=((Hi)*(ID/OD)); //Hio=(hio/phyp), using eq.6.5
printf("\t Correct Hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",Hio);
muw=1.5*2.42; // lb/(ft)*(hr), from fig.14
phyt=(mu2/muw)^0.14;
printf("\t phyt is : %.2f \n",phyt); // from fig.24
hio=(Hio)*(phyt); // from eq.6.37
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
```

```
tw=(tc)+(((Ho)/(Hio+Ho))*(Tc-tc)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
muw=0.56*2.42; // lb/(ft)*(hr), from fig.14
phys=(mu1/muw)^0.14;
printf("\t phys is : %.2f \n",phys); // from fig.24
ho=(Ho)*(phys); // from eq.6.36
printf("\t Correct h0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
A2=0.2618; // actual surface supplied for each tube, ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.5f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
f=0.00175; // friction factor for reynolds number 25300, using fig.29
s=0.73; // for reynolds number 25300,using fig.6
Ds=21.25/12; // ft
N=(12*L/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t allowable delPa is 10 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.000285; // friction factor for reynolds number 8220, using fig.26
s=0.83;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(
```

```
phyt))); // using eq.7.45,psi
printf(”\t delPt is : %.1f psi \n”,delPt);
X1=0.15; // X1=((V^2)/(2*g)), for Gt 1060000,using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf(”\t delPr is : %.1f psi \n”,delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf(”\t delPT is : %.1f psi \n”,delPT);
printf(”\t allowable delPs is 10 psi \n”);
//end
```

Scilab code Exa 7.4 `Calculation of a Distilled water Raw water Exchanger`

```
printf(”\t example 7.4 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=93; // inlet hot fluid,F
T2=85; // outlet hot fluid,F
t1=75; // inlet cold fluid,F
t2=80; // outlet cold fluid,F
W=175000; // lb/hr
w=280000; // lb/hr
printf(”\t 1.for heat balance \n”);
printf(”\t for distilled water \n”);
c=1; // Btu/(lb)*(F)
Q=((W)*(c)*(T1-T2)); // Btu/hr
printf(”\t total heat required for distilled water is : %.1e Btu/hr \n”,Q);
printf(”\t for raw water \n”);
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf(”\t total heat required for raw water is : %.1e Btu/hr \n”,Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
```

```
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.2f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.3f \n",S);
printf("\t FT is 0.945 \n"); // from fig 18
delt=(0.945*LMTD); // F
printf("\t delt is : %.2f F \n",delt);
X=((delt1)/(delt2));
printf("\t ratio of two local temperature difference is : %.3f \n",X);
Fc=0.42; // from fig.17
Kc=0.20; // crude oil controlling
Tc=((T2)+(T1))/2; // caloric temperature of hot fluid,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+(t2))/2; // caloric temperature of cold fluid,F
printf("\t caloric temperature of cold fluid is : %.1f F \n",tc);
printf("\t hot fluid:shell side,distilled water \n");
ID=15.25; // in
C=0.1875; // clearance
B=12; // baffle spacing,in
PT=0.9375;
as=((ID*C*B)/(144*PT)); // flow area,ft^2,using eq.7.1
printf("\t flow area is : %.3f ft^2 \n",as);
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2),using eq.7.2
printf("\t mass velocity is : %.1e lb/(hr)*(ft^2) \n",Gs);
mu1=0.81*2.42; // at 89F,lb/(ft)*(hr), from fig.14
```

```
De=0.55/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
jH=73; // from fig.28
c=1; // Btu/(lb)*(F),at 89F,from fig.table 4
k=0.36; // Btu/(hr)*(ft^2)*(F/ft), from table 4
Pr=((c)*(mu1)/k)^(1/3); // prandelt number raised to power 1/3
printf("\t Pr is : %.3f \n",Pr);
ho=((jH)*(k/De)*(Pr)); // using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.2e Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,raw water \n");
Nt=160;
n=2; // number of passes
L=16; //ft
at1=0.334; // flow area, in^2,from table 10
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.3e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*62.5));
printf("\t V is %.1f fps \n",V);
mu2=0.92*2.42; // at 77.5F,lb/(ft)*(hr)
D=0.65/12; //ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=1350*0.99; //using fig.25,Btu/(hr)*(ft^2)*(F)
ID=0.65; // ft
OD=0.75; //ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
Uc=((hio)*(ho)/(hio+ho)); // clean overall
```

```
coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
printf("\t  when  both. film coefficients are high the thermal resistance of the tube metal is not necessarily insignificant as assumed in the derivation of Eq. (6.38). For a steel 1.8 BWG tube Rm= 0.00017 and for copper Rm= 0.000017 \n");
A2=0.1963; // actual surface supplied for each tube,ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
f=0.0019; // friction factor for reynolds number 16200, using fig.29
s=1; // for reynolds number 25300,using fig.6
Ds=15.25/12; // ft
phys=1;
N=(12*L/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t allowable delPs is 10 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.00019; // friction factor for reynolds number 36400, using fig.26
s=1;
phyt=1;
D=0.054; // ft
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(
```

```
   phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.33; // X1=((V^2)/(2*g)), for Gt 1060000,using
   fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPT is 10 psi \n");
//end
```

Scilab code Exa 7.5 Calculation of the Optimum Outlet water Temperature

```
printf("\t example 7.5 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=175; // inlet hot fluid,F
T2=150; // outlet hot fluid,F
t1=85; // inlet cold fluid,F
delt1=T2-t1; //F
printf("\t delt1 is : %.0f F \n",delt1);
U=15; // assumption,Btu/(hr)*(ft^2)*(F)
theta=8000; // operating hours,hr
CW=(0.01/8300); // water cost,$/lb
printf("\t For annual charges assume 20 per cent
   repair and maintenanc.e and 10 per cent
   depreciation \n");
CF=(0.3*4); // annual fixed charges/ft^2
c=1; // Btu/(lb)*(F)
X=((U)*(theta)*(CW)/(CF*c));
printf("\t X is : %.4f \n",X);
Y=((T1-T2)/delt1);
printf("\t Y is : %.2f \n",Y);
A=0.96; // A=(delt2/delt1), from fig 7.24
delt2=0.96*delt1;
```

```
printf("\t delt2 is : %.1f F \n",delt2);
t2=T1-delt2; // F
printf("\t t2 is : %.1f F \n",t2);
//end
```

Scilab code Exa 7.6 `Calculation of a Phosphate Solution Cooler`

```
printf("\t example 7.6 \n");
printf("\t approximate values are mentioned in the book \n");
T1=150; // inlet hot fluid ,F
T2=90; // outlet hot fluid ,F
t1=68; // inlet cold fluid ,F
t2=90; // outlet cold fluid ,F
W=20160; // lb/hr
w=41600; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for solution \n");
c=(0.3*0.19)+(0.7*1); // Btu/(lb)*(F), bcoz of 30 percent of solution
Q1=((W)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for solution is : %.2e Btu/hr \n",Q1);
printf("\t for water \n");
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for water is : %.2e Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
R=((T1-T2)/(t2-t1));
```

```
printf("\t R is : %.2f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.3f \n",S);
printf("\t FT is 0.81 \n"); // from fig 18
delt=(0.81*LMTD); // F
printf("\t delt is : %.1f F \n",delt);
Tc=((T2)+(T1))/2; // caloric temperature of hot fluid,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+(t2))/2; // caloric temperature of cold fluid,F
printf("\t caloric temperature of cold fluid is : %.1f F \n",tc);
printf("\t hot fluid:shell side,phosphate solution \n");
ID=10.02; // in
C=0.25; // clearance
B=2; // baffle spacing,in
PT=1;
as=((ID*C*B)/(144*PT)); // flow area,ft^2,using eq.7.1
printf("\t flow area is : %.4f ft^2 \n",as);
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2),using eq.7.2
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=1.20*2.42; // at 120F,lb/(ft)*(hr), from fig.14
De=0.95/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.3e \n",Res);
jH=71; // from fig.28
c=1; // Btu/(lb)*(F),at 120F,from fig.table 4
k=0.33; // Btu/(hr)*(ft^2)*(F/ft), from table 4
Pr=((0.757)*(mu1)/k)^(1/3); // prandelt number raised to power 1/3
printf("\t Pr is : %.3f \n",Pr);
ho=((jH)*(k/De)*(Pr)); // using eq.6.15,Btu/(hr)*(ft
```

```
^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,raw water \n");
Nt=52;
n=2; // number of passes
L=16; //ft
at1=0.302; // flow area, in^2,from table 10
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.4f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*62.5));
printf("\t V is %.1f fps \n",V);
mu2=0.91*2.42; // at 79F,lb/(ft)*(hr),from table 14
D=(0.62/12); // from table 10
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=800*1; //using fig.25,Btu/(hr)*(ft^2)*(F)
ID=0.62; // ft
OD=0.75; //ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
A2=0.1963; // actual surface supplied for each tube,ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
```

```
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.5f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
f=0.0019; // friction factor for reynolds number 15750, using fig.29
s=1.3; // for reynolds number 25300,using fig.6
Ds=10.02/12; // ft
phys=1;
N=(12*L/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t allowable delPs is 10 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.00023; // friction factor for reynolds number 17900, using fig.26
s=1;
phyt=1;
D=0.0517; // ft
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.08; // X1=((V^2)/(2*g)), for Gt 1060000,using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPT is 10 psi \n");
//end
```

Scilab code Exa 7.7 The Optimum Use of Exhaust and Process Steam

```
printf("\t example 7.7 \n");
printf("\t approximate values are mentioned in the
   book \n");
U=50; // Btu/(hr)*(ft^2)*(F)
TP=328; // F
TE=228; // F
CP=(0.30/(888.8*1000));
CE=(0.05/(960*1000));
CF=1.20;
theta=8000; // annual hours
X=((CF*(TP-TE))/((CP-CE)*U*theta)); // from eq 7.53
printf("\t X is : %.9f \n",X);
a=(1); // coefficient of t^2
b=(-556); // coefficient of t
c=(74784-X); // constant
printf("\t coefficient of t^2 is : %.2f \n",a);
printf("\t coefficient of t is : %.2f \n",b);
printf("\t constant term is : %.9f \n",c);
P=poly([c b a], 't','c');
t=roots(P);
printf("\t t is :%.0f \n",t);
printf("\t t cannot be greater than 328F \n \t t is
   218F \n");
//end
```

Scilab code Exa 7.8 Calculation of a Sugar solution Heater without Baflles

```
printf("\t example 7.8 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=228; // inlet hot fluid,F
T2=228; // outlet hot fluid,F
t1=100; // inlet cold fluid,F
t2=122; // outlet cold fluid,F
W=200000; // lb/hr
```

```
w=3950; // lb/hr
printf(”\t 1.for heat balance \n”);
printf(”\t for solution \n”);
c=(0.2*0.30)+(0.8*1); // bcoz of 20 percent solution,Btu/(lb)*(F)
Q1=((W)*(c)*(t2-t1)); // Btu/hr
printf(”\t total heat required for solution is : %.2e Btu/hr \n”,Q1);
printf(”\t for steam \n”);
l=960.1; // latent heat of condensation,Btu/(lb)
Q=((w)*(l)); // Btu/hr
printf(”\t total heat required for steam is : %.2e Btu/hr \n”,Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf(”\t delt1 is : %.0f F \n”,delt1);
printf(”\t delt2 is : %.0f F \n”,delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf(”\t LMTD is :%.1f F \n”,LMTD);
R=((T1-T2)/(t2-t1));
printf(”\t R is : %.0f \n”,R);
delt=(LMTD); // when R=0,F
printf(”\t delt is : %.1f F \n”,delt);
printf(”\t The steam coefficient will be very great compared with that for the sugar solution , and the tube wall will be considerably nearer 228 F than the caloric temperature of the fluid . Obtain Fc from U1 and U0 Failure to correct for wall effects , however , will keep the heater calculation on the safe side .\n”);
ta=111; //F
Ta=228; //f
printf(”\t hot fluid:tube side,steam \n”);
Nt=76;
n=2; // number of passes
L=16; //ft
at1=0.302; // flow area , in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq
```

```
.7.48
printf("\t flow area is : %.4f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2elb/(hr)*(ft^2) \n",Gt);
mu2=0.0128*2.42; // at 228F,lb/(ft)*(hr)
D=(0.62/12); // from table 10,ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hio=1500; // for condensation of steam
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
printf("\t cold fluid:shell side,sugar solution \n");
ID=12; // in
d=0.75/12; // diameter of tube,ft
Nt=76; // number of tubes
as=((3.14*(12^2)/4)-(76*3.14*(0.75^2)/4))/144; // flow area,ft^2
printf("\t flow area is : %.2f ft^2 \n",as);
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=1.30*2.42; // at 111F,lb/(ft)*(hr), from fig.14
De=((4*as)/(Nt*3.14*d)); // from eq.6.3,ft
printf("\t De is : %.3f ft \n",De);
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
jH=61.5; // from fig.24, tube side data
c=0.86; // Btu/(lb)*(F),at 111F,from fig.4
k=0.333; // Btu/(hr)*(ft^2)*(F/ft)
Pr=((c)*(mu1)/k)^(1/3); // prandelt number raised to power 1/3
printf("\t Pr is : %.0f \n",Pr);
Ho=((jH)*(k/De)*(Pr)); // H0=(h0/phya),using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Ho);
```

```
muw=0.51*2.42; //  at 210F,lb/(ft)*(hr), from fig.14
phys=(mu1/muw)^0.14;
printf("\t phys is : %.2f \n",phys); // from fig.24
ho=(Ho)*(phys); // from eq.6.36
printf("\t Correct h0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
tw=(ta)+(((hio)/(hio+Ho))*(Ta-ta)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
A2=0.1963; // actual surface supplied for each tube, ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(LMTD)));
printf("\t actual design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for inner pipe \n");
f=0.000155; // friction factor for reynolds number 82500, using fig.26
s=0.0008;
phyt=1;
D=0.0517;
delPt=((f*(Gt^2)*(L)*(2))/(5.22*(10^10)*(D)*(s)*(phyt)))/2; // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
printf("\t pressure drop  for annulus \n");
De1=((4*as)/((Nt*3.14*d)+(3.14*1))); // from eq.6.4, ft
printf("\t De1 is : %.3f ft \n",De1);
Res1=(De1*Gs/mu1); // from eq 7.3
printf("\t Res1 is : %.2e \n",Res1);
f=0.00025; // friction factor, using fig.26
```

```
s=1.08; // for reynolds number 25300,using fig.6
delPs=((f*(Gs^2)*(L)*(1))/(5.22*(10^10)*(De1)*(s)*(phys))); // using eq.7.44,psi
printf(”\t delPs is : %.2f psi \n”,delPs);
//end
```

Scilab code Exa 7.9 `Outlet Temperatures`

```
printf(”\t example 7.9 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=390; // F
t1=100; // F
U=69.3; // Btu/(hr)*(ft^2)*(F)
A=662; // ft^2
W=43800; // lb/hr
w=149000; // lb/hr
C=0.60; // Btu/(lb)*(F)
c=0.49; // Btu/(lb)*(F)
X=((U*A)/(w*c));
printf(”\t X is : %.2f \n”,X);
R=((w*c)/(W*C));
printf(”\t R is : %.2f \n”,R);
S=0.265; // from fig 7.25, by comparing X an R
t2=(t1)+((0.265)*(T1-t1)); // S=((t2-t1)/(T1-t1))
printf(”\t t2 is : %.0f F \n”,t2);
T2=((T1)-((R)*(t2-t1))); // R=((T1-T2)/(t2-t1))
printf(”\t T2 is : %.0f F \n”,T2);
// end
```

Chapter 8

Flow Arrangements for increased heat recovery

Scilab code Exa 8.1 `Calculation of a Oil Cooler`

```
printf("\t example 8.1 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=358; // inlet hot fluid ,F
T2=100; // outlet hot fluid ,F
t1=90; // inlet cold fluid ,F
t2=120; // outlet cold fluid ,F
W=49600; // lb/hr
w=233000; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for oil \n");
c=0.545; // Btu/(lb)*(F)
Q=((W)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for oil is : %.2e Btu
   /hr \n",Q);
printf("\t for water \n");
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for water is : %.2e
```

```
Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.1f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.3f \n",S);
printf("\t FT is 0.93 \n"); // from fig 19 for 2-4 exchanger
delt=(0.93*LMTD); // F
printf("\t delt is : %.1f F \n",delt);
X=((delt1)/(delt2));
printf("\t ratio of two local temperature difference is : %.3f \n",X);
Fc=0.25; // from fig.17
Kc=0.47; // crude oil controlling
Tc=((T2)+((Fc)*(T1-T2))); // caloric temperature of hot fluid,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+((Fc)*(t2-t1))); // caloric temperature of cold fluid,F
printf("\t caloric temperature of cold fluid is : %.0f F \n",tc);
printf("\t hot fluid:shell side,oil \n");
ID=35; // in
C=0.25; // clearance
B=7; // baffle spacing,in
PT=1.25;
as=((ID*C*B)/(144*PT))/2; // flow area,ft^2,from eq 7.1
printf("\t flow area is : %.2f ft^2 \n",as);
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2),from eq 7.2
```

```
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=1.12*2.42; // at 165F,lb/(ft)*(hr), from fig.14
De=0.99/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.1e \n",Res);
jH=52.5; // from fig.28
Z=0.2; // Z=(k)*(Pr*(1/3)) prandelt number
Ho=((jH)*(1/De)*(Z)); // H0=(h0/phys),using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Ho);
printf("\t cold fluid:inner tube side,water \n");
Nt=454;
n=6; // number of passes
L=12; //ft
at1=0.455; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*62.5)); // fps
printf("\t V is : %.2f fps \n",V);
mu2=0.73*2.42; // at 98F,lb/(ft)*(hr),from fig 14
D=(0.76/12); // ft,from table 10
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=1010*0.96; // using fig 25,Btu/(hr)*(ft^2)*(F)
printf("\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=0.76; // ft
OD=1; //ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
tw=(tc)+(((Ho)/(hio+Ho))*(Tc-tc)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
```

```
muw=1.95*2.42; // lb/(ft)*(hr), from fig.14
phys=(mu1/muw)^0.14;
printf("\t phys is : %.2f \n",phys); // from fig.24
ho=(Ho)*(phys); // from eq.6.36
printf("\t Correct h0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
A2=0.2618; // actual surface supplied for each tube,ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
Q=6980000; // taking rounded value,Btu/hr
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
f=0.00215; // friction factor for reynolds number 8900, using fig.29
s=0.82; // for reynolds number 25300,using fig.6
Ds=35/12; // ft
N=(12*L/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(2*N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPs is : %.0f psi \n",delPs);
printf("\t allowable delPs is 10 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.000195; // friction factor for reynolds number 34900, using fig.26
s=1;
D=0.0633; //ft
phyt=1;
```

```
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf(”\t delPt is : %.1f psi \n”,delPt);
X1=0.13; // X1=((V^2)/(2*g)), for Gt 1060000,using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf(”\t delPr is : %.1f psi \n”,delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf(”\t delPT is : %.1f psi \n”,delPT);
printf(”\t allowable delPT is 10 psi \n”);
//end
```

Scilab code Exa 8.2 calculation of an Acetone Acetic Acid Exchanger

```
printf(”\t example 8.2 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=250; // inlet hot fluid,F
T2=100; // outlet hot fluid,F
t1=90; // inlet cold fluid,F
t2=150; // outlet cold fluid,F
W=60000; // lb/hr
w=168000; // lb/hr
printf(”\t 1.for heat balance \n”);
printf(”\t for acetone \n”);
c=0.57; // Btu/(lb)*(F)
Q=((W)*(c)*(T1-T2)); // Btu/hr
printf(”\t total heat required for acetone is : %.2e Btu/hr \n”,Q); // calculation mistake in problem
printf(”\t for acetic acid \n”);
c=0.51; // Btu/(lb)*(F)
Q1=((w)*(c)*(t2-t1)); // Btu/hr
printf(”\t total heat required for acetic acid is : %.2e Btu/hr \n”,Q1); // calculation mistake in problem
```

```
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.1f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.3f \n",S);
printf("\t FT is 0.88 \n"); // from fig 20,for 3-6
    exchanger
delt=(0.88*LMTD); // F
printf("\t delt is : %.1f F \n",delt);
Tc=((T2)+(T1))/2; // caloric temperature of hot
    fluid ,F
printf("\t caloric temperature of hot fluid is : %.0
    f F \n",Tc);
tc=((t1)+(t2))/2; // caloric temperature of cold
    fluid ,F
printf("\t caloric temperature of cold fluid is : %
    .0f F \n",tc);
printf("\t hot fluid:shell side,acetone \n");
ID=21.25; // in
C=0.25; // clearance
B=5; // baffle spacing,in
PT=1;
as=((ID*C*B)/(144*PT)); // flow area,ft^2
printf("\t flow area is : %.3f ft^2 \n",as);
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n
    ",Gs);
mu1=0.20*2.42; // at 175F,lb/(ft)*(hr), from fig.14
De=0.95/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
phys=1;
jH=137; // from fig.28
```

```
c=0.63; // Btu/(lb)*(F),at 175F,from fig.2
k=0.095; // Btu/(hr)*(ft^2)*(F/ft), from table 4
Pr=((c)*(mu1)/k)^(1/3); // prandelt number raised to power 1/3
printf("\t Pr is : %.3f \n",Pr);
ho=((jH)*(k/De)*(Pr)*1); // using eq.6.15b,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,acetic acid \n");
Nt=270;
n=2; // number of passes
L=16; //ft
at1=0.268; // flow area, in^2,from table 10
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.1e lb/(hr)*(ft^2) \n",Gt);
mu2=0.85*2.42; // at 120F,lb/(ft)*(hr)
D=(0.584/12); // ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
jH=56; // from fig.24
c=0.51; // Btu/(lb)*(F),at 120F,from fig.2
k=0.098; // Btu/(hr)*(ft^2)*(F/ft), from table 4
phyt=1;
Pr=((c)*(mu2)/k)^(1/3); // prandelt number raised to power 1/3
printf("\t Pr is : %.3f \n",Pr);
hi=((jH)*(k/D)*(Pr)*(1^0.14)); // using eq.6.15a,Btu/(hr)*(ft^2)*(F)
printf("\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=0.584; // ft
OD=0.75; //ft
hio=((hi)*(ID/OD)); //Hio=(hio/phyp), using eq.6.5
```

```
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient ,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
A2=0.1963; // actual surface supplied for each tube ,ft^2,from table 10
A=3*(Nt*L*A2); // ft^2
printf("\t total surface area is : %.2e ft^2 \n",A);
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
f=0.00155; // friction factor for reynolds number 52200, using fig.29
s=0.79; // for reynolds number 25300,using table.6
Ds=21.25/12; // ft
N=(12*L/B)+1; // number of crosses ,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(3*N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44 ,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t allowable delPs is 10 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.00024; // friction factor for reynolds number 158000, using fig.26
s=1.07;
D=0.0487; // ft
delPt=((f*(Gt^2)*(L)*(n*3))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45 ,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.063; // X1=((V^2)/(2*g)), for Gt 1060000,using fig.27
delPr=(3)*((4*n*X1)/(s)); // using eq.7.46 ,psi
```

```
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPT is 10 psi \n");
//end
```

Chapter 9

Gases

Scilab code Exa 9.1 `Calculation of an Ammonia Compressor Aftercooler`

```
printf("\t example 9.1 \n");
printf("\t approximate values are mentioned in the book \n");
T1=245; // inlet hot fluid ,F
T2=95; // outlet hot fluid ,F
t1=85; // inlet cold fluid ,F
t2=95; // outlet cold fluid ,F
W=9872; // lb/hr
w=78500; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for ammonia gas \n");
c=0.53; // Btu/(lb)*(F)
Q=((W)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for ammonia gas is : %.2e Btu/hr \n",Q);
printf("\t for water \n");
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for water is : %.2f Btu/hr \n",Q);
delt1=T2-t1; //F
```

```
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.0f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.4f \n",S);
printf("\t FT is 0.837 \n"); // from fig 18
delt=(0.837*LMTD); // F
printf("\t delt is : %.1f F \n",delt);
Tc=((T2)+(T1))/2; // caloric temperature of hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+(t2))/2; // caloric temperature of cold fluid ,F
printf("\t caloric temperature of cold fluid is : %.0f F \n",tc);
printf("\t hot fluid:shell side,ammonia at 83psia \n");
ID=23.25; // in
C=0.1875; // clearance
B=12; // baffle spacing,in
PT=0.937;
as=((ID*C*B)/(144*PT)); // flow area,ft^2,from eq 7.1
printf("\t flow area is : %.3f ft^2 \n",as);
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2),from eq 7.2
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=0.012*2.42; // at 170F,lb/(ft)*(hr), from fig.15
De=0.55/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
jH=118; // from fig.28
```

```
k=0.017; // Btu/(hr)*(ft^2)*(F/ft),from table 5
Z=0.97; // Z=(Pr*(1/3)) prandelt number
ho=((jH)*(k/De)*(Z)*1); // using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,water \n");
Nt=364;
n=8; // number of passes
L=8; //ft
at1=0.302; // flow area, in^2,from table 10
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.4f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*62.5)); // fps
printf("\t V is : %.2f fps \n",V);
mu2=0.82*2.42; // at 90F,lb/(ft)*(hr),from fig 14
D=(0.62/12); // ft,from table 10
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=900; // using fig 25,Btu/(hr)*(ft^2)*(F)
printf("\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=0.62; // ft
OD=0.75; //ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
A2=0.1963; // actual surface supplied for each tube,ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
```

```
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.3f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
f=0.00162; // friction factor for reynolds number 40200, using fig.29
Ds=23.25/12; // ft
phys=1;
N=(12*L/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
rowgas=0.209;
printf("\t rowgas is %.3f lb/ft^3 \n",rowgas);
s=rowgas/62.5;
printf("\t s is %.5f \n",s);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPs is : %.0f psi \n",delPs);
printf("\t allowable delPs is 2 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.000225; // friction factor for reynolds number 21400, using fig.26
s=1;
D=0.0517; //ft
phyt=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.090; // X1=((V^2)/(2*g)), for Gt 1060000,using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPT is 10 psi \n");
//end
```

Scilab code Exa 9.2 `Calculation of the Heat Load for an Air Intercooler`

```
printf("\t example 9.2 \n");
printf("\t approximate values are mentioned in the
   book \n");
V1=4670; // inlet air volume,cfm
Pp=0.8153; // Saturation partial pressure of water
   at 95F,psi,from table 7
Ps=404.3;// Saturation specific volume of water at
   95F,ft^3/lb, from table 7
printf("\t The air and water both occupy the same
   volume at their respective partial pressures \n")
   ;
Vw1=(V1*60/Ps); // water entering per hr,lb
printf("\t volume of water entering is : %.0f lb \n"
   ,Vw1);
printf("\t for first stage \n");
c=2.33; // compression ratio
P1=14.7; // psi
P2=(P1*c); // (c=(P2/P1)),psi
printf("\t P2 is : %.1f psi \n",P2);
gama=1.4; // for air
T1abs=95; // F
T2absr=((T1abs+460)*(P2/P1)^((gama-1)/gama));
printf("\t T2absr is : %.0f R \n",T2absr);
T2abs=(T2absr-459.67); // F
printf("\t T2abs is : %.0f F \n",T2abs);
printf("\t for intercooler \n");
V2=(V1*60*P1/P2); // ft^3/hr
printf("\t final gas volume is : %.1e ft^3/hr \n",V2
   );
Vw2=(V2/Ps); // water remaining in air, lb/hr
printf("\t water remaining in air is : %.0f lb/hr \n
   ",Vw2);
```

```
C=(Vw1-Vw2); // condensation in inter cooler, lb/hr
printf("\t condensation in inter cooler is : %.0f lb/hr \n",C);
Vs=14.8; // Specific volume of atmospheric air,ft^3/lb
printf("\t Specific volume of atmospheric air is : %.1f ft^3/lb \n",Vs);
Va=(V1*60/Vs); // air in inlet gas, lb/hr
printf("\t air in inlet gas is : %.2e lb/hr\n",Va);
printf("\t heat load(245 to 95F) \n)");
printf("\t sensible heat \n");
Qair=((Va)*(0.25)*(245-T1abs)); // Btu/hr
printf("\t Qair is : %.2e Btu/hr \n",Qair);
Qwaters=(Vw1*0.45*(245-T1abs)); // Btu/hr
printf("\t Qwaters is : %.2e Btu/hr \n",Qwaters);
printf("\t latent heat \n");
l=1040.1; // latent heat
Qwaterl=(C*l); // Btu/hr
printf("\t Qwaterl is : %.2e Btu/hr \n",Qwaterl);
Qt1=Qair+Qwaters+Qwaterl;
printf("\t total heat is : %.3e Btu/hr \n",Qt1);
printf("\t for second stage \n");
c=2.33; // compression ratio
P3=(P2*c); // (c=(P3/P1)),psi
printf("\t P3 is : %.1f psi \n",P3);
V3=(V1*60*P1/P3); // ft^3/hr
printf("\t final gas volume is : %.2e ft^3/hr \n",V3);
Vw3=(V3/Ps); // water remaining in air, lb/hr
printf("\t water remaining in air is : %.1f lb/hr \n",Vw3);
C1=(297-Vw3); // condensation in inter cooler, lb/hr
printf("\t condensation in inter cooler is : %.1f lb/hr \n",C1);
printf("\t heat load(245 to 95F) \n)");
printf("\t sensible heat \n");
Qair=(Va*0.25*(245-T1abs)); // Btu/hr
printf("\t Qair is : %.2e Btu/hr \n",Qair);
```

```
Qwaters=(Vw2*0.44*(245-T1abs)); // Btu/hr
printf(”\t Qwater is : %.2e Btu/hr \n”,Qwaters);
printf(”\t latent heat \n”);
l=1040.1; // latent heat
Qwaterl=(C1*l); // Btu/hr, calculation mistake in book
printf(”\t Qwater is : %.2e Btu/hr \n”,Qwaterl);
Qt1=Qair+Qwaters+Qwaterl;
printf(”\t total heat is : %.3e Btu/hr \n”,Qt1);
// end
```

Scilab code Exa 9.3 Calculation of the Dew Point after Compression

```
printf(”\t example 9.3 \n”);
printf(”\t approximate values are mentioned in the book \n”);
Va=18900; // air in inlet gas
Vw1=692; // water entering
Ma=(Va/29); // moles
Mw=(Vw1/18); // moles
M=(Ma+Mw); // moles
printf(”\t total number of moles re : %.1f \n”,M);
printf(”\t Moles of air is : %.0f \n”,Ma);
printf(”\t Moles of water is : %.1f \n”,Mw);
printf(”\t after compression \n”);
P=34.2; // pressure, psi
pw=(Mw/M)*(P); // partial pressure
printf(”\t partial pressure is :%.1f psi \n”,pw);
Td=124; // F, table table 7
printf(”\t dew point is : %.0f F \n”,Td);
// end
```

Chapter 10

Streamlineflow and steam convection

Scilab code Exa 10.1 `Crude Oil Heater`

```
printf("\t example 10.1 \n");
printf("\t approximate values are mentioned in the book \n");
T1=250; // inlet hot fluid,F
T2=250; // outlet hot fluid,F
t1=95; // inlet cold fluid,F
t2=145; // outlet cold fluid,F
W=16000; // lb/hr
w=410; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for crude \n");
c=0.485; // Btu/(lb)*(F)
Q=((W)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for crude is : %.2e Btu/hr \n",Q);
printf("\t for steam \n");
l=945.5; // Btu/(lb)
Q=((w)*(l)); // Btu/hr
printf("\t total heat required for steam is : %.2e
```

```
    Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
printf("\t On the assumption that the fluids are
    mixed between passes, each pass must be solved
    independently. Since only two passes are present
    in this exchanger, it is simply a matter of
    assuming the temperature at the end of the first
    pass. More than half the heat load must be
    transferred in the first pass; therefore assume
    ti at the end of the first pass is 125 F  \n");
ti=125; // F
tc=((t1)+(ti))/2; // caloric temperature of cold
    fluid ,F
printf("\t caloric temperature of cold fluid is : %
    .0f F \n",tc);
printf("\t hot fluid:shell side,steam \n");
ho=(1500); // condensation of steam Btu/(hr)*(ft^2)
    *(F)
printf("\t individual heat transfer coefficient is :
     %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,crude \n");
Nt=86;
n=2; // number of passes
L=12; //ft
at1=0.594; // flow area, in^2,from table 10
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq
    .7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(W/(.177)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n
    ",Gt);
mu2=2.95*2.42; // at 145F,lb/(ft)*(hr)
D=(0.87/12); // ft
```

```
Ret1=((D)*(Gt)/mu2); // reynolds number
printf(”\t reynolds number is : %.0f \n”,Ret1);
mu3=4.8*2.42; // at 110F,lb/(ft)*(hr)
D=(0.87/12); // ft
Ret2=((D)*(Gt)/mu3); // reynolds number
printf(”\t reynolds number is : %.0f \n”,Ret2);
c=0.485; // Btu/(lb)*(F),at 120F,from fig.2
k=0.0775; // Btu/(hr)*(ft^2)*(F/ft), from table 4
Pr=((c)*(mu3)/k); // prandelt number
printf(”\t prandelt number is : %.1f \n”,Pr);
Hi=((1.86)*(k/D)*((Ret2*(D/L)*Pr)^(1/3))); // using eq.6.1,Btu/(hr)*(ft^2)*(F)
printf(”\t Hi is : %.1f Btu/(hr)*(ft^2)*(F) \n”,Hi);
muw=1.2*2.42; // lb/(ft)*(hr),at 249F from fig.14
phyt=(mu3/muw)^0.14;
printf(”\t phyt is : %.1f \n”,phyt); // from fig.24
hi=(Hi)*(phyt); // from eq.6.37
printf(”\t Correct hi to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n”,hi);
tp=(tc)+(((ho)/(hi+ho))*(T1-tc)); // from eq.5.31
printf(”\t tp is : %.0f F \n”,tp);
delt=tp-tc; //F
printf(”\t delt is : %.0f F \n”,delt);
Ai1=0.228 // internal surface per foot of length,ft
Ai=(Nt*L*Ai1/2); // ft^2
printf(”\t total surface area is : %.1f ft^2 \n”,Ai);
delt3=((hi*Ai*delt)/(W*c)); // delt3=ti-t1, F
printf(”\t delt3 is : %.1f F \n”,delt3);
ti=t1+delt3; // F
printf(”\t ti is : %.1f F \n”,ti);
printf(”\t The oil now enters the second pass at 126.9 F  \n”);
// end
```

Scilab code Exa 10.2 Kerosene Heater

```
printf("\t example 10.2 \n");
printf("\t approximate values are mentioned in the book \n");
T1=250; // inlet hot fluid,F
T2=250; // outlet hot fluid,F
t1=95; // inlet cold fluid,F
t2=145; // outlet cold fluid,F
W=16000; // lb/hr
w=423; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for kerosene \n");
c=0.5; // Btu/(lb)*(F)
Q=((W)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for kerosene is : %.0f Btu/hr \n",Q);
printf("\t for steam \n");
l=945.5; // Btu/(lb)
Q=((w)*(l)); // Btu/hr
printf("\t total heat required for steam is : %.2e Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
tc=((t1)+(t2))/2; // caloric temperature of cold fluid,F
printf("\t caloric temperature of cold fluid is : %.0f F \n",tc);
printf("\t hot fluid:shell side,steam \n");
ho=(1500); // condensation of steam Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,kerosene \n");
```

```
Nt=86;
n=2; // number of passes
L=12; //ft
at1=0.594; // flow area, in^2,from table 10
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(W/(.177)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu2=1.36*2.42; // at 145F,lb/(ft)*(hr)
D=(0.87/12); // ft
Ret1=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.0f \n",Ret1);
mu3=1.75*2.42; // at 120F,lb/(ft)*(hr)
D=(0.87/12); // ft
Ret2=((D)*(Gt)/mu3); // reynolds number
printf("\t reynolds number is : %.1e \n",Ret2);
Z1=331; // Z1=(L*n/D)
jH=3.1; // from fig 24
mu4=1.75; // cp and 40 API
Z2=0.24; // Z2=((k)*(c*mu4/k)^(1/3)), from fig 16
Hi=((jH)*(1/D)*(Z2)); // using eq.6.15a,Btu/(hr)*(ft^2)*(F)
printf("\t Hi is : %.2f Btu/(hr)*(ft^2)*(F) \n",Hi);
ID=0.87; // ft
OD=1; //ft
Hio=(Hi*(ID/OD)); //Btu/(hr)*(ft^2)*(F), from eq.6.5
printf("\t Hio is : %.2f Btu/(hr)*(ft^2)*(F) \n",Hio);
tw=(tc)+(((ho)/(Hio+ho))*(T1-tc)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
muw=1.45; // lb/(ft)*(hr),at 249F from fig.14
phyt=(mu3/muw)^0.14;
printf("\t phyt is : %.1f \n",phyt); // from fig.24
hio=(Hio)*(phyt); // from eq.6.37
printf("\t Correct hio to the surface at the OD is : %.1f Btu/(hr)*(ft^2)*(F) \n",hio);
```

```
delt=tw-tc; //F
printf("\t delt is : %.0f F \n",delt);
printf("\t Since the kerosene has a viscosity of only 1.75 cp at the caloric temperature and delt =129F, free convection should be investigated. \n");
s=0.8;
row=50; // lb/ft^3, from fig 6
s1=0.810; // at 95F
s2=0.792; // at 145F
bita=((s1^2-s2^2)/(2*(t2-t1)*s1*s2)); // /F
printf("\t beta is : %.6f /F \n",bita);
G=((D^3)*(row^2)*(bita)*(delt)*(4.18*10^8)/(mu3^2));
printf("\t G is : %.1e  \n",G);
psy=((2.25)*(1+(0.01*G^(1/3)))/(log10(Ret2)));
printf("\t psy is : %.2f  \n",psy);
hio1=(hio*psy);
printf("\t corrected hio1 is : %.1f Btu/(hr)*(ft^2)*(F) \n",hio1);
Uc=((hio1)*(ho)/(hio1+ho)); // clean overall coefficient ,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
A2=0.2618; // actual surface supplied for each tube, ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.2f (hr)*(ft^2)*(F)/Btu \n",Rd);
// end
```

Scilab code Exa 10.3 `Gas Oil Heater Using Cores`

```
printf(”\t example 10.3 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=250; // inlet hot fluid,F
T2=250; // outlet hot fluid,F
t1=105; // inlet cold fluid,F
t2=130; // outlet cold fluid,F
w=50000; // lb/hr
W=622; // lb/hr
printf(”\t 1.for heat balance \n”);
printf(”\t for gas oil \n”);
c=0.47; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf(”\t total heat required for gas oil is : %.2e Btu/hr \n”,Q);
printf(”\t for steam \n”);
l=945.5; // Btu/(lb)
Q=((W)*(l)); // Btu/hr
printf(”\t total heat required for steam is : %.2e Btu/hr \n”,Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf(”\t delt1 is : %.0f F \n”,delt1);
printf(”\t delt2 is : %.0f F \n”,delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf(”\t LMTD is :%.1f F \n”,LMTD);
tc=((t1)+(t2))/2; // caloric temperature of cold fluid,F
printf(”\t caloric temperature of cold fluid is : %.1f F \n”,tc);
printf(”\t hot fluid:shell side,steam \n”);
ID=15.25; // in
C=0.25; // clearance
B=15; // baffle spacing,in
PT=1.25;
as=((ID*C*B)/(144*PT)); // flow area,ft^2, eq 7.1
```

```
printf("\t flow area is : %.3f ft^2 \n",as);
Gs=(6220/as); // mass velocity,lb/(hr)*(ft^2), calculation mistake
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=0.0314; // at 250F,lb/(ft)*(hr), from fig.15
De=0.060; // from fig.29,ft
Res=((De)*(Gs)/mu1); // reynolds number, calculation mistake
printf("\t reynolds number is : %.2e \n",Res);
ho=1500; //Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,crude oil \n");
d1=0.5; // in
d2=0.87; // in
at1=((3.14*(d2^2-d1^2))/4);
printf("\t at1 is : %.1f in^2 \n",at1);
Nt=86;
n=2; // number of passes
L=12; //ft
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.1e lb/(hr)*(ft^2) \n",Gt);
De=(d2^2-d1^2)/(12*d2);
printf("\t De is : %.4f ft \n",De);
mu2=16.7; // at 117F,lb/(ft)*(hr)
Ret=((De)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
jH=3.1; // from fig.24
Z=0.35; // Z=(K*(c*mu3/k)^(1/3)),Btu/(hr)(ft^2)(F/ft), at mu3=6.9cp and 28 API
Hi=((jH)*(1/De)*(Z)); //Hi=(hi/phyp),using eq.6.15a,Btu/(hr)*(ft^2)*(F)
```

```
printf("\t Hi is : %.1f Btu/(hr)*(ft^2)*(F) \n",Hi);
ID=0.87; // ft
OD=1; //ft
Hio=((Hi)*(ID/OD)); //Hio=(hio/phyp), using eq.6.5
printf("\t Correct Hi0 to the surface at the OD is : %.1f Btu/(hr)*(ft^2)*(F) \n",Hio);
muw=4.84; // lb/(ft)*(hr), from fig.14
phyt=(mu2/muw)^0.14;
printf("\t phyt is : %.2f \n",phyt); // from fig.24
hio=(Hio)*(phyt); // from eq.6.37
printf("\t Correct hi0 to the surface at the OD is : %.1f Btu/(hr)*(ft^2)*(F) \n",hio);
tw=(tc)+(((ho)/(hio+ho))*(T1-tc)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
A=270; // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(LMTD)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.3f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
f=0.0016; // friction factor for reynolds number 25300, using fig.29
s=0.00116; // for reynolds number 25300,using fig.6
Ds=15.25/12; // ft
phys=1;
N=(12*L/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(19600^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys)))/(2); // using eq.7.44,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t pressure drop  for inner pipe \n");
```

```
dt=(d2-d1)/(12); // ft
printf("\t dt is : %.4f ft \n",dt);
Ret2=(dt*Gt/mu2);
printf("\t Ret2 is : %.0f \n",Ret2);
f=0.00066; // friction factor for reynolds number 8220, using fig.26
phyt=1.35; // fig 6
printf("\t phyt is : %.2f \n",phyt);
s=0.85;
delPt=((f*(420000^2)*(L)*(n))/(5.22*(10^10)*(0.0309)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
printf("\t delPr is negligible \n");
//end
```

Scilab code Exa 10.4 Calculation of a Heating Bundle for an Aniline Storage Tank

```
printf("\t example 10.4 \n");
printf("\t approximate values are mentioned in the book \n");
t1=100; // F
t2=0; // F
T1abs=100+460; // R
T2abs=460; //R
delt=t1-t2;
printf("\t delt is : %.f F \n",delt);
hc=0.3*(delt^0.25); // convection loss , Btu/(hr)*(ft^2)*( F )
printf("\t convection loss is : %.2f Btu/(hr)(ft^2)(F) \n",hc);
e=0.8; // emissivity
hr=((0.173*e*((T1abs/100)^4-(T2abs/100)^4))/(T1abs-T2abs)); // radiation rate , from 4.32 , Btu/(hr)(ft^2)(F)
printf("\t radiation loss is : %.2f Btu/(hr)(ft^2)(F
```

```
) \n",hr);
hl=hc+hr; // combined loss, Btu/(hr)(ft^2)(F)
printf("\t combined loss is : %.1f Btu/(hr)(ft^2)(F) \n",hl);
D=5; // ft
L=12; // ft
A1=((2*3.14*D^2)/(4))+(3.14*D*L); // total tank area
printf("\t total tank area is : %.1f ft^2 \n",A1);
Q=(hl*A1*delt); // total heat loss
printf("\t total heat loss : %.2e Btu/hr \n",Q);
printf("\t This heat must be supplied by the pipe bundle,Assuming exhaust steam to be at 212 F  \n");
d0=1.32;
X=(delt/d0);
tf=((t1+212)/2); // F
printf("\t X is : %.0f \n",X);
printf("\t tf is : %.0f F \n",tf);
hio=48; // from fig 10.4, Btu/(hr)(ft^2)(F)
ho=1500; // condensation of steam,Btu/(hr)(ft^2)(F)
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=0.02; // dirt factor, (hr)(ft^2)(F)/Btu
UD=((Uc)/((1)+(Uc*Rd))); // design overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
A2=((Q)/((UD)*(212-100))); // total surface,ft^2
printf("\t total surface is : %.1f ft^2 \n",A2);
A3=2.06; // area/pipe
N=(A2/A3);
printf("\t number of pipes are : %.0f \n",N);
//end
```

Chapter 11

Calculations for process conditions

Scilab code Exa 11.1 `Calculation of a Straw Oil Naphtha Exchanger`

```
printf("\t example 11.1 \n");
printf("\t approximate values are mentioned in the book \n");
T1=340; // inlet hot fluid ,F
T2=240; // outlet hot fluid ,F
t1=200; // inlet cold fluid ,F
t2=230; // outlet cold fluid ,F
W=29800; // lb/hr
w=103000; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for straw oil \n");
c=0.58; // Btu/(lb)*(F)
Q=((W)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for straw oil is : %.2e Btu/hr \n",Q);
printf("\t for naphtha \n");
c=0.56; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for naphtha is : %.2e
```

```
    Btu/hr \n”,Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf(”\t delt1 is : %.0f F \n”,delt1);
printf(”\t delt2 is : %.0f F \n”,delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf(”\t LMTD is :%.1f F \n”,LMTD);
R=((T1-T2)/(t2-t1));
printf(”\t R is : %.1f \n”,R);
S=((t2-t1)/(T1-t1));
printf(”\t S is : %.3f \n”,S);
printf(”\t FT is 0.885 \n”); // from fig 18
delt=(0.885*LMTD); // F
printf(”\t delt is : %.1f F \n”,delt);
X=((delt1)/(delt2));
printf(”\t ratio of two local temperature difference
    is : %.3f \n”,X);
L=16;
Fc=0.405; // from fig.17
Kc=0.23; // crude oil controlling
Tc=((T2)+((Fc)*(T1-T2))); // caloric temperature of
   hot fluid,F
printf(”\t caloric temperature of hot fluid is : %.1
   f F \n”,Tc);
tc=((t1)+((Fc)*(t2-t1))); // caloric temperature of
   cold fluid,F
printf(”\t caloric temperature of cold fluid is : %
   .0f F \n”,tc);
UD1=70; // assume, from table 8a
A1=((Q)/((UD1)*(delt)));
printf(”\t A1 is : %.0f ft^2 \n”,A1);
a1=0.1963; // ft^2/lin ft
N1=(A1/(16*a1));
printf(”\t number of tubes are : %.0f \n”,N1);
N2=124; // assuming two tube passes, from table 9
A2=(N2*L*a1); // ft^2
printf(”\t total surface area is : %.1e ft^2 \n”,A2)
   ;
```

```
UD=((Q)/((A2)*(delt)));
printf("\t correct design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
printf("\t hot fluid:shell side,straw oil \n");
ID=15.25; // in
C=0.25; // clearance
B=3.5; // minimum baffle spacing,from eq 11.4,in
PT=1;
as=((ID*C*B)/(144*PT)); // flow area,from eq 7.1,ft^2
printf("\t flow area is : %.4f ft^2 \n",as);
Gs=(W/as); // mass velocity,from eq 7.2,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=3.63; // at 280.5F,lb/(ft)*(hr), from fig.14
De=0.95/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.0e \n",Res);
jH=46; // from fig.28
Z=0.224; // Z=(K*(c*mu3/k)^(1/3)),Btu/(hr)(ft^2)(F/ft), at mu3=1.5cp and 35 API
Ho=((jH)*(1/De)*(Z)); // H0=(h0/phya),using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Ho);
phys=1;
ho=(Ho)*(phys); // from eq.6.36
printf("\t Correct h0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,naphtha \n");
Nt=124;
n=2; // number of passes
L=16; //ft
at1=0.302; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
```

```
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf(”\t mass velocity is : %.2e lb/(hr)*(ft^2) \n”,Gt);
mu2=1.31; // at 212F,lb/(ft)*(hr)
D=0.0517; // ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf(”\t reynolds number is : %.2e \n”,Ret);
jH=102; // from fig.24
Z=0.167; // Z=(K*(c*mu3/k)^(1/3)),Btu/(hr)(ft^2)(F/ft), at mu4=0.54cp and 48 API
Hi=((jH)*(1/D)*(Z)); //Hi=(hi/phyp),using eq.6.15a,Btu/(hr)*(ft^2)*(F)
printf(”\t Hi is : %.0f Btu/(hr)*(ft^2)*(F) \n”,Hi);
ID=0.62; // ft
OD=0.75; //ft
Hio=((Hi)*(ID/OD)); //Hio=(hio/phyp), using eq.6.5
printf(”\t Correct Hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n”,Hio);
phyt=1;
hio=(Hio)*(phyt); // from eq.6.37
printf(”\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n”,hio);
printf(”\t pressure drop for annulus \n”);
f=0.00225; // friction factor for reynolds number 7000, using fig.29
s=0.76; // for reynolds number 7000,using fig.6
Ds=15.25/12; // ft
N=(12*L/B); // number of crosses,using eq.7.43
printf(”\t number of crosses are : %.0f \n”,N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf(”\t delPs is : %.1f psi \n”,delPs);
printf(”\t pressure drop for inner pipe \n”);
f=0.0002; // friction factor for reynolds number 31300, using fig.26
s=0.72;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
```

```
printf("\t delPt is : %.1f psi \n",delPt);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t The first trial is disqualified because of failure to meet the required dirt factor \n");
printf("\t Proceeding as above and carrying the viscosity correction and pressure drops to completion the new summary is given using a 17.25 in. ID shell with 166 tubes on two passes and a 3.5in. baffle space \n");
UD1=60; // assumption for 2 tube passes,3.5 baffle spacing and 17.25in ID
UC1=74.8;
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UC1);
UD2=54.2;
printf("\t correct design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD2);
Rd1=0.005;
printf("\t actual Rd is : %.3f (hr)*(ft^2)*(F)/Btu \n",Rd1);
delPs1=4.7;
printf("\t delPs is : %.1f psi \n",delPs1);
delPt1=2.1;
printf("\t delPt is : %.1f psi \n",delPt1);
//end
```

Scilab code Exa 11.2 Calculation of a Flue Gas Cooler

```
printf("\t example 11.2 \n");
```

```
printf("\t approximate values are mentioned in the book \n");
T1=350; // inlet hot fluid ,F
T2=160; // outlet hot fluid ,F
t1=100; // inlet cold fluid ,F
t2=295; // outlet cold fluid ,F
W=84438; // lb/hr
w=86357; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for lean oil \n");
c=0.56; // Btu/(lb)*(F)
Qh=((W)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for lean oil is : %.2e Btu/hr \n",Qh);
printf("\t for rich oil \n");
c=0.53; // Btu/(lb)*(F)
Qc=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for rich oil is : %.2e Btu/hr \n",Qc);
Q=(Qh+Qc)/(2);
printf("\t Q is : %.2e V \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.3f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.2f \n",S);
printf("\t FT is 0.875 \n"); // for 4-8 exchanger, from fig 21
delt=(0.875*LMTD); // F
printf("\t delt is : %.1f F \n",delt);
X=((delt1)/(delt2));
printf("\t ratio of two local temperature difference is : %.2f \n",X);
```

```
Fc=0.48; // from fig.17
Kc=0.32; // crude oil controlling
Tc=((T2)+((Fc)*(T1-T2))); // caloric temperature of hot fluid,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+((Fc)*(t2-t1))); // caloric temperature of cold fluid,F
printf("\t caloric temperature of cold fluid is : %.1f \n",tc);
UD1=50; // assume, from table 8a
A1=((Q)/((UD1)*(delt)));
printf("\t A1 is : %.2e ft^2 \n",A1);
a1=0.1963; // ft^2/lin ft
N1=(A1/(16*a1*2)); // 2-4 exchanger in series
printf("\t number of tubes are : %.0f \n",N1);
N2=580; // assuming six tube passes,31in ID, from table 9
A2=(N2*16*a1*2); // ft^2
printf("\t total surface area is : %.2e ft^2 \n",A2);
UD=((Q)/((A2)*(delt)));
printf("\t correct design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
printf("\t hot fluid:inner tube side,lean oil \n");
Nt=580;
n=6; // number of passes
L=16; //ft
at1=0.302; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(W/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu2=2.13; // at 212F,lb/(ft)*(hr)
D=0.0517; // ft
Ret=((D)*(Gt)/mu2); // reynolds number
```

```
printf(”\t reynolds number is : %.2e \n”,Ret);
jH=36.5; // from fig.24
Z=0.185; // Z=(K*(c*mu3/k)ˆ(1/3)),Btu/(hr)(ftˆ2)(F/ft), at mu4=0.88cp and 35 API
Hi=((jH)*(1/D)*(Z)); //Hi=(hi/phyp),using eq.6.15a,Btu/(hr)*(ftˆ2)*(F)
printf(”\t Hi is : %.0f Btu/(hr)*(ftˆ2)*(F) \n”,Hi);
ID=0.62; // ft
OD=0.75; //ft
Hio=((Hi)*(ID/OD)); //Hio=(hio/phyp), using eq.6.5
printf(”\t Correct Hi0 to the surface at the OD is : %.0f Btu/(hr)*(ftˆ2)*(F) \n”,Hio);
phyt=1;
hio=(Hio)*(phyt); // from eq.6.37
printf(”\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ftˆ2)*(F) \n”,hio);
printf(”\t cold fluid:shell side,rich oil \n”);
ID=31; // in
C=0.25; // clearance
B=12; // minimum baffle spacing,from eq 11.4,in
PT=1;
as=((ID*C*B)/(144*PT))/(2); // flow area,from eq 7.1,ftˆ2
printf(”\t flow area is : %.3f ftˆ2 \n”,as);
Gs=(w/as); // mass velocity,from eq 7.2,lb/(hr)*(ftˆ2)
printf(”\t mass velocity is : %.2e lb/(hr)*(ftˆ2) \n”,Gs);
mu1=3.15; // at 193.5F,lb/(ft)*(hr), from fig.14
De=0.95/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf(”\t reynolds number is : %.2e \n”,Res);
jH=45; // from fig.28
Z=0.213; // Z=(K*(c*mu3/k)ˆ(1/3)),Btu/(hr)(ftˆ2)(F/ft), at mu3=1.3cp and 35 API
Ho=((jH)*(1/De)*(Z)); // H0=(h0/phya),using eq.6.15,Btu/(hr)*(ftˆ2)*(F)
printf(”\t individual heat transfer coefficient is :
```

```
%.0f Btu/(hr)*(ft^2)*(F) \n",Ho);
phys=1;
ho=(Ho)*(phys); // from eq.6.36
printf("\t Correct h0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t pressure drop  for inner pipe \n");
f=0.00027; // friction factor for reynolds number 10100, using fig.26
s=0.77;
delPt=((2*f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.024; // X1=((V^2)/(2*g)), for Gt 1060000,using fig.27
delPr=((4*2*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPT is 10 psi \n");
printf("\t pressure drop  for annulus \n");
f=0.0023; // friction factor for reynolds number 6720, using fig.29
s=0.79; // for reynolds number 6720,using fig.6
Ds=31/12; // ft
De=0.0792;
N=(4*12*L/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t allowable delPa is 10 psi \n");
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
```

```
printf("\t The initial assumptions have provided an exchanger which very nearly meets all the requirements. Eight-pass units would meet the heat-transfer requirement but would give a tube-side pressure drop of 14 psi. The trial exchanger will be somewhat less suitable when the value of Q, is also taken into account. If the minimum dirt factor of 0.0040 is to be taken literally, it will be necessary to try the next size shell \n");
printf("\t Assume a 33 in. ID shell with six1 tube passes and baffies spaced 12-in. apart, since the pressure drop increases with the diameter of the shell for a given mass velocity. \n");
UC1=52.3;
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UC1);
UD2=42;
printf("\t correct design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD2);
Rd1=0.0047;
printf("\t calculated Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd1);
Rd2=0.004;
printf("\t actual Rd is : %.3f (hr)*(ft^2)*(F)/Btu \n",Rd2);
delPs1=4.4;
printf("\t delPs is : %.1f psi \n",delPs1);
delPT1=7.9;
printf("\t delPt is : %.1f psi \n",delPT1);
//end
```

Scilab code Exa 11.3 Calculation of a Caustic Solution Cooler

```
printf("\t example 11.3 \n");
```

```
printf(”\t approximate values are mentioned in the book \n”);
T1=190; // inlet hot fluid ,F
T2=120; // outlet hot fluid ,F
t1=80; // inlet cold fluid ,F
t2=120; // outlet cold fluid ,F
W=100000; // lb/hr
w=154000; // lb/hr
printf(”\t 1.for heat balance \n”);
printf(”\t for caustic \n”);
c=0.88; // Btu/(lb)*(F)
Q=((W)*(c)*(T1-T2)); // Btu/hr
printf(”\t total heat required for caustic is : %.2e Btu/hr \n”,Q);
printf(”\t for water \n”);
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf(”\t total heat required for water is : %.2e Btu/hr \n”,Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf(”\t delt1 is : %.0f F \n”,delt1);
printf(”\t delt2 is : %.0f F \n”,delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf(”\t LMTD is :%.1f F \n”,LMTD);
R=((T1-T2)/(t2-t1));
printf(”\t R is : %.2f \n”,R);
S=((t2-t1)/(T1-t1));
printf(”\t S is : %.3f \n”,S);
printf(”\t FT is 0.815 \n”); // for 4–8 exchanger , from fig 21
delt=(0.815*LMTD); // F
printf(”\t delt is : %.1f F \n”,delt);
Tc=((T2)+(T1))/(2); // caloric temperature of hot fluid ,F
printf(”\t caloric temperature of hot fluid is : %.0f F \n”,Tc);
tc=((t1)+(t2))/(2); // caloric temperature of cold
```

```
fluid ,F
printf("\t caloric temperature of cold fluid is : %.1f \n",tc);
UD1=250; // assume, from table 8
A1=((Q)/((UD1)*(delt)));
printf("\t A1 is : %.0f ft^2 \n",A1);
a1=0.2618; // ft^2/lin ft
L=16;
N1=(A1/(16*a1));
printf("\t number of tubes are : %.0f \n",N1);
N2=140; // assuming four tube passes,19.25in ID, from table 9
A2=(N2*L*a1); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A2);
UD=((Q)/((A2)*(delt)));
printf("\t correct design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
printf("\t hot fluid:shell side,caustic \n");
ID=19.25; // in
C=0.25; // clearance
B=7; // minimum baffle spacing,from eq 11.4,in
PT=1.25;
as=((ID*C*B)/(144*PT)); // flow area,from eq 7.1,ft^2
printf("\t flow area is : %.4f ft^2 \n",as);
Gs=(W/as); // mass velocity,from eq 7.2,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=1.84; // at 155F,lb/(ft)*(hr), from fig.14
De=0.72/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
jH=75; // from fig.28
Z=0.575; // Z=(K*(c*mu3/k)^(1/3)),Btu/(hr)(ft^2)(F/ft)
Ho=((jH)*(1/De)*(Z)); // H0=(h0/phya),using eq.6.15,
```

```
Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Ho);
phys=1; // low viscosity
ho=(Ho)*(phys); // from eq.6.36
printf("\t Correct h0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,water \n");
Nt=140;
n=4; // number of passes
L=16; //ft
at1=0.546; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
V=Gt/(3600*62.5);
printf("\t V is %.2f fps \n",V);
mu2=1.74; // at 100F,lb/(ft)*(hr)
D=0.0695; // ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=1240*0.94; // from fig 25
printf("\t Hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=0.834; // ft
OD=1; //ft
hio=((hi)*(ID/OD)); //Hio=(hio/phyp), using eq.6.5
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
printf("\t pressure drop  for annulus \n");
f=0.0019; // friction factor for reynolds number 17400, using fig.29
s=1.115; // for reynolds number 17400,using fig.6
Ds=19.25/12; // ft
De=0.06;
N=(12*L/B)+1; // number of crosses,using eq.7.43
```

```
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPs is : %.0f psi \n",delPs);
printf("\t allowable delPa is 10 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.00018; // friction factor for reynolds number 46300, using fig.26
s=1;
phyt=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.18; // X1=((V^2)/(2*g)), for Gt 1060000,using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPa is 10 psi \n");
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t Adjustment of the baffie space to use the full 10 psi will still not permit the exchanger to make the 0.002 dirt factor. The value of UD has been assumed too high \n");
printf("\t Try a 21.25in.ID shell with four tube passes and a 6in.baffie space.This corresponds to 170 tubes \n");
UC1=390;
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UC1);
UD2=200;
```

```
printf("\t correct design overall coefficient is : %
   .1f Btu/(hr)*(ft^2)*(F) \n",UD2);
Rd1=0.0024;
printf("\t calculated Rd is : %.4f (hr)*(ft^2)*(F)/
   Btu \n",Rd1);
Rd2=0.002;
printf("\t actual Rd is : %.3f (hr)*(ft^2)*(F)/Btu \
   n",Rd2);
delPs1=9.8;
printf("\t delPs is : %.1f psi \n",delPs1);
delPT1=4.9;
printf("\t delPt is : %.1f psi \n",delPT1);
//end
```

Scilab code Exa 11.4 `Calculation of an Alcohol Heater`

```
printf("\t example 11.4 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=225; // inlet hot fluid ,F
T2=225; // outlet hot fluid ,F
t1=80; // inlet cold fluid ,F
t2=200; // outlet cold fluid ,F
W=10350; // lb/hr
w=115000; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for steam \n");
l=962; // Btu/(lb)
Qh=((W)*(l)); // Btu/hr
printf("\t total heat required for steam is : %.2e
   Btu/hr \n",Qh);
printf("\t for alcohol \n");
c=0.72; // Btu/(lb)*(F)
Qc=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for alcohol is : %.2e
```

```
    Btu/hr \n",Qc);Q=(Qh+Qc)/(2);
Q=(Qh+Qc)/(2);
printf("\t Q is : %.2e V \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
Tc=((T2)+(T1)); // caloric temperature of hot fluid ,
   F
printf("\t caloric temperature of hot fluid is : %.0
   f F \n",Tc);
tc=((t1)+(t2)); // caloric temperature of cold fluid
   ,F
printf("\t caloric temperature of cold fluid is : %
   .1f \n",tc);
L=12;
UD1=200; // assume, from table 8
A1=((Q)/((UD1)*(LMTD)));
printf("\t A1 is : %.0f ft^2 \n",A1);
a1=0.2618; // ft^2/lin ft
N1=(A1/(12*a1));
printf("\t number of tubes are : %.0f \n",N1);
N2=232; // assuming two tube passes,23.25in ID, from
    table 9
A2=(N2*L*a1); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A2)
   ;
UD=((Q)/((A2)*(LMTD)));
printf("\t correct design overall coefficient is : %
   .0f Btu/(hr)*(ft^2)*(F) \n",UD);
printf("\t hot fluid:inner tube side,steam \n");
Nt=232;
n=2; // number of passes
L=12; //ft
at1=0.546; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq
```

```
.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(W/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu2=0.0314; // at 225F,lb/(ft)*(hr)
D=0.0695; // ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.1e \n",Ret);
hio=1500; // condensation of steam
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
printf("\t cold fluid:shell side,alcohol \n");
ID=23.25; // in
C=0.25; // clearance
B=7; // minimum baffle spacing,from eq 11.4,in
PT=1.25;
as=((ID*C*B)/(144*PT)); // flow area,from eq 7.1,ft^2
printf("\t flow area is : %.3f ft^2 \n",as);
Gs=(w/as); // mass velocity,from eq 7.2,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=1.45; // at 193.5F,lb/(ft)*(hr), from fig.14
De=0.72/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.1e \n",Res);
jH=83; // from fig.28
Z=0.195; // Z=(K*(c*mu3/k)^(1/3)),Btu/(hr)(ft^2)(F/ft)
Ho=((jH)*(1/De)*(Z)); // H0=(h0/phya),using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Ho);
phys=1;
ho=(Ho)*(phys); // from eq.6.36
printf("\t Correct h0 to the surface at the OD is :
```

```
%.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t pressure drop  for inner pipe \n");
f=0.000175; // friction factor for reynolds number 52000, using fig.26
s=0.00076;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(1)))/(2); // using eq.7.45,psi
printf("\t delPt is : %.2f psi \n",delPt);
printf("\t delPr is negligible \n");
printf("\t allowable delPa is negligible \n");
printf("\t pressure drop  for annulus \n");
f=0.0018; // friction factor for reynolds number 21000, using fig.29
s=0.78; // for reynolds number 21000,using fig.6
Ds=1.94; // ft
De=0.06;
N=(12*L/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t allowable delPa is 10 psi \n");
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.6f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t This is clearly an instance in which UD was assumed too high.It is now a question of how much too high. With the aid of the summary it is apparent thatin a larger shell a clean overall coefficient of about 200 may be expected \n");
printf("\t Assume a 27in. ID shell with 2 tube passes,334 tubes and baffies spaced 7in. apart, since the pressure drop increases with the diameter of the shell for a given mass velocity.
```

```
   \n");
UC1=214;
printf("\t clean overall coefficient is : %.1f Btu/(
   hr)*(ft^2)*(F) \n",UC1);
UD2=138.5;
printf("\t correct design overall coefficient is : %
   .1f Btu/(hr)*(ft^2)*(F) \n",UD2);
Rd1=0.0025;
printf("\t calculated Rd is : %.4f (hr)*(ft^2)*(F)/
   Btu \n",Rd1);
Rd2=0.002;
printf("\t actual Rd is : %.3f (hr)*(ft^2)*(F)/Btu \
   n",Rd2);
delPs1=0.23;
printf("\t delPs is : %.2f psi \n",delPs1);
delPT1=7.1;
printf("\t delPt is : %.1f psi \n",delPT1);
//end
```

Scilab code Exa 11.5 `Calculation of a Flue Gas Cooler`

```
printf("\t example 11.5 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=250; // inlet hot fluid ,F
T2=125; // outlet hot fluid ,F
t1=80; // inlet cold fluid ,F
t2=100; // outlet cold fluid ,F
W=41300; // lb/hr
w=64500; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for gas \n");
c=0.25; // Btu/(lb)*(F)
Q=((W)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for gas is : %.2e Btu
```

```
    /hr \n”,Q);
printf(”\t for water \n”);
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf(”\t total heat required for water is : %.2e
    Btu/hr \n”,Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf(”\t delt1 is : %.0f F \n”,delt1);
printf(”\t delt2 is : %.0f F \n”,delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf(”\t LMTD is :%.1f F \n”,LMTD);
R=((T1-T2)/(t2-t1));
printf(”\t R is : %.2f \n”,R);
S=((t2-t1)/(T1-t1));
printf(”\t S is : %.3f \n”,S);
printf(”\t FT is 0.935 \n”); // from fig 18
delt=(0.935*LMTD); // F
printf(”\t delt is : %.1f F \n”,delt);
Tc=((T2)+(T1))/(2); // caloric temperature of hot
    fluid ,F
printf(”\t caloric temperature of hot fluid is : %.0
    f F \n”,Tc);
tc=((t1)+(t2))/(2); // caloric temperature of cold
    fluid ,F
printf(”\t caloric temperature of cold fluid is : %
    .0f F \n”,tc);
UD1=15; // assume, from table 8
A1=((Q)/((UD1)*(delt)));
printf(”\t A1 is : %.0f ft^2 \n”,A1);
a1=0.2618; // ft^2/lin ft
N1=(A1/(12*a1));
printf(”\t number of tubes are : %.0f \n”,N1);
N2=358; // assuming 12 tube passes, from table 9
L=12;
A2=(N2*L*a1); // ft^2
printf(”\t total surface area is : %.0f ft^2 \n”,A2)
    ;
```

```
UD=((Q)/((A2)*(delt)));
printf("\t correct design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
printf("\t When solved in a manner identical with the preceding examples and using the smallest integral number of bundle crosses (five) corresponding to a 28.8in.spacing \n");
UC1=22.7;
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UC1);
UD2=14;
printf("\t correct design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD2);
Rd1=0.027;
printf("\t calculated Rd is : %.3f (hr)*(ft^2)*(F)/Btu \n",Rd1);
Rd1=0.005;
printf("\t required Rd is : %.3f (hr)*(ft^2)*(F)/Btu \n",Rd1);
delPs1=5.2;
printf("\t delPs is : %.1f psi \n",delPs1);
delPt1=1.0;
printf("\t delPt is : %.1f psi \n",delPt1);
printf("\t The first trial is disqualified because of failure to meet the required dirt factor and the the pressure drop is five times greater than the allowable \n");
printf("\t This would be unsatisfactory, since gases require large inlet connections and the flow distribution on the first and third bundle crosses would be poor and the conditions of allowable pressure drop would still not be met \n");
UD1=15; // assume, from table 8
A1=((Q)/((UD1)*(delt)));
printf("\t A1 is : %.0f ft^2 \n",A1);
a1=0.2618; // ft^2/lin ft
N1=(A1/(12*a1));
```

```
printf("\t number of tubes are : %.0f \n",N1);
N2=340; // assuming eight tube passes , from table 9
A2=(N2*L*a1); // ft^2
printf("\t total surface area is : %.2e ft^2 \n",A2);
UD=((Q)/((A2)*(delt)));
printf("\t correct design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
printf("\t hot fluid:shell side,gas \n");
ID=31; // in
C=0.25; // clearance
B=24; // baffle spacing,in
PT=1.25;
as=((ID*C*B)/(144*PT)); // flow area,from eq 7.1,ft^2
printf("\t flow area is : %.2f ft^2 \n",as);
Gs=(W/as)/(2); // mass velocity,from eq 7.2,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.0e lb/(hr)*(ft^2) \n",Gs);
mu1=0.050; // at 187.5F,lb/(ft)*(hr), from fig.15
De=0.99/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.1e \n",Res);
jH=105; // from fig.28
k=0.015; // Btu/(hr)(ft^2)( F /ft)
Z=0.94; // Z=((c*mu3/k)^(1/3)),Btu/(hr)(ft^2)(F/ft)
Ho=((jH)*(k/De)*(Z)); // H0=(h0/phya),using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Ho);
phys=1;
ho=(Ho)*(phys); // from eq.6.36
printf("\t Correct h0 to the surface at the OD is : %.1f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,crude oil \n");
Nt=340;
```

```
n=12; // number of passes
L=12; //ft
at1=0.546; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*62.5));
printf("\t V is : %.2f fps \n",V);
mu2=1.96; // at 90F,lb/(ft)*(hr)
D=0.0695; // ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=667; //Btu/(hr)*(ft^2)*(F)
printf("\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=0.83; // ft
OD=1; //ft
hio=((hi)*(ID/OD)); //Hio=(hio/phyp), using eq.6.5
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio); // calculation mistake
phyt=1;
printf("\t pressure drop for annulus \n");
f=0.0017; // friction factor for reynolds number 33000, using fig.29
s=0.0012; // for reynolds number 33000,using fig.6
Ds=31/12; // ft
N=(3); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t pressure drop for inner pipe \n");
f=0.00022; // friction factor for reynolds number 21300, using fig.26
s=1;
```

```
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.052; // X1=((V^2)/(2*g)), for Gt 1060000,using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficent ,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
// end
```

Chapter 12

Condensation of single vapor

Scilab code Exa 12.1 `Calculation of a Horizontal n Propanol Condenser`

```
printf(”\t example 12.1 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=244; // inlet hot fluid ,F
T2=244; // outlet hot fluid ,F
t1=85; // inlet cold fluid ,F
t2=120; // outlet cold fluid ,F
W=60000; // lb/hr
w=488000; // lb/hr
printf(”\t 1.for heat balance \n”);
printf(”\t for propanol \n”);
l=285; // Btu/(lb)
Q=((W)*(l)); // Btu/hr
printf(”\t total heat required for propanol is : %.2e Btu/hr \n”,Q);
printf(”\t for water \n”);
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf(”\t total heat required for water is : %.2e Btu/hr \n”,Q);
delt1=T2-t1; //F
```

```
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
Tc=((T2)+(T1))/(2); // caloric temperature of hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+(t2))/(2); // caloric temperature of cold fluid ,F
printf("\t caloric temperature of cold fluid is : %.1f F \n",tc);
UD1=100; // assume, from table 8
A1=((Q)/((UD1)*(LMTD)));
printf("\t A1 is : %.0f ft^2 \n",A1);
a1=0.1963; // ft^2/lin ft
N1=(A1/(8*a1));
printf("\t number of tubes are : %.0f \n",N1);
N2=766; // assuming 4 tube passes, from table 9
A2=(N2*8*a1); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A2);
UD=((Q)/((A2)*(LMTD)));
printf("\t correct design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
printf("\t hot fluid:shell side,propanol \n");
ID=31; // in
C=0.1875; // clearance
B=31; // baffle spacing,in
PT=0.937;
L=8; // ft
as=((ID*C*B)/(144*PT)); // flow area,from eq 7.1,ft^2
printf("\t flow area is : %.2f ft^2 \n",as);
Gs=(W/as); // mass velocity,from eq 7.2,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n
```

```
",Gs);
G1=(W/(L*N2^(2/3))); // from eq.12.43
printf("\t G1 is : %.1f lb/(hr)*(lin ft) \n",G1);
printf("\t cold fluid:inner tube side,water \n");
Nt=766;
n=4; // number of passes
L=8; //ft
at1=0.302; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*62.5));
printf("\t V is : %.2f fps \n",V);
mu2=1.74; // at 102.5F,lb/(ft)*(hr)
D=0.0517; // ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=1300; //Btu/(hr)*(ft^2)*(F)
printf("\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=0.62; // ft
OD=0.75; //ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio); // calculation mistake
ho=200; // assumption
tw=(tc)+(((ho)/(hio+ho))*(Tc-tc)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
tf=(Tc+tw)/(2); // from eq 12.19
printf("\t tf is : %.1f F \n",tf);
kf=0.094; // Btu/(hr)*(ft^2)*(F/ft), from table 4
sf=0.8; // from table 6
muf=0.62; // cp, from fig 14
ho=172; // Btu/(hr)*(ft^2)*(F), from fig 12.9
printf("\t Correct ho to the surface at the OD is :
```

```
%.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t Based on h=172 instead of the assumed 200 a new value of tw,and tf could be obtained to give a more exact value of h based on fluid properties at a value of tf more nearly correct \n");
printf("\t pressure drop  for annulus \n");
mu1=0.0242; // lb/(ft)*(hr), fir 15
De=0.0458; // fig 28
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
f=0.00141; // friction factor for reynolds number 84600, using fig.29
s=0.00381; // for reynolds number 84600,using fig.6
Ds=31/12; // ft
phys=1;
N=(3); // number of crosses ,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys)))/(2); // using eq.12.47,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t allowable delPa is 2 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.00019; // friction factor for reynolds number 36200, using fig.26
s=1;
phyt=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.2; // X1=((V^2)/(2*g)),using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPT is 10 psi \n");
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
```

```
printf("\t clean overall coefficient is : %.1f Btu/(
   hr)*(ft^2)*(F) \n",Uc);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \
   n",Rd);
// end
```

Scilab code Exa 12.2 `Design of a Vertical n Propanol Condenser`

```
printf("\t example 12.2 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=244; // inlet hot fluid ,F
T2=244; // outlet hot fluid ,F
t1=85; // inlet cold fluid ,F
t2=120; // outlet cold fluid ,F
W=60000; // lb/hr
w=488000; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for propanol \n");
l=285; // Btu/(lb)
Q=((W)*(l)); // Btu/hr
printf("\t total heat required for propanol is : %.2
   e Btu/hr \n",Q);
printf("\t for water \n");
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for water is : %.2e
   Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
```

```
Tc=((T2)+(T1))/(2); // caloric temperature of hot fluid ,F
printf(”\t caloric temperature of hot fluid is : %.0f F \n”,Tc);
tc=((t1)+(t2))/(2); // caloric temperature of cold fluid ,F
printf(”\t caloric temperature of cold fluid is : %.1f F \n”,tc);
UD1=70; // assume, from table 8
A1=((Q)/((UD1)*(LMTD)));
printf(”\t A1 is : %.2e ft^2 \n”,A1);
N2=766; // assuming 4 tube passes, from table 9
a1=0.1963; // ft^2/lin ft
L=(A1/(N2*a1));
printf(”\t L is : %.1f ft \n”,L);
A2=(N2*12*a1); // ft^2
printf(”\t total surface area is : %.0f ft^2 \n”,A2);
UD=((Q)/((A2)*(LMTD)));
printf(”\t correct design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n”,UD);
printf(”\t hot fluid:shell side,propanol \n”);
Do=0.0625; // ft
G1=(W/(3.14*N2*Do)); // from eq.12.36
printf(”\t G1 is : %.0f lb/(hr)*(lin ft) \n”,G1);
printf(”\t cold fluid:inner tube side,water \n”);
Nt=766;
n=4; // number of passes
L=12; //ft
at1=0.302; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf(”\t flow area is : %.3f ft^2 \n”,at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf(”\t mass velocity is : %.2e lb/(hr)*(ft^2) \n”,Gt);
V=(Gt/(3600*62.5));
printf(”\t V is : %.2f fps \n”,V);
```

```
mu2=1.74; // at 102.5F,lb/(ft)*(hr)
D=0.0517; // ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=1300; //Btu/(hr)*(ft^2)*(F)
printf("\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=0.62; // ft
OD=0.75; //ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
ho=100; // assumption
tw=(tc)+(((ho)/(hio+ho))*(Tc-tc)); // from eq.5.31
printf("\t tw is : %.1f F \n",tw);
tf=(Tc+tw)/(2); // from eq 12.19
printf("\t tf is : %.0f F \n",tf);
kf=0.0945; // Btu/(hr)*(ft^2)*(F/ft), from table 4
sf=0.76; // from table 6
muf=0.65; // cp, from fig 14
ho=102; // Btu/(hr)*(ft^2)*(F), from fig 12.9
printf("\t Correct ho to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t pressure drop  for annulus \n");
ID=31; // in
C=0.1875; // clearance
B=29; // baffle spacing,in
PT=0.937;
as=((ID*C*B)/(144*PT)); // flow area,from eq 7.1,ft^2
printf("\t flow area is : %.2f ft^2 \n",as);
Gs=(W/as); // mass velocity,from eq 7.2,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=0.0242; // lb/(ft)*(hr), fig 15
De=0.0458; // fig 28
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.1e \n",Res);
```

```
f=0.0014; // friction factor for reynolds number 91000, using fig.29
s=0.00381; // for reynolds number 91000,using fig.6
Ds=31/12; // ft
phys=1;
N=(5); // number of crosses ,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys)))/(2); // using eq.12.47 , psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t allowable delPa is 2 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.00019; // friction factor for reynolds number 36200, using fig.26
s=1;
phyt=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45 , psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.2; // X1=((V^2)/(2*g)) ,using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46 , psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47 , psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPT is 10 psi \n");
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient ,eq 6.38 ,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=((Uc-UD)/((UD)*(Uc))); // eq 6.13 ,(hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.5f (hr)*(ft^2)*(F)/Btu \n",Rd);
// end
```

Scilab code Exa 12.3 Calculation of a Butane Desuperheater condenser

```
printf(”\t example 12.3 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=200; // inlet hot fluid ,F
T2=130; // outlet hot fluid ,F
T3=125; // after condensation
t1=65; // inlet cold fluid ,F
t3=100; // outlet cold fluid ,F
W=27958; // lb/hr
w=135500; // lb/hr
printf(”\t 1.for heat balance \n”);
printf(”\t for butane \n”);
c=0.44; // Btu/(lb)(F)
qd=((W)*(c)*(T1-T2)); // Btu/hr
printf(”\t total heat required for desuperheating of butane is : %.1e Btu/hr \n”,qd);
HT2=309; // enthalpy at T2, Btu/lb
HT3=170; // enthalpy at T3, Btu/lb
qc=(W*(HT2-HT3)); // for condensation
printf(”\t total heat required for condensing of butane is : %.2e Btu/hr \n”,qc);
Q=qd+qc;
printf(”\t total heat required for  butane is : %.2e Btu/hr \n”,Q);
printf(”\t for water \n”);
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t3-t1)); // Btu/hr
printf(”\t total heat required for water is : %.2e Btu/hr \n”,Q);
deltw=(qc/w);
printf(”\t deltw is : %.1f F \n”,deltw);
t2=t1+deltw;
printf(”\t t2 is : %.1f F \n”,t2)
printf(”\t for desuperheating \n”);
delt1=T2-t2; //F
delt2=T1-t3; // F
```

```
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTDd=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTDd);
w1=(qd/LMTDd);
printf("\t w1 is : %.3e lb/hr \n",w1);
printf("\t for condensing \n");
delt3=T3-t1; //F
delt4=T2-t2; // F
printf("\t delt1 is : %.0f F \n",delt3);
printf("\t delt2 is : %.0f F \n",delt4);
LMTDc=((delt4-delt3)/((2.3)*(log10(delt4/delt3))));
printf("\t LMTD is :%.0f F \n",LMTDc);
w2=(qc/LMTDc);
printf("\t w1 is : %.2e lb/hr \n",w2);
delt=(Q/(w1+w2));
printf("\t delt is : % .1f F \n",delt);
Tc=((T3)+(T2))/(2); // caloric temperature of hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.1f F \n",Tc);
tc=((t1)+(t3))/(2); // caloric temperature of cold fluid ,F
printf("\t caloric temperature of cold fluid is : %.1f F \n",tc);
printf("\t hot fluid:shell side,butane \n");
ID=23.25; // in
C=0.25; // clearance
B=12; // baffle spacing,in
PT=1;
as=((ID*C*B)/(144*PT)); // flow area,ft^2
printf("\t flow area is : %.3f ft^2 \n",as);
printf("\t desuperheating \n");
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=0.0242; // at 165F,lb/(ft)*(hr), from fig.15
De=0.73/12; // from fig.28,ft
```

```
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
jH=239; // from fig.28
k=0.012; // Btu/(hr)*(ft^2)*(F/ft), from table 5
Z=0.96; // Z=((c)*(mu1)/k)^(1/3)
ho=((jH)*(k/De)*(Z)); // H0=(h0/phya),using eq.6.15b,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",ho);
printf("\t cold fluid:inner tube side,water \n");
Nt=352;
n=4; // number of passes
L=16; //ft
at1=0.302; // flow area,table 10, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*62.5));
printf("\t V is : %.2f fps \n",V);
mu2=2.11; // at 82.5F, fig 14,lb/(ft)*(hr)
D=0.0517; // ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=800; // fig 25,Btu/(hr)*(ft^2)*(F)
printf("\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=0.62; // ft
OD=0.75; //ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hio to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
Ud=((hio)*(ho)/(hio+ho)); // clean overall coefficient ,eq 6.38,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Ud);
Ad=(qd/(Ud*LMTDd));
```

```
printf("\t clean surface required for desuperheating : %.0f ft^2 \n",Ad);
printf("\t for condensaton \n");
Lc=16*0.6; // condensation occurs 60% of the tube length
G1=(W/(Lc*Nt^(2/3))); // from eq.12.43
printf("\t G1 is : %.1f lb/(hr)*(lin ft) \n",G1);
ho=200; // assumption
tw=(tc)+(((ho)/(hio+ho))*(Tc-tc)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
tf=(Tc+tw)/(2); // from eq 12.19
printf("\t tf is : %.0f F \n",tf);
kf=0.075; // Btu/(hr)*(ft^2)*(F/ft)
sf=0.55; // from table 6
muf=0.14; // cp, from fig 14
ho=207; // Btu/(hr)*(ft^2)*(F), from fig 12.9
printf("\t Correct ho to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
Ac=(qc/(Uc*LMTDc));
printf("\t clean surface required for desuperheating : %.0f ft^2 \n",Ac);
AC=Ad+Ac;
printf("\t total clean surface : %.0f ft^2 \n",AC);
lc=(Ac/(Ac+Ad));
printf("\t assumed condensing length percentage : %.2f \n",lc);
UC=((Ud*Ad)+(Uc*Ac))/(AC);
printf("\t weighted clean overall coefficient : %.0f Btu/(hr)*(ft^2)*(F) \n",UC);
A2=0.1963; // actual surface supplied for each tube, ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(delt)));
```

```
printf("\t actual design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((UC-UD)/((UD)*(UC))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
printf("\t desuperheating \n");
Ld=6.4; //ft
De=0.0608; // fig 28
f=0.0013; // friction factor for reynolds number 145000, using fig.29
Ds=1.94; // ft
phys=1;
N=(12*Ld/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
row=(58.1/((359)*(625/492)*(14.7/99.7)));
printf("\t row is %.3f lb/ft^3 \n",row);
s=(row/62.5);
printf("\t s is %.4f \n",s);
delPsd=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPs is : %.1f psi \n",delPsd);
printf("\t condensation \n");
N=(12*Lc/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPsc=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys)))/(2); // using eq 12.47,psi
printf("\t delPsc is : %.1f psi \n",delPsc);
delPS=delPsd+delPsc;
printf("\t delPS is : %.0f psi \n",delPS);
printf("\t allowable delPa is 2 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.00023; // friction factor for reynolds number 17900, using fig.26
s=1;
phyt=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
```

```
printf("\t delPt is : %.0f psi \n",delPt);
X1=0.075; // X1=((V^2)/(2*g)),using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPa is 10 psi \n");
//end
```

Scilab code Exa 12.4 `Calculation of a Vertical Condenser subeooler`

```
printf("\t example 12.4 \n");
printf("\t approximate values are mentioned in the book \n");
T1=130; // inlet hot fluid,F
T2=125; // outlet hot fluid,F
T3=100; // after sucooling
t1=80; // inlet cold fluid,F
t3=100; // outlet cold fluid,F
W=21000; // lb/hr
w=167000; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for pentane \n");
HT1=315; // enthalpy at T1, Btu/lb
HT2=170; // enthalpy at T2, Btu/lb
qc=(W*(HT1-HT2)); // for condensation
printf("\t total heat required for condensing of pentane is : %.2e Btu/hr \n",qc);
c=0.57; // Btu/(lb)(F)
qs=((W)*(c)*(T2-T3)); // Btu/hr
printf("\t total heat required for subcooling of pentane is : %.0e Btu/hr \n",qs);
Q=qs+qc;
printf("\t total heat required for  pentane is : %.2e Btu/hr \n",Q);
```

```
printf("\t for water \n");
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t3-t1)); // Btu/hr
printf("\t total heat required for water is : %.2e Btu/hr \n",Q);
deltw=(qc/w);
printf("\t deltw is : %.1f F \n",deltw);
t2=t3-deltw;
printf("\t t2 is : %.1f F \n",t2)
printf("\t for condensing \n");
delt1=T2-t2; //F
delt2=T1-t3; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTDc=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTDc);
w1=(qc/LMTDc);
printf("\t w1 is : %.2e lb/hr \n",w1);
printf("\t subcooling \n");
delt3=T3-t1; //F
delt4=T2-t2; // F
printf("\t delt1 is : %.0f F \n",delt3);
printf("\t delt2 is : %.0f F \n",delt4);
LMTDs=((delt4-delt3)/((2.3)*(log10(delt4/delt3))));
printf("\t LMTD is :%.1f F \n",LMTDs);
w2=(qs/LMTDs);
printf("\t w1 is : %.2e lb/hr \n",w2);
delt=(Q/(w1+w2));
printf("\t delt is : % .1f F \n",delt);
Tc=((T1)+(T2))/(2); // caloric temperature of hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.1f F \n",Tc);
tc=((t1)+(t3))/(2); // caloric temperature of cold fluid ,F
printf("\t caloric temperature of cold fluid is : %.0f F \n",tc);
printf("\t hot fluid:shell side,pentane \n");
```

```
printf(”\t for condensaton \n”);
Do=0.0625; // ft
Nt=370; // number of tubes
G1=(W/(3.14*Nt*Do)); // from eq.12.42
printf(”\t G1 is : %.1e lb/(hr)*(lin ft) \n”,G1);
printf(”\t cold fluid:inner tube side,water \n”);
n=4; // number of passes
L=16; //ft
at1=0.302; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq
   .7.48
printf(”\t flow area is : %.3f ft^2 \n”,at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf(”\t mass velocity is : %.1e lb/(hr)*(ft^2) \n
   ”,Gt);
V=(Gt/(3600*62.5));
printf(”\t V is : %.2f fps \n”,V);
mu2=1.98; // at 90F,lb/(ft)*(hr)
D=0.0517; // ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf(”\t reynolds number is : %.2e \n”,Ret);
hi=940; //Btu/(hr)*(ft^2)*(F)
printf(”\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n”,hi);
ID=0.62; // ft
OD=0.75; //ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf(”\t Correct hio to the surface at the OD is :
    %.0f Btu/(hr)*(ft^2)*(F) \n”,hio);
ho=125; // assumption
tw=(tc)+(((ho)/(hio+ho))*(Tc-tc)); // from eq.5.31
printf(”\t tw is : %.0f F \n”,tw);
tf=(Tc+tw)/(2); // from eq 12.19
printf(”\t tf is : %.0f F \n”,tf);
kf=0.077; // Btu/(hr)*(ft^2)*(F/ft), table 4
sf=0.6; // from table 6
muf=0.19; // cp, from fig 14
ho=120; // Btu/(hr)*(ft^2)*(F), from fig 12.9
printf(”\t Correct ho to the surface at the OD is :
```

```
%.0f Btu/(hr)*(ft^2)*(F) \n",ho);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
Ac=(3040000/(104*36.4));
printf("\t clean surface required for dcondensation : %.0f ft^2 \n",Ac);
printf("\t subcooling \n");
ID=25; // in
C=0.25; // clearance
B=12; // baffle spacing,in
PT=1;
as=((ID*C*B)/(144*PT)); // flow area,ft^2
printf("\t flow area is : %.3f ft^2 \n",as);
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu1=0.46; // at 112.5F,lb/(ft)*(hr), from fig.14
De=0.95/12; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
jH=46.5; // from fig.28
k=0.077; // Btu/(hr)*(ft^2)*(F/ft), from table 4
Z=1.51; // Z=((c)*(mu1)/k)^(1/3)
ho=((jH)*(k/De)*(Z)); // using eq.6.15b,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
Us=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",Us);
As=(qs/(Us*LMTDs));
printf("\t clean surface required for desuperheating : %.1f ft^2 \n",As);
AC=As+Ac;
printf("\t total clean surface : %.0f ft^2 \n",AC);
```

```
UC=((Us*As)+(Uc*Ac))/(AC);
printf("\t weighted clean overall coefficient : %.1f Btu/(hr)*(ft^2)*(F) \n",UC);
A2=0.1963; // actual surface supplied for each tube, ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((UC-UD)/((UD)*(UC))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
printf("\t condensation \n");
Lc=13.4; //ft
De=0.0792; // fig 28
f=0.0012; // friction factor for reynolds number 193000, using fig.29
mu3=0.0165; // at 127.5F
Ds=2.08; // ft
phys=1;
Res1=(De*Gs/mu3);
printf("\t reynolds number is %.2e \n",Res1);
rowvap=(72.2/((359)*(590/492)*(14.7/25)));
printf("\t rowvapour is %.3f ld/ft^3 \n",rowvap);
s=(rowvap/62.5);
printf("\t s is %.5f \n",s);
N=(12*Lc/B)+(1); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPsc=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys)))/(2); // using eq.12.47,psi
printf("\t delPsc is : %.1f psi \n",delPsc);
printf("\t delPss is negligible \n");
printf("\t allowable delPa is 2 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.00022; // friction factor for reynolds number 22500, using fig.26
```

```
s=1;
phyt=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(
    phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.1; // X1=((V^2)/(2*g)),using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPT is 10 psi \n");
//end
```

Scilab code Exa 12.5 Calculation of a Horizontal Condenser subcooler

```
printf("\t example 12.5 \n");
printf("\t approximate values are mentioned in the
    book \n");
T1=130; // inlet hot fluid,F
T2=125; // outlet hot fluid,F
T3=100; // after subcooling
t1=80; // inlet cold fluid,F
t3=100; // outlet cold fluid,F
W=21000; // lb/hr
w=167000; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for pentane \n");
c=0.57; // Btu/(lb)(F)
qs=((W)*(c)*(T2-T3)); // Btu/hr
printf("\t total heat required for subcooling of
    pentane is : %.0e Btu/hr \n",qs);
HT1=315; // enthalpy at T1, Btu/lb
HT2=170; // enthalpy at T2, Btu/lb
qc=(W*(HT1-HT2)); // for condensation
printf("\t total heat required for condensing of
```

```
    pentane is : %.2e Btu/hr \n",qc);
Q=qs+qc;
printf("\t total heat required for  pentane is : %.2
    e Btu/hr \n",Q);
printf("\t for water \n");
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t3-t1)); // Btu/hr
printf("\t total heat required for water is : %.2e
    Btu/hr \n",Q);
deltw=18.2;
printf("\t deltw is : %.1f F \n",deltw);
t2=t3-deltw;
printf("\t t2 is : %.1f F \n",t2)
printf("\t for condensing \n");
delt1=T2-t2; //F
delt2=T1-t3; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTDc=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTDc);
w1=(qc/LMTDc);
printf("\t w1 is : %.2e lb/hr \n",w1);
printf("\t subcooling \n");
delt3=T3-t1; //F
delt4=T2-t2; // F
printf("\t delt1 is : %.0f F \n",delt3);
printf("\t delt2 is : %.0f F \n",delt4);
LMTDs=((delt4-delt3)/((2.3)*(log10(delt4/delt3))));
printf("\t LMTD is :%.1f F \n",LMTDs);
w2=(qs/LMTDs);
printf("\t w1 is : %.2e lb/hr \n",w2);
delt=(Q/(w1+w2));
printf("\t delt is : % .1f F \n",delt);
Tc=((T1)+(T2))/(2); // caloric temperature of hot
    fluid ,F
printf("\t caloric temperature of hot fluid is : %.1
    f F \n",Tc);
tc=((t1)+(t3))/(2); // caloric temperature of cold
```

```
fluid,F
printf("\t caloric temperature of cold fluid is : %.0f F \n",tc);
printf("\t hot fluid:shell side,pentane \n");
C1=0.198; // for 0.3Ds
Ds=25; // in
L=16; // ft
N=370
a=(C1*Ds^2);
printf("\t a is : %.0f in^2 \n",a);
N1=((N*a*4)/(3.14*Ds^2));
printf("\t number of submerged tubes are : %.0f \n",N1);
Nt=N-N1;
printf("\t number of tubes for condensation are : %.0f \n",Nt);
Af=(N1/N);
printf("\t flooded surface : %.2f \n",Af);
printf("\t for condensaton \n");
G1=(W/(L*Nt^(2/3))); // from eq.12.43
printf("\t G1 is : %.1f lb/(hr)*(lin ft) \n",G1);
printf("\t cold fluid:inner tube side,water \n");
n=4; // number of passes
L=16; //ft
at1=0.302; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.1e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*62.5));
printf("\t V is : %.2f fps \n",V);
mu2=1.98; // lb/(ft)*(hr)
D=0.0517; // ft
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=940; //Btu/(hr)*(ft^2)*(F)
```

```
printf("\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=0.62; // ft
OD=0.75; //ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hio to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
ho=251; // Btu/(hr)*(ft^2)*(F), from fig 12.9
printf("\t Correct ho to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
Ac=(qc/(Uc*LMTDc));
printf("\t clean surface required for dcondensation : %.0f ft^2 \n",Ac);
printf("\t subcooling \n");
ho=50; // Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
Us=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Us);
As=(qs/(Us*LMTDs));
printf("\t clean surface required for desuperheating : %.0f ft^2 \n",As);
AC=As+Ac;
printf("\t total clean surface : %.0f ft^2 \n",AC);
UC=((Us*As)+(Uc*Ac))/(AC);
printf("\t weighted clean overall coefficient : %.0f Btu/(hr)*(ft^2)*(F) \n",UC);
A=1160; // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((UC-UD)/((UD)*(UC))); // (hr)*(ft^2)*(F)/Btu
```

```
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
printf("\t condensation \n");
printf("\t It will be necessary to spread the batHes to a spacing of 18in.to compensate for the reduction in crossfiow area due to the flooded subcooling zone. The tube-side pressure drop will be the same as before. Assume bundle flooded to 0.3Ds.\n");
As=0.547; // ft^2
Gs=(W/(As)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.1e lb/(hr)*(ft^2) \n",Gs);
De=0.0792; // fig 28
Res=((De)*(Gs)/0.0165); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
f=0.00121; // friction factor for reynolds number 193000, using fig.29
s=0.00454; // for reynolds number 193000,using fig.6
Ds=2.08; // ft
B=18
phys=1;
N=(12*L/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
delPsc=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys)))/(2); // using eq.12.47,psi
printf("\t delPsc is : %.1f psi \n",delPsc);
printf("\t delPss is negligible \n");
printf("\t allowable delPa is 2 psi \n");
//end
```

Scilab code Exa 12.6 Calculation of Vertical Reflux type CSo Condenser

```
printf("\t example 12.6 \n");
```

```
printf("\t approximate values are mentioned in the book \n");
T1=176; // inlet hot fluid ,F
T2=176; // outlet hot fluid ,F
t1=85; // inlet cold fluid ,F
t2=120; // outlet cold fluid ,F
W=30000; // lb/hr
w=120000; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for carbon disulfide \n");
l=140; // Btu/(lb)
Q=((W)*l); // Btu/hr
printf("\t total heat required for carbon disulfide is : %.1e Btu/hr \n",Q);
printf("\t for water \n");
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for water is : %.0f Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
Tc=((T2)+T1)/2; // caloric temperature of hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+(t2))/2; // caloric temperature of cold fluid ,F
printf("\t caloric temperature of cold fluid is : %.1f F \n",tc);
printf("\t hot fluid:inner tube side,carbon disulfide \n");
hio=300; // Btu/(hr)*(ft^2)*(F)
printf("\t cold fluid:shell side,water \n");
ID=17.25; // in
```

```
C=0.25; // clearance
B=6; // baffle spacing,in
PT=1;
as=((ID*C*B)/(144*PT)); // flow area,ft^2
printf(”\t flow area is : %.2f ft^2 \n”,as);
Gs=(w/as); // mass velocity,lb/(hr)*(ft^2)
printf(”\t mass velocity is : %.2e lb/(hr)*(ft^2) \n”,Gs);
mu1=1.7; // at 280F,lb/(ft)*(hr), from fig.14
De=0.0792; // from fig.28,ft
Res=((De)*(Gs)/mu1); // reynolds number
printf(”\t reynolds number is : %.1e \n”,Res);
jH=103; // from fig.28
k=0.36; // Btu/(hr)*(ft^2)*(F/ft), from fig.1
Z=1.68; // Z=((c)*(mu1)/k)^(1/3); // prandelt number
ho=((jH)*(k/De)*(Z)); // using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf(”\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n”,ho);
tw=(tc)+(((hio)/(hio+ho))*(Tc-tc)); // from eq.5.31
printf(”\t tw is : %.1f F \n”,tw);
tf=(Tc+tw)/(2); // from eq 12.19
printf(”\t tf is : %.1f F \n”,tf);
printf(”\t hot fluid:inner tube side,carbon disulfide \n”);
kf=0.09; // Btu/(hr)*(ft^2)*(F/ft), from fig 14
sf=1.26; // from table 6
rowf=78.8; // lb/ft^3
muf=0.68; // cp, from fig 24
Nt=177;
D=0.0517; // ft
G1=(W/(3.14*Nt*D));
printf(”\t G1 is : %.f lb/(hr)*(lin ft) \n”,G1);
Ret=((4)*(G1)/muf); // reynolds number
printf(”\t reynolds number is : %.0f \n”,Ret);
hi=(0.251*(((kf^3)*(rowf^2)*(4.17*10^8))/(muf^2))^(1/3)); // hi*(((kf^3)*(rowf^2)*(4.17*10^8))/(muf^2))^(-1)=0.251, from fig 12.12
```

```
printf("\t hi is : %.0e Btu/(hr)*(ft^2)*(F) \n",hi);
ID=0.62; // ft
OD=.75; //ft
hio1=((hi)*(ID/OD)); //Hio=(hio/phyp), using eq.6.5
printf("\t Correct hio1 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio1);
Uc=((hio1)*(ho)/(hio1+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
A2=0.1963; // actual surface supplied for each tube,ft^2,from table 10
L=16;
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(LMTD)));
printf("\t actual design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.5f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for inner pipe \n");
n=1; // number of passes
at1=0.302; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(30000/(0.372)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu2=0.029; // at inlet,lb/(ft)*(hr)
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
row=(76.1/((359)*(636/492)*(14.7/39.7)));
printf("\t row is %.3f ld/ft^3 \n",row);
s=(row/62.5);
printf("\t s is %.4f \n",s);
f=0.000138; // friction factor for reynolds number
```

```
143000, using fig.26
delPt=((f*(Gt^2)*(16)*(1))/(5.22*(10^10)*(0.0517)*(s)))/(2); // using eq.7.45,psi
printf(”\t delPt is : %.1f psi \n”,delPt);
printf(”\t allowable delPa is negligible psi \n”);
printf(”\t pressure drop  for annulus \n”);
f=0.0017; // friction factor for reynolds number 31000, using fig.29
s=1; // for reynolds number 31000,using fig.6
Ds=17.25/12; // ft
B=6;
N=(12*L/B); // number of crosses,using eq.7.43
printf(”\t number of crosses are : %.0f \n”,N);
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s))); // using eq.7.44,psi
printf(”\t delPs is : %.1f psi \n”,delPs);
printf(”\t allowable delPT is 10 psi \n”);
//end
```

Scilab code Exa 12.7 Calculation of a Surface Condenser

```
printf(”\t example 12.7 \n”);
printf(”\t approximate values are mentioned in the book \n”);
V=7.5; // fps
W=250000;
CCl=0.85;
CT=1;
CL=1;
Ct=263;
UD=(CCl*CT*CL*Ct*(V^(1/2)));
printf(”\t design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n”,UD);
A=(W/8);
printf(”\t area is : %.0f ft^2 \n”,A);
```

```
a1=0.229; // ft^2/ft, table 10
at=0.475; // in^2, table 10
t1=70;
Ts=91.72; //F
n=2;
L=26;
t2=(Ts)-((Ts-t1)/((10)^(0.000279*UD*L*n*a1/(V*at))));
printf("\t t2 is : %.1f F \n",t2); // calculation mistake in book
Go=(W*950)/((t2-t1)*500);
printf("\t circulation rate is : %.0f gpm \n",Go);
// end
```

Chapter 13

Condensation of Mixed vapors

Scilab code Exa 13.1 `Calculation of the Bubble Point`

```
printf("\t example 13.1 \n");
// at atmospheric pressure,Pt=760 mm Hg
printf("\t approximate values are mentioned in the book \n");
x(1)=0.077; // mole fraction of C4
x(2)=0.613; // mole fraction of C5
x(3)=0.310; // mole fraction of C6
printf("\t for T 100 F \n");
Pp(1)=3170; // vapour pressure of C4, from fig 13.3
Pp(2)=790; // vapour pressure of C5,from fig 13.3
Pp(3)=250; // vapour pressure of C6,from fig 13.3
i=1;
while(i<4)
    p(i)=(Pp(i)*x(i));
    printf(" \n x(i)      Pp(i)      p(i) \n "+string(x(i))+"      "+string(Pp(i))+"      "+string(p(i))+" \n");
i=i+1;
end
pt=p(1)+p(2)+p(3);
printf("\t total pressure is : %.1f mm Hg \n",pt);
```

```
printf("\t pressure is too high \n");
printf("\t for T 96 F \n");
Pp(1)=2990; // vapour pressure of C4, from fig 13.3
Pp(2)=725; // vapour pressure of C5,from fig 13.3
Pp(3)=229; // vapour pressure of C6,from fig 13.3
i=1;
while(i<4)
    p(i)=(Pp(i)*x(i));
    printf(" \n x(i)       Pp(i)       p(i) \n "+
       string(x(i))+"       "+string(Pp(i))+"       "+
       string(p(i))+" \n");
i=i+1;
end
pt=p(1)+p(2)+p(3);
printf("\t total pressure is : %.1f mm Hg \n",pt);
printf("\t pressure is too low \n");
printf("\t for T 97 F \n");
Pp(1)=3040; // vapour pressure of C4, from fig 13.3
Pp(2)=740; // vapour pressure of C5,from fig 13.3
Pp(3)=234; // vapour pressure of C6,from fig 13.3
i=1;
while(i<4)
    p(i)=(Pp(i)*x(i));
    printf(" \n x(i)       Pp(i)       p(i) \n "+
       string(x(i))+"       "+string(Pp(i))+"       "+
       string(p(i))+" \n");
i=i+1;
end
pt=p(1)+p(2)+p(3);
printf("\t total pressure is : %.1f mm Hg \n",pt);
i=1;
while(i<4)
    y(i)=(Pp(i)*x(i)/pt);
    printf("\n x(i)       y(i) \n "+string(x(i))+"
           "+string(y(i))+" \n");
    i=i+1;
end
printf("\t solution for b \n");
```

```
// Similarly at what temperature will the mixture start to boil if the system is under a pressure of 35 psia
printf("\t for T 150 F \n");
Pp(1)=6100; // vapour pressure of C4, from fig 13.3
Pp(2)=1880; // vapour pressure of C5,from fig 13.3
Pp(3)=680; // vapour pressure of C6,from fig 13.3
i=1;
while(i<4)
    p(i)=(Pp(i)*x(i));
    printf(" \n x(i)       Pp(i)       p(i) \n "+string(x(i))+"       "+string(Pp(i))+"       "+string(p(i))+" \n");
i=i+1;
end
pt=p(1)+p(2)+p(3);
printf("\t total pressure is : %.0f mm Hg \n",pt);
printf("\t pressure is too high \n");
printf("\t for T 149F \n");
Pp(1)=6050; // vapour pressure of C4, from fig 13.3
Pp(2)=1850; // vapour pressure of C5,from fig 13.3
Pp(3)=670; // vapour pressure of C6,from fig 13.3
i=1;
while(i<4)
    p(i)=(Pp(i)*x(i));
    printf(" \n x(i)       Pp(i)       p(i) \n "+string(x(i))+"       "+string(Pp(i))+"       "+string(p(i))+" \n");
i=i+1;
end
pt=p(1)+p(2)+p(3);
printf("\t total pressure is : %.0f mm Hg \n",pt);
i=1;
while(i<4)
    y(i)=(Pp(i)*x(i)/pt);
    printf("\n x(i)       y(i) \n "+string(x(i))+"       "+string(y(i))+" \n");
    i=i+1;
```

```
end
printf("\t solution for c \n");
printf("\t for T 95F \n");
K(1)=3.13; // fig 7
K(2)=0.92; // fig 7
K(3)=0.30; // fig 7
i=1;
while(i<4)
    y(i)=(K(i)*x(i));
    printf("\n x(i)      K(i)      y(i) \n "+string(x(i))+"      "+string(K(i))+"      "+string(y(i))+" \n");
    i=i+1;
end
yt=y(1)+y(2)+y(3);
printf("\t yt is : %.3f \n",yt);
printf("\t yt is too low \n");
printf("\t for T 100F \n");
K(1)=3.35; // fig 7
K(2)=1; // fig 7
K(3)=0.335; // fig 7
i=1;
while(i<4)
    y(i)=(K(i)*x(i));
    printf("\n x(i)      K(i)      y(i) \n "+string(x(i))+"      "+string(K(i))+"      "+string(y(i))+" \n");
    i=i+1;
end
yt=y(1)+y(2)+y(3);
printf("\t yt is : %.3f \n",yt);
printf("\t yt is too low \n");
printf("\t for T 102F \n");
K(1)=3.45; // fig 7
K(2)=1.02; // fig 7
K(3)=0.35; // fig 7
i=1;
while(i<4)
```

```
    y(i)=(K(i)*x(i));
    printf("\n x(i)      K(i)      y(i) \n "+string(x(i))+"      "+string(K(i))+"      "+string(y(i))+" \n");
    i=i+1;
end
yt=y(1)+y(2)+y(3);
printf("\t yt is : %.3f \n",yt);
printf("\t solution for d \n");
// The use of K values gives y, directly and permits use of the total mol fraction of yt = 1.00 as the criterion for equilibrium
printf("\t for T 150F \n");
K(1)=2.8; // fig 7
K(2)=1.01; // fig 7
K(3)=0.4; // fig 7
i=1;
while(i<4)
    y(i)=(K(i)*x(i));
    printf("\n x(i)      K(i)      y(i) \n "+string(x(i))+"      "+string(K(i))+"      "+string(y(i))+" \n");
    i=i+1;
end
yt=y(1)+y(2)+y(3);
printf("\t yt is : %.3f \n",yt);
printf("\t yt is too low \n");
printf("\t for T 153F \n");
K(1)=2.90; // fig 7
K(2)=1.06; // fig 7
K(3)=0.415; // fig 7
i=1;
while(i<4)
    y(i)=(K(i)*x(i));
    printf("\n x(i)      K(i)      y(i) \n "+string(x(i))+"      "+string(K(i))+"      "+string(y(i))+" \n");
    i=i+1;
```

```
end
yt=y(1)+y(2)+y(3);
printf("\t yt is : %.3f \n",yt);
printf("\t solution for e at pt=760mm Hg \n");
y(1)=0.077; // mole fraction of C4
y(2)=0.613; // mole fraction of C5
y(3)=0.310; // mole fraction of C6
printf("\t for T 130F \n");
K(1)=5; // fig 7
K(2)=1.65; // fig 7
K(3)=0.62; // fig 7
i=1;
while(i<4)
    x(i)=(y(i)/K(i));
    printf("\n y(i)        K(i)        x(i) \n "+string(
       y(i))+"        "+string(K(i))+"        "+string(x
       (i))+" \n");
    i=i+1;
end
xt=x(1)+x(2)+x(3);
printf("\t xt is : %.3f \n",xt);
printf("\t xt is too low \n");
printf("\t for T 120F \n");
K(1)=4.4; // fig 7
K(2)=1.4; // fig 7
K(3)=0.51; // fig 7
i=1;
while(i<4)
    x(i)=(y(i)/K(i));
    printf("\n y(i)        K(i)        x(i) \n "+string(
       y(i))+"        "+string(K(i))+"        "+string(x
       (i))+" \n");
    i=i+1;
end
xt=x(1)+x(2)+x(3);
printf("\t xt is : %.3f \n",xt);
printf("\t xt is high \n");
printf("\t for T 123F \n");
```

```
K(1)=4.6; // fig 7
K(2)=1.49; // fig 7
K(3)=0.545; // fig 7
i=1;
while(i<4)
    x(i)=(y(i)/K(i));
    printf("\n y(i)      K(i)      x(i) \n "+string(y(i))+"      "+string(K(i))+"      "+string(x(i))+" \n");
    i=i+1;
end
xt=x(1)+x(2)+x(3);
printf("\t xt is : %.3f \n",xt);
printf("\t dew point at 760mm is 123F \n");
printf("\t dew point at 35psia \n");
printf("\t for T 174F \n");
K(1)=3.7; // fig 7
K(2)=1.38; // fig 7
K(3)=0.58; // fig 7
i=1;
while(i<4)
    x(i)=(y(i)/K(i));
    printf("\n y(i)      K(i)      x(i) \n "+string(y(i))+"      "+string(K(i))+"      "+string(x(i))+" \n");
    i=i+1;
end
xt=x(1)+x(2)+x(3);
printf("\t xt is : %.3f \n",xt);
printf("\t dew point is 174F \n");
// end
```

Scilab code Exa 13.2 Calculation of the Bubble Point and Vapor Composition by Rela

```
printf("\t example 13.2 \n");
```

```
printf("\t approximate values are mentioned in the book \n");
printf("\t bubble point at 95F and 14.7psia \n");
x(1)=0.077; // mole fraction of C4
x(2)=0.613; // mole fraction of C5
x(3)=0.310; // mole fraction of C6
K(1)=3.13; // fig 7
K(2)=0.92; // fig 7
K(3)=0.3; // fig 7
a(1)=3.4; // a= alpha
a(2)=1;
a(3)=0.326;
i=1;
while(i<4)
    Z(i)=(a(i)*x(i));
    i=i+1;
end
Zt=Z(1)+Z(2)+Z(3);
printf("\t Zt is : %.3f \n",Zt);
i=1;
while(i<4)
    y(i)=(a(i)*x(i)/(Zt));
    printf(" \n x(i)       K(i)       a(i)       Z(i)       y(i) \n "+string(x(i))+"       "+string(K(i))+"       "+string(a(i))+"       "+string(Z(i))+"       "+string(y(i))+" \n");
    i=i+1;
end
yt=y(1)+y(2)+y(3);
printf("\t yt is : %.3f \n",yt);
K2=(y(2)/x(2));
printf("\t K2 is : %.3f \n",K2);
printf("\t bubble point is 102 \n"); // from fig 7 , comparing K2 value
printf("\t dew point at 130F and 14.7psia \n");
y(1)=0.077; // mole fraction of C4
y(2)=0.613; // mole fraction of C5
y(3)=0.310; // mole fraction of C6
```

```
K(1)=5; // fig 7
K(2)=1.65; // fig 7
K(3)=0.62; // fig 7
a(1)=3.03; // a= alpha
a(2)=1;
a(3)=0.376;
i=1;
while(i<4)
    Z(i)=(y(i)/a(i));
    i=i+1;
end
Zt=Z(1)+Z(2)+Z(3);
printf("\t Zt is : %.3f \n",Zt);
i=1;
while(i<4)
    x(i)=(Z(i)/Zt);
    printf(" \n y(i)      K(i)       a(i)      Z(i)      x(i) \n "+string(y(i))+"      "+string(K(i))+"      "+string(a(i))+"      "+string(Z(i))+"      "+string(x(i))+" \n");
    i=i+1;
end
xt=x(1)+x(2)+x(3);
printf("\t xt is : %.0f \n",xt);
K2=(y(2)/x(2));
printf("\t K2 is : %.2f \n",K2);
printf("\t dew point is 122F \n"); // from fig 7, comparing K2 value
// end
```

Scilab code Exa 13.3 Condenser Calculations for a Multicomponent Mixture

```
printf("\t example 13.3 \n");
printf("\t approximate values are mentioned in the book \n");
```

```
printf(”\t for condensing range \n”);
V(1)=170.5; // volume of C3,Mol/hr
V(2)=284; // volume of C4,Mol/hr
V(3)=56.8; // volume of C6,Mol/hr
V(4)=341.1; // volume of C7,Mol/hr
V(5)=284; // volume of C8,Mol/hr
Tw=283; // dew point assumption
Tb=120; // bubble point assumption
K(1)=13.75 // at 283F
K(2)=6.18 // at 283F
K(3)=1.60 // at 283F
K(4)=0.825 // at 283F
K(5)=0.452 // at 283F
i=1;
while(i<6)
    Z(i)=(V(i)/K(i));
    i=i+1;
end
Vt=V(1)+V(2)+V(3)+V(4)+V(5);
Zt=Z(1)+Z(2)+Z(3)+Z(4)+Z(5);
L(1)=170.5; // volume of C3,Mol/hr
L(2)=284; // volume of C4,Mol/hr
L(3)=56.8; // volume of C6,Mol/hr
L(4)=341.1; // volume of C7,Mol/hr
L(5)=284; // volume of C8,Mol/hr
Kl(1)=4.1 // at 283F
Kl(2)=1.39 // at 283F
Kl(3)=0.17 // at 283F
Kl(4)=0.06 // at 283F
Kl(5)=0.023 // at 283F
i=1;
while(i<6)
    Zl(i)=(L(i)*Kl(i));
    printf(” \n V(i)       K(i)       Z(i)       L(i)
             Kl(i)        Zl(i) \n ”+string(V(i))+”
            ”+string(K(i))+”       ”+string(Z(i))+”
            ”+string(L(i))+”       ”+string(Kl(i))+”
            ”+string(Zl(i))+” \n”);
```

```
    i=i+1;
end
Lt=L(1)+L(2)+L(3)+L(4)+L(5);
Zlt=Zl(1)+Zl(2)+Zl(3)+Zl(4)+Zl(5);
printf("\t total volume in vapour phase : %.1f \n",Vt);
printf("\t total Zt in vapour phase : %.1f \n",Zt);
printf("\t total volume in liquid phase : %.1f \n",Lt);
printf("\t total Zlt in liquid phase : %.1f \n",Zlt);
// Range: 283 to 270 F
// Trial: Assume V /L = 4.00.
R=4; // R=(V/L), assumption
K(1)=12.75 // at 270F
K(2)=5.61 // at 270F
K(3)=1.40 // at 270F
K(4)=0.705 // at 270F
K(5)=0.375 // at 270F
i=1;
Y(i)=V(i);
while(i<6)
    P(i)=(K(i)*R);
    L1(i)=(V(i)/(1+P(i))); // V(i)=Y(i)
    printf(" \n Y(i)      K(i)      P(i)      L1(i) \n "+string(V(i))+"      "+string(K(i))+"      "+string(P(i))+"      "+string(L1(i))+" \n");
    i=i+1;
end
L1t=L1(1)+L1(2)+L1(3)+L1(4)+L1(5);
V1t=(Vt-L1t);
R1=(V1t/L1t);
printf("\t total liquid at 270F : %.0f \n",L1t);
printf("\t total vapour at 270F : %.0f \n",V1t);
printf("\t R1 is : %.0f \n",R1);
// If the assumed and calculated values of V /L had not checked, a new value would have been assumed.
```

```
printf("\t for condensing curve \n");
R270=4; // V/L at 270, from table 13.2
R270=1.567; // V/L at 250, from table 13.2
R270=0.916; // V/L at 230, from table 13.2
R270=0.520; // V/L at 200, from table 13.2
R270=0.226; // V/L at 160, from table 13.2
H270=30835500; // 4th table in solution ,enthalpies calculated from fig 10
printf("\t heat load at 270F is : %.0f Btu/hr \n",H270);
H250=27042400; // 5th table in solution ,enthalpies calculated from fig 10
printf("\t heat load at 250F is : %.0f Btu/hr \n",H250);
Q=H270-H250;
printf("\t heat load for interval 270-250F : %.0f Btu/hr \n",Q);
qt=21203000; // 6th table in solution, calculated from fig 10
printf("\t heat load for entire range is : %.0f Btu/hr \n",qt);
M=210410; // M=sum(U*A), 6th table in solution, calculated from fig 10
w=(qt/(120-80));
printf("\t water flow rate : %.1e lb/hr \n",w);
W=95450; // flow rate of feed,lb/hr
delt=(qt/M);
printf("\t weighted delt is : %.1f F \n",delt);
q1=[0 3.4765 7.2696 10.109 13.468 17.399 21.203];
T1=[283 270 250 230 200 160 120];
plot2d(q1,T1,style=3,rect=[0,0,25,300]);
q2=[0 21.203];
T2=[283 120];
plot2d(q2,T2,style=5,rect=[0,0,25,300]);
xtitle("condensing curve","heat load,Btu/hr","temperature,F");
legend("green-differential vapour","red-vapour");
printf("\t calculation of the exchanger \n");
```

```
T1=283; // inlet hot fluid,F
T2=120; // outlet hot fluid,F
t1=80; // inlet cold fluid,F
t2=120; // outlet cold fluid,F
L=16;
Nt=774;
n=4;
row=62.5;
Qs=21203000; // Btu/hr
Qw=(w*1*(120-80));
printf(”\t heat absorbed by water : %.4e Btu/hr \n”,Qw);
Mavg=84; // This corresponds very closely to hexane (mol. Wt. = 86.2) whose properties will be used throughout.
Qc=W*(0.6/2)*(283-120);
printf(”\t condensate sensible heat load: %.2e Btu/hr \n”,Qc);
S=(Qc*(100/Qs));
printf(”\t submergence : %.0f \n”,S);
Tc=((T1+T2)/2); // caloric temperature of hot fluid,F
printf(”\t caloric temperature of hot fluid is : %.0f F \n”,Tc);
tc=((t1+t2)/2); // caloric temperature of cold fluid,F
printf(”\t caloric temperature of cold fluid is : %.0f F \n”,tc);
printf(”\t hot fluid:shellside,vapour \n”);
Nts=(774*(1-.22)); // as submergence is 22%
printf(”\t unmerged tubes : %.0f \n”,Nts);
Gs=(W/(L*(Nts^(2/3)))); // eq 12.43
printf(”\t Gs is : %.1f \n”,Gs);
Ho=200; // assumption
printf(”\t cold fluid:inner tube side,water \n”);
at1=0.302; // flow area, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
```

```
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.1e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*row));
printf("\t V is : %.2f fps \n",V);
hi=1355; // fig 25
ID=0.62;
OD=0.75;
hio=((hi)*(ID/OD)); //Hio=(hio/phyp), using eq.6.5
printf("\t Correct hio to the surface at the OD is : %.2e Btu/(hr)*(ft^2)*(F) \n",hio);
tw=(tc)+(((Ho)/(hio+Ho))*(Tc-tc)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
tf=(Tc+tw)/(2); // from eq 12.19
printf("\t tf is : %.0f F \n",tf);
kf=0.077; //table 4, Btu/(hr)*(ft^2)*(F/ft)
sf=0.60; // from table 6
muf=0.21; // cp, from fig 14
ho=206; // Btu/(hr)*(ft^2)*(F), from fig 12.9
printf("\t Correct ho to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
Uc=((hio)*(ho)/(hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
Ac=(Qw/(174*delt));
printf("\t clean surface required for condensation : %.2e ft^2 \n",Ac);
As=1210*0.22;
printf("\t clean surface required for subcooling : %.0f ft^2 \n",As);
AG=As+Ac;
printf("\t total clean surface : %.0f ft^2 \n",AG);
UC=(Qw/(AG*delt));
printf("\t weighted clean overall coefficient : %.0f Btu/(hr)*(ft^2)*(F) \n",UC);
A2=0.1963; // actual surface supplied for each tube,
```

```
ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.2e ft^2 \n",A);
UD=((Qw)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
Rd=((UC-UD)/((UD)*(UC))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.5f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for annulus \n");
B=30;
as=33*0.25*(30/144)*1; // eq 7.1
printf("\t as is : %.2f ft^2 \n",as);
Gs=(W/as);
printf("\t Gs is : %.2e lb/(hr)*(ft^2) \n",Gs); // eq 7.2
mu1=0.0218; // at 283F
De=0.0608; // ft , from fig 15
Res=(De*Gs)/(mu1);
printf("\t reynolds number is : %.2e \n",Res);
f=0.00125; // fig 29
N=(12*L/B); // eq 7.43
printf("\t number crosses : %.0f \n",N);
row1=0.527; //lb/ft^3
s=0.00844;
Ds=2.75; // ft
delPs=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(1)))/(2); // using eq 12.47,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t pressure drop  for inner pipe \n");
mu2=1.74; // fig 14
D=0.0517; // ft
s=1;
Ret=(D*Gt/mu2);
printf("\t reynolds number : %.2e \n",Ret);
f=0.00019; // ft^2/in^2
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(1)*(1))); // using eq.7.45,psi
```

```
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.23; // X1=((V^2)/(2*g)),using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPa is 10 psi \n");
// end
```

Scilab code Exa 13.4 Vapor

```
printf("\t example 13.4 \n");
printf("\t approximate values are mentioned in the book \n");
vA=2*3.7+(7.4); // for steam
vB=14.8+(2*7.4); // for CO2
MA=18;
MB=44;
T=403; // K
Pt=3.04; // atm
kd=(0.0166)*(((403^(3/2))/(3.04*(14.8^(1/3)+29.6^(1/3))^(2)))*((1/18)+(1/44))^(1/2)); // eq 13.31
printf("\t diffusivity is : %.2f ft^2/hr \n",kd);
// end
```

Scilab code Exa 13.5 Calculation of a Steam Carbon Dioxide Condenser

```
printf("\t example 13.5 \n");
// for a Basis of one Hour
printf("\t approximate values are mentioned in the book \n");
c(1)=1544; // Flow rate of CO2, Lb/hr
```

```
h(1)=4500; // Flow rate of H20, Lb/hr

c(2)=35; //Flow rate of CO2, Mol/hr
h(2)=250;//Flow rate of H20, Mol/hr

t(1)=c(1)+h(1); //Total flow rate , Lb/hr
t(2)=c(2)+h(2); //Total flow rate, Mol/hr

Pt = (30+14.7)/(14.7); //Total Pressure in atm
printf(”\t Pt is %.2f\n”,Pt);
Pw = ( h(2)/t(2) )*Pt; //Partial pressure of Water in atm

printf(”\t Partial Pressure of Water: %.2f atm \n”,Pw);

Tw = 267; // from table 7 at 2.68atm
Mm = (t(1)/t(2));

printf(”\t mean molecular weight : %.1f \n”,Mm);
// weighted temperature difference
// overall balance
//for Inlet
Pv=2.68; // water vapour pressure, atm
Pg=Pt-Pv; // Inert pressure
//for Exit
Pw1 = 0.1152 // Partial pressure of water at 120 F
Pv1 = 0.115; // Water vapor pressure
Pg1 = 2.935; // Inert pressure

w1 = 250; //Pound mols steam inlet
w2 = c(2)*(Pv1/Pg1);
printf(”\tPound mols steam exit:%.2f\n”,w2);
w3 = w1 - w2;
printf(”\tPound mols steam condessed:%.2f\n”,w3);
//Assume points at 267, 262, 255,225,150,120 deg F
//For the interval from 267 to 262 F

```

```
Pv2 = 2.49; // From table 7 at 262 F
Pg2 = Pt - Pv2; //Inert pressure
printf(”\tPg is %.2f”,Pg2);

w4 = c(2) * (Pv2/Pg2); //Mol steam remaining
w5 = h(2) - w4; //Mol steam condensed

printf(”\tMol steam remaining:%.0f\n”,w4);
printf(”\tMol steam condensed:%.0f\n”,w5);

h1 = (w5*18*937.3) + (0.46*(267-262) * w5 * 18); // Heat of condensation
h2 = (w4 * 18 * 0.46*(267-262)); //Heat from uncondensed steam
h3 = c(1)*0.22*5.0; //Heat from noncondensable

printf(”\tHeat of condensation:%.2e\n”,h1);
printf(”\tHeat from uncondensed steam:%.2e\n”,h2);
printf(”\tHeat from noncondensable:%.1e\n”,h3);

ht = h1+h2+h3;//Total heat
printf(”\tTotal heat:%.0f\n”,ht);

//Similarily calculating the Heat balance for other intervals
printf(”\tInterval,F\tTotal Heat\n\t267-262\t1,598,000\n\t262-255\t1,104,000\n\t255-225\t1,172,000\n\t225-150\t751,000\n\t150-120\t177,000\n\tTotal\t4,802,000\n”);

w=4802000/(115-80); //Total water
printf(”\tTotal water: %.2e\n”,w);
//Water coefficient
Nt = 246;
at1 = 0.302;
n = 4;

at = Nt * (at1/(144*n)); // From eq 7.48
```

```
printf("\tat is %.3f ft^2\n",at);
Gt = w/at;
printf("\tGt is %.2e lb/(hr)(ft^2)\n",Gt);
ro = 62.5;
V = Gt/(3600*ro);
printf("\tV is %.2f fps\n",V);
hi = 1120; // From fig. 25
ID = 0.62;
OD = 0.75;
hi0= hi *(ID/OD); //From eq 6.5
printf("\thi0 is %.0f\n",hi0);
//Mean properties at 267 F
c = ((c(1)*0.22)+(h(1)*0.46))/t(1);  // Calculation mistake in Book
printf("\tMean c:%.3f Btu/(lb)(F)\n",c);

k = ((c(1)*0.0128)+(h(1)*0.015))/t(1); // Calculation mistake in Book
printf("\tMean k:%.4f Btu/(hr)(ft^2)(F/ft)\n",k);

mu = (((c(1)*0.019)+(h(1)*0.0136))/t(1))* 2.42; // Calculation mistake in Book
printf("\tMean mu:%.4f lb/(hr)(ft)\n",mu);

ID1 = 21.25;
C = 0.25;
B = 12;
PT = 1.0;

as = ID1 * C * (B/(144*PT)); //From eq 7.1
printf("\tas is %.3f ft^2\n",as);
Gs = t(1)/as //From eq 7.2
printf("\tGs is %.3e lb/(hr)(ft^2)\n",Gs);
Ds = 0.0792; // From Fig 28
Res = Ds * (Gs/0.0363); // From eq 7.3
printf("\tRes is %.2e\n",Res);
jH = 102; // From Fig 28
x = ((c*mu)/k)^(1/3);
```

```
printf(”\t(c.mu/k)ˆ1/3 is %.0f\n”,x);
h0 = jH * 0.0146 * (x/Ds); //From eq 6.15b
printf(”\th0 is %.0f\n”,h0);
y = 0.62 // y = (mu/ro ∗ kd)ˆ(2/3)
z = 1.01; // z = ((c∗mu)/k)ˆ(2/3)

K = (h0*z)/(0.407*Mm*y); //KG = K/p0f
printf(”\tK is %.2f\n”,K);
//at point 1
Tg = 244; // F
tW = 115;
delt=(Tg-tW);
printf(”\t delt is %.0f F \n”,delt);
```

Scilab code Exa 13.6 `Calculation of a Condenser`

```
printf(”\t example 13.6a \n”);
printf(”\t approximate values are mentioned in the book \n”);

ds=[0 10 20 30 40 50 60 70 80 90 100];
tmp=[90 145 180 208 234 260 286 312 338 367 400];
clf();
subplot(3,2,1);
plot2d(ds,tmp,style=2,rect=[0,80,100,400]);
xtitle(”Plot of ASTM curve”,boxed=1);
xlabel(”Per cent distilled off”);
ylabel(”Temperature  F ”);

//From the plotted ASTM curve and reference line
s = (312-145)/60; // (70% − 10%)/60%
printf(”\tSlope of ASTm = %.2f  F \n”,s);
ap = (180+260+338)/3; // (20% +50% +80%)/3
printf(”\tAverage 50prcnt point = %.1f  F \n”,ap);

```

```
fc = 38; // F , from Fig.13.8
printf("\t50prcnt point ASTM = 50prcnt point flash curve = %.0f  F \n",fc);
fc1 = ap - fc; // F , fixing first point on EFC
printf("\t50prcnt on EFC = %.0f  F \n",fc1);

s1 = 1.65; // ( F /%) from fig 13.10, upper curve
ten = 221 - 40*s1; //
printf("\t10prcnt on EFC = 50prcnt - 40prcnt = %.0f  F \n",ten);
sty = 221 + 20*s1; //
printf("\t70prcnt on EFC = 50prcnt + 20prcnt %.0f  F \n",sty);

//Draw this line as a reference through the 50% point. Calculate the flash curve for different percentages off

//0% off
printf("\n\t0 prcnt off:\n");
dela = 90 - 117; // Step (8)
printf("\t\tDelT ASTM = %.0f  F \n",dela);
delE = dela * 0.50; // Step (9)
printf("\t\tDelT EFC = %.1f  F \n",delE);
FE = 139 - delE; // Step (10)
printf("\t\t F EFC = %.1f\n",FE);
//end
ov=13300; //lb/hr
ng=90;//lb/hr
mng=50;// mol. wt
st=370;//lb/hr
avG=50;// F  API
//For 80%
ouc=ov*0.80;//lb/hr
printf("\toil uncondensed = %.0f lb/hr\n",ouc);
avB=269;// F ,from Fig. 13.13
printf("\tAverage boiling point from the EFC at 1 atm = %.0f  F\n",avB);
```

```
avB1=avB+17;// F ,from Fig. 13.13
printf("\tAverage boiling point from the EFC at 19.7 psia = %.0f F\n",avB1);
mwt=113;//mol. wt
mtoc=ouc/mwt;
printf("\tThe moles of oil still to be condensed = %.1f\n",mtoc);
mg1=ng/mng;
ms1=st/18;
tm=mg1+ms1+mtoc;
printf("\t\tMols gas = %.2f\n\t\tMols steam = %.1f\n",mg1,ms1);
printf("\t\t\t    -----\n\t\tMols total = %.1f\n",tm);
tp=19.7;//psia
poil=(mtoc/tm)*tp;//psia
printf("\tPartial pressure of oil = %.1f psia\n",poil);
pgas=(mg1/tm)*tp;//psia
printf("\tPartial pressure of NC gas = %.3f psia\n",pgas);
tm(1)=95;// F
tm(2)=127;// F
tm(3)=163;// F
tm(4)=205;// F
tm(5)=240;// F
pp(1)=6.73;
pp(2)=9.40;
pp(3)=12.25;
pp(4)=14.64;
pp(5)=15.65;
psat(1)=0.815;//From steam table
psat(2)=2.050;//From steam table
psat(3)=5.09;//From steam table
psat(4)=12.77;//From steam table
psat(5)=24.97;//From steam table
printf("\n\t\tCALCULATION OF DEW POINT OF THE STEAM\n");
```

```
printf(”\tT, F \t[pt - (poil+pNC)] = psteam\tpsat(steam tables)\n”);
i=1;
while(i<6)
    ps=tp-pp(i);
    printf(”\t”+string(tm(i))+”\t%.1f\t %.2f\t\t%.2f\t%.3f\n”,tp,pp(i),ps,psat(i));
    i=i+1;
end
subplot(3,2,2);
plot2d(psat,tm,style=3,rect=[0,25,90,250]);
xtitle(”Computed pressure of steam”,boxed=1);
xlabel(”Pressure of steam, psi”);
ylabel(”Temperature F ”);

ds=6.417;//psia, at 173 F ,
printf(”\tAt 173 F , the dew point of steam, psat = %.3f psia\n”,ds);
pd1=tp-ds;//psia
printf(”\tpoil + pNC = %.2f psia\n”,pd1);
x=((tp*ms1)/ds)-(ms1+mg1);// mols oil
printf(”\tOil = %.2f mols oil\n”,x);
mw=85;//From fig. 13.14
printf(”\tThe molecular weight of the vapors is %.0f\n”,mw);
lv=x*mw;//lb
printf(”\tLb/hr vapor = %.0f\n”,lv);
prc=((ov-lv)*100)/ov;//%
printf(”\tpercent Condensed = %.0f\n”,prc);
printf(”\n\t\t\tOIL CONDENSING CURVE\n”);
printf(”\tprcnt\tCondensables\t\tAv BP on EFC\t\t50 API\t\tMol oil\t\tMol NC gas\tMol steam\tMol total\tTotal pressure\tPartial pressure\tPartial pressure\tCond temp, F \n\t\tlb.hr\t\t14.7 psia F \t19.7 psia, F \tmol.wt\t\t\t\t\t\t\t\t\tpsia\t\toil,psia\t\tNC gas, psia\n”);
mo(1)=107.5;
mo(2)=94.3;
```

```
mo(3)=77.7;
mo(4)=57.4;
mo(5)=31.8;
mo(6)=17.1;
mo(7)=8.9;
i=1;
while(i<8)
    mt(i)=mo(i)+mg1+ms1;
    ppo(i)=(mo(i)/mt(i))*tp;
    ppg(i)=(mg1/mt(i))*tp;
    i=i+1;
end
printf("\t
    ——————————————————————————————————————————————————————————————————
    n");
printf("\t100\t13330\t\t300\t\t317\t\t124\t\t%.1f\t\t1.8\t\t20.6\t\t%.1f\t\t19.7\t\t%.1f\t\t\t%.3f\t\t\t305\n",mo(1),mt(1),ppo(1),ppg(1));
printf("\t80\t10664\t\t269\t\t286\t\t113\t\t%.1f\t\t1.8\t\t20.6\t\t%.1f\t\t19.7\t\t%.1f\t\t\t%.3f\t\t\t277\n",mo(2),mt(2),ppo(2),ppg(2));
printf("\t60\t7998\t\t239\t\t256\t\t103\t\t%.1f\t\t1.8\t\t20.6\t\t%.1f\t\t19.7\t\t%.1f\t\t\t%.3f\t\t\t240\n",mo(3),mt(3),ppo(3),ppg(3));
printf("\t40\t5332\t\t207\t\t224\t\t93\t\t%.1f\t\t1.8\t\t20.6\t\t%.1f\t\t19.7\t\t%.1f\t\t\t%.3f\t\t\t205\n",mo(4),mt(4),ppo(4),ppg(4));
printf("\t20\t2666\t\t178\t\t195\t\t84\t\t%.1f\t\t1.8\t\t20.6\t\t%.1f\t\t19.7\t\t%.1f\t\t\t%.3f\t\t\t163\n",mo(5),mt(5),ppo(5),ppg(5));
printf("\t10\t1333\t\t155\t\t172\t\t78\t\t%.1f\t\t1.8\t\t20.6\t\t%.1f\t\t19.7\t\t%.1f\t\t\t%.3f\t\t\t127\n",mo(6),mt(6),ppo(6),ppg(6));
printf("\t5\t667\t\t141\t\t158\t\t75\t\t%.1f\t\t1.8\t\t20.6\t\t%.1f\t\t19.7\t\t%.1f\t\t\t%.3f\t\t\t95\n",mo(7),mt(7),ppo(7),ppg(7));

//Trail 1:
```

```
m=78;//50   API mol. wt. for condesables 1333
vap=(ov*0.10)/78;//Mol/hr
printf("\n\t\t\t\tMol/hr\n\tOil vapor\t\t%.1f\n\tNC gas\t\t\t%.1f\n\tSteam\t\t\tX\n\tTotal\t\t\t18.9+X\n",vap,mg1);
vap1=vap+mg1;//Mol/hr
psteam=5.09;//psia, For 163 F
x1=(psteam*vap1)/(tp-psteam);//mols steam
printf("\tX = %.2f mols steam\n",x1);
tv=vap1+x1;
printf("\n\t\t\tMol/hr\tmf\tmf*pt = p-partial\n");
mf1=vap/(tv);
ppar1=mf1*tp;
printf("\tOil vapor\t%.1f\t%.3f\t%.2f\n",vap,mf1,ppar1);
mf2=mg1/tv;
ppar2=mf2*tp;
printf("\tNC gas\t\t%.1f\t%.3f\t%.2f\n",mg1,mf2,ppar2);
mf3=x1/tv;
ppar3=mf3*tp;
printf("\tSteam\t\t%.2f\t%.3f\t%.2f\n",x1,mf3,ppar3);
tot1=vap+mg1+x1;
tot2=mf1+mf2+mf3;
tot3=ppar1+ppar2+ppar3;
printf("\tTotal\t\t%.2f\t%.3f\t%.2f\n",tot1,tot2,tot3);
//Error was found. So trail 2 is done in a similar way
printf("\n\tSimilarly,\n\tT, F \tOil cond, prcnt\tOil cond, lb\tSteam cond,lb\n");
printf("\t173\t74\t\t9863\t\t0\n\t163\t85\t\t11350\t\t204\n\t127\t97.5\t\t13000\t\t357\n\t95\t100\t\t13330\t\t370\n");
//Condensing curve
printf("\n\t\t\tOil\t\t\t\tSteam\n\t
--------------------------------------------------------------------------------
```

```
n\tTc, F \tHv,vapor\tHl,liquid\tHg or Hv,\tHl,liquid\n\t\t\t\t\t\tgas or vapor\n");
printf("\t--------------------------------------------------------------------------------n")
printf("\t305\t368\t\t242\t\t1197.0\t\tSuperheated\n\t277\t359\t\t225\t\t1184.1\t\tSuperheated\n\t240\t337\t\t204\t\t1167.0\t\tSuperheated\n\t205\t322\t\t185\t\t1150.6\t\tSuperheated\n\t173\t310\t\t168\t\t1135.4\t\t140.9\n");//From fig.11 in Appendix and steam tables
//Heat load
//305 F :
hvv=368;
hvg=1197.0;
olv=ov*hvv;
stm=st*hvg;
ncg=ng*(0.46*273);
thh=olv+stm+ncg;
printf("\n\t\t\t\tH\t\tq\n");
printf("\tOil vapor\t\t%.2e\n\tSteam\t\t\t%.2e\n\tNC gas\t\t\t%.2e\n\t\t\t\t---------\n\t\t\t\t%.4e\t0\n",olv,stm,ncg,thh);
//Similarily at other temperatures
ttp(1)=305;// F
ttp(2)=277;// F
ttp(3)=240// F
ttp(4)=205;// F
ttp(5)=173;// F , Dew point of steam
ttp(6)=163;// F
ttp(7)=127;// F
ttp(8)=95;// F

hld(1)=0;//million Btu
hld(2)=0.55;//milllion Btu
hld(3)=1.2;//million Btu
hld(4)=1.75;//million Btu
hld(5)=2.3;//million Btu
```

```
hld(6)=2.73;//million Btu
hld(7)=3.3;//million Btu
hld(8)=3.66;//million Btu
subplot(2,2,3);
plot2d(hld,ttp,style=6,rect=[0,60,3.8,320]);
xtitle("Condensation of mixed hydrocarbons with gas and steam",boxed=1);
xlabel("Heat load, million Btu");
ylabel("Temperature  F ");
//summary
dp=3042800;//Btu/hr
ttt=3638400;//Btu/hr
i2s=thh-dp;//Btu/hr
printf("\tInlet to steam dew point = %.4eBtu/hr\n",i2s);
so=dp-1735900;//Btu/hr
printf("\tSteam dew point to outlet = %.4e Btu/hr\n",so);
totl=i2s+so;//Btu/hr
printf("\tTotal\t\t\t= %.4e Btu/hr\n",totl);
twa=ttt/(120-85);
printf("\tTotal water = %.2e lb/hr\n",twa);
wt=85+((1306900/ttt)*35);// F
printf("\tWater temperature at dew point of steam = %.0f  F\n",wt);
//Weighted true temperature difference, delT:
    //Inlet to dew point of steam:
delq=2331500;
delt1=122.2;
UA1=delq/delt1;
printf("\tUA = %.0f\n",UA1);
printf("\n\tDew point of steam to oulet\n");
printf("\tq\tdelq\tTc\ttw\tdelTav\t(delq/delTav) = UA\n");
printf("\t------------------------------------------------------------------------\n");
q(1)=2331500;
```

```
q(2)=2500000;
q(3)=2750000;
q(4)=3000000;
q(5)=3250000;
q(6)=3500000;
q(7)=3638000;
i=1;
while(i<7)
    dq(i)=q(i+1)-q(i);
    i=i+1;
end
dpt(1)=173;
dpt(2)=169;
dpt(3)=161;
dpt(4)=149;
dpt(5)=134;
dpt(6)=112;
dpt(7)=95;
dtw(1)=97.5;
dtw(2)=96;
dtw(3)=93;
dtw(4)=91;
dtw(5)=89;
dtw(6)=86;
dtw(7)=85;
i=1;
tua=0;
while(i<7)
    dpdelt(i)=((dpt(i+1)-dtw(i+1))+(dpt(i)-dtw(i)))/2;
    UA(i)=dq(i)/dpdelt(i);
    tua=tua+UA(i);
    i=i+1;
end
printf("\t2331500\t......\t173\t173\t97.5\n");
i=1;
while(i<7)
    printf("\t"+string(q(i+1))+"\t"+string(dq(i))+"\
```

```
t"+string(dpt(i+1))+"\t"+string(dtw(i+1))+"\t"+string(dpdelt(i))+"\t%.0f\n",UA(i));//from Fig. 13.16
i=i+1;
end

printf("\t\t\t\t\t\t%.0f\tUA = sigma{delq/delt}\n",tua);
wdt=1306900/tua;// F
printf("\tWeighted delt = %.1f  F\n",wdt);
owdt=ttt/(tua+UA1);// F
printf("\tOverall weighted temperature difference = %.1f  F \n",owdt);
printf("\tThe uncorrected LMTD is 60.1 F \n");
//end


printf("\t example 13.6b \n");
printf("\t approximate values are mentioned in the book \n");

// EXCHANGER
//Shell side
Id = 27; // inches
Bs = 16; // inches
Ps = 1; // passes

//Tube side
N = 286; // number
l = 12; // inches
Od = 1; // inch
BWG = 14; // bWG
Ptc = 1.25; //inches
Ps1 = 8; // passes

//Clesan surface requirements

//Head load inlet to dew point of steam
```

```
st = 2331500; // Btu/hr
delT = 122.2 //  F
hio = 700; // Btu/((hr)(ft^2)( F )) for water

//From table 13.4 at inlet
NC = 1.8; //NC gas, mol/hr
sm = 20.6;// steam, mol/hr
tt = NC + sm;// mol/hr
printf("\tNC gas + steam is %.1f mol/hr\n",tt);
pN = tt/129.9; // mol/hr
printf("\tpercentage NC gas is %.4f\n",pN);

//From Fig 13.17
hn = 205; //Btu/((hr)(ft^2)( F ))
//At dew point of steam
No=40.75; // Mol/hr
t1 = tt + No; // Mol/hr, total
pN1 = tt/t1; // Mol/hr, %NC
printf("\tpercentage NC is %.3f\n",pN1);

//From  fig 13.7
hn1 = 140; //Btu/((hr)(ft^2)( F ))
lm = 136.5; //Btu/((hr)(ft^2)( F ))
delT = 122.2; //  F
Ac1 = st/(lm * delT); // ft^2
printf("\tAc1 = Q/(U * delT) is %.1f ft^2\n",Ac1);

//At dew point of steam to oulet
sm1 = 20.64; // Mol/hr , Steam
t2 = NC + sm1; // total, Mol/hr
printf("\tNC gas + steam is %.1f mol/hr\n",t2);
pN1 = NC/t2; // % NC gas
printf("\tpercentage NC gas is %.3f \n",pN1);

Uc = 212; // From Fig 13.17, weighted for oil and steam

//At outlet, steam = negligible
```

```

Uc = 15;//From Fig 13.17

//Log mean overall coefficient
lm = 74.5; // Btu/((hr)(ft^2)( F )) , From Fig 13.17
delT = 44.8; //  F
Ac2 = 1306900/(lm * delT);
printf("\tAc2 is %.0f ft^2\n",Ac2);

hl = 770000; // Btu/hr
printf("\tHeat of Liquid(50 API ) is %.1ef\n",hl);
wr = (hl/3638400)*35; //  F
printf("\tWater rise = %.1f  F \n",wr);

LMTD = 66.3; // F
U1=50 //for free convection
As = hl/(U1*LMTD);// ft^2
printf("\tAs = %.1f ft^2\n",As);
Ac = Ac1 + Ac2 + As; //ft^2
printf("\tTotal clean surface %.0f ft^2\n",Ac);

Uc = 3638400/(Ac * 75.5); // Btu/((hr)(ft^2)( F ))
printf("\tClean overall coefficient Uc = %.1f Btu/((
    hr)(ft^2)( F ))\n",Uc);

x = 0.2618; // ft , from table 10
A = N * l * x; //ft^2
Ud = 3638400/(A * 75.5);
printf("\tDesign coefficient Ud is %.1f\n",Ud);
Rd =(Uc - Ud)/(Uc * Ud); // ((hr)(ft^2)( F ))/Btu
printf("\tDirt factor Rd is %.4f ((hr)(ft^2)( F ))/
    Btu\n",Rd);

yo = (As/Ac)*A; // ft^2
printf("\tSubmerge = %.0f ft^2 of surface\n",yo);
//end
```

Chapter 14

Evopartion

Scilab code Exa 14.1 `Calculation of Evaporator Surface`

```
printf("\t example 14.1 \n");
printf("\t approximate values are mentioned in the book \n");

t1 = 300; // F
t2 = 226; // F
bs = 700; // Btu/((hr)(ft^2)( F ))
//Heat Balance
Qv = 10000 * 961; // Btu/hr
printf("\tQevap is %.2e Btu/hr\n",Qv);
Q3 = 10550 * 910; //Btu/hr
printf("\tQ300 F is %.2e Btu/hr\n",Q3);

delT = t1-t2; // F
printf("\tTemperature head = %.0f  F \n",delT);
Ud = bs * 0.865;
printf("\tOverall coefficient %.0f\n",Ud);
A = Qv/(Ud * delT); //ft^2
printf("\tSurface required is %.0f ft^2\n",A); // Wrong calculation in book
//end
```

Scilab code Exa 14.2 Calculation of a Triple effect Forward feed Evaporator

```
printf("\t example 14.2 \n");
printf("\t approximate values are mentioned in the book \n");

wf = 50000; //  lb/hr
sf = wf * 0.10; //  lb/hr
tp = sf/0.50; // lb/hr
printf("\tTotal product is %.0f lb/hr\n",tp);
te = wf - tp;
printf("\tTotal evaporation is %.0f lb/hr\n",te);
cf = 1.0;
tF = 100;  //  F
T1 = 244; //  F
T2 = 125; //  F
U1=600; // Btu/((hr)*(ft^2)*( F ))
U2=250; // Btu/((hr)*(ft^2)*( F ))
U3=125; // Btu/((hr)*(ft^2)*( F ))

T = T1-T2;
printf("\tTotal temperature difference is delT%.0f  F \n",T);
df = (26.70- 1.95)/3; // psi/effect
printf("\tAverage pressure difference is delP%.2f psi/effect \n",df);

printf("\n\t\t\t\t\tPressure, psia\t\t delP, psi \t Steam or vapor,  F  \t lambda, Btu/lb\n\tSteam chest, 1st effect \t 26.70 \t\t\t .... \t\t Ts = 244 \t\t ls = 949 \n\tSteam chest, 2nd effect \t 18.45 \t\t\t 8.25 \t\t t1 = 224 \t\t l1 = 961 \n\tSteam chest, 3rd effect \t 10.20(20.7 in. Hg) \t 8.25 \t\t t2 = 194 \t\t l1 = 981 \n\tVapor to
```

```
condenser \t\t 1.95(26 in. Hg) \t 8.25 \t\t t2 = 125 \t\t l1 = 1022 \n");

printf("\t949*Ws + 50000*(100-224) = 961*w1\n\t961*w1 + (50000 - w1)*(224-194) = 981 * w2\n\t981*w2 + (50000-w1-w2)(194-125) = 1022 * w2\n\tw1+w2+w3 = 40000\n");
printf("\tSolving simultaneously\n");
w1=12400;
printf("\tw1 = %.2e \n",w1);
w2=13300;
printf("\tw2 = %.2e \n",w2);
w3=14300;
printf("\tw3 = %.2e \n",w3);

Wt = w1+w2+w3;
printf("\tW1-3 is %.0e \n",Wt);
Ws = 19100;
lms = 949;
lm1 = 961;
lm2 = 981;
lm3 = 1022;
Ts = 244;
t1 = 224;
t2 = 194;
t3 = 125;

A1 = (Ws * lms)/(U1*(Ts-t1)); //ft^2
printf("\tA1 is %.0f ft^2 \n",A1);
A2 = (w1*lm1)/(U2*(t1-t2)); //ft^2
printf("\tA2 is %.0f ft^2 \n",A2);
A3 = (w2 * lm2)/(U3*(t2-t3)); //ft^2
printf("\tA3 is %.0f ft^2 \n",A3);

hc = w3 * lm3; // Btu/hr, WRONG CALCULATION IN TEXT BOOK
printf("\tHeat to condenser is %.3e Btu/hr\n",hc);
wr = hc/(120-85); //lb/hr
```

```
printf("\tWater requirement is %.1e lb/hr\n",wr);
wr1 = wr/500;
printf("\t= %.0f gpm \n",wr1);
//end
```

Scilab code Exa 14.3 `Backward feed Multiple effect Evaporator`

```
printf("\t example 14.3 \n");
printf("\t approximate values are mentioned in the book \n");
//Same conditions as example 14.2
U1 = 400; //Btu/((hr)*(ft^2)*( F ))
U2 = 250; //Btu/((hr)*(ft^2)*( F ))
U3 = 175; //Btu/((hr)*(ft^2)*( F ))

w1 = 50000; // lb/hr      From example 14.2
wt = 40000; // lb/hr      From example 14.2
cf = 1; // From example 14.2

printf("\t981*w2 + 50000*(100-125) = 1022*w3\n\t961*w1 + (50000 - w3)*(125-194) = 981 * w2\n\t949*Ws + (50000-w3-w2)(194-224) = 961 * w1\n\tw1+w2+w3 = 40000\n");
printf("\tSolving simultaneously\n");
w1 = 15950;
w2 = 12900;
w3 = 11150;
lms = 949;
lm1 = 961;
lm2 = 981;
lm3 = 1022;

wt = w1+w2+w3;
printf("\tw1-3 = %.0f \n",wt);
Ws = 16950;
```

```
A1 = (Ws*lms)/(U1*20); //ft^2
printf("\tA1 is %.0f ft^2\n",A1);
A2 = (w1*lm1)/(U2*30); //ft^2
printf("\tA2 is %.0f ft^2\n",A2);
A3 = (w2*lm2)/(U3*69); //ft^2
printf("\tA3 is %.0f ft^2\n",A3);

Avs = (A1 + A2 + A3)/3; //ft^2
printf("\tAverage surface is %.0f ft^2\n",Avs);
Av1 = 3 * Avs; //ft^2
printf("\n\tWith a better distribution temperatures and pressure, Average surface is  %.0f ft^2\n",Av1);
printf("\tRecalculation\n");
Av2 = 1500; //ft^2, assume
dT1 = 28; // F
A4 = (20/dT1)*A1; //ft^2
printf("\tA1 is %.0f ft^2\n",A4);
dT2 = 41; // F
A5  = (30/dT2)*A2; //ft^2
printf("\tA2 is %.0f ft^2\n",A5);
dT3 = 50; // F
A6 = (69/50)*A3; //ft^2
printf("\tA3 is %.0f ft^2\n",A6);
del1 = 119; // F
printf("\tTs-t3 is %.0f  F \n",del1);
printf("\t\t\t\t\tPressure, psia\t\t Steam or vapor, F \t lambda, Btu/lb\n\tSteam chest, 1st effect \t 26.70 \t\t\tTs = 244 \t\t 949 \n\tSteam chest, 2nd effect \t 16.0 \t\t\t t1 = 216 \t\t 968 \n\tSteam chest, 3rd effect \t 16.4 in. Hg) \t\t t2 = 175 \t\t 992 \n\tVapor to condenser \t\t 26 in. Hg \t\t t3 = 125 \t\t l1 = 1022 \n");

w1 = 15450; //Solving again for
printf("\tw1 is %.0f\n",w1);
w2 = 13200;
printf("\tw2 is %.0f\n",w2);
```

```
w3 = 11350;
printf("\tw3 is %.0f\n",w3);
Ws = 16850;
printf("\tWs is %.0f\n",Ws);
Hc = w3 * 1022;
printf("\tHeat to condenser is %.2e Btu/hr\n",Hc);
wr = Hc/(120-85); //lb/hr
printf("\tWater requirement %.2e lb/hr\n",wr);
wr1 = wr/500;
printf("\t\t\t= %.0fgpm\n",wr1);
ec = wt/Ws;
printf("\tEconomy, lb evaporation/lb steam %.2f\n",ec);

//comparision of forward and backward feed
printf("\t\t\t\tForward\t\tBackward\n\tTotal steam, lb/hr\t19100\t\t16850\n\tCooling water, gpm\t840\t\t664\n\tTotal surface, ft^2\t4800\t\t4500");
```

Scilab code Exa 14.4 `evoparator installer`

```
printf("\texample 14.4 \n");
printf("\tapproximate values are mentioned in the book \n");
//Assumed that 37500 lb/hr of 15 psig vapor is bled from the first effect for use in thevaccum pans
printf("\n\tAVERAGE EVAPORATION PER SQUARE FOOT HEATING SURFACE FOR SUGAR EVAPORATORS\n");
printf("\tEffects\t\tWater evaporated(lb/(hr)*(ft^2))\n");
printf("\t1\t\t14-16\n\t2\t\t6-8\n\t3\t\t5-6\n\t4\t\t4-5\n\t5\t\t3-4\n");
printf("\n\tEVAPORATOR SUMMARY\n");
printf("\t______________________________________________________________
```

```
   n");
printf("\tItem\t\t\t\t\t\t\t\t\tEffects\nt\t\t\t\t\t
   ---------------------------------------------------------------------------------
   n\t\t\t\t\t1A\t\t1B\t\t2\t\t3\t\t4\t\t5\n");
printf("\t
   ---------------------------------------------------------------------------------
   n");
printf("\t1.Steam  flow, lb/hr\t\t42600\t\t38000\n\
   t2.Steam pressure, psi/in.Hg\t30\t\t30\t\t15\t\t5
   \t\t4\t\t14.5\n");
printf("\t3.Steam temp, F \t\t\t274\t\t274\t\t250\t\
   t227\t\t205\t\t181\n");
printf("\t4.delT, F \t\t\t23\t\t23\t\t21\t\t20\t\t20
   \t\t27\n\t5.Liquor temp,  F \t\t251\t\t251\t\t229
   \t\t207\t\t185\t\t164\n\t6.BPR,  F \t\t\t1\t\t1\t
   \t2\t\t2\t\t4\t\t7\n\t7.Vapor temp,  F \t\t250\t\
   t250\t\t227\t\t205\t\t181\t\t147\n\t8.Vapor
   pressure, pis/in.Hg\t15\t\t15\t\t5\t\t4\t\t14.5\t
   \t23\n\t9.Lambda, Btu/lb\t\t946\t\t946\t\t960\t\
   t975\t\t990\t\t1010\n\t10.Liquor in, lb/hr\t\
   t229000\t\t190200\t\t154000\t\t117100\t\t87800\t\
   t64000\n\t11.Liqour out, lb/hr\t\t190200\t\
   t154000\t\t117100\t\t87800\t\t64000\t\t49600\n\
   t12.Evaporation,lb/hr\t\t38800\t\t36200\t\t36900\
   t\t29300\t\t23800\t\t14400\n\t13. Brix (out)\t\t\
   t15.7\t\t19.4\t\t25.5\t\t34.4\t\t46.5\t\t50.0\n\
   t14.A, ft^2\t\t\t3500\t\t3500\t\t5000\t\t5000\t\
   t5000\t\t3500\n\t15.UD,Btu/(hr)*(ft^2)*( F )\t478
   \t\t425\t\t310\t\t264\t\t219\t\t138\n\t16.UD delT
   ,Btu/(hr)*(ft^2)\t11000\t\t9780\t\t6520\t\t5270\t
   \t4390\t\t3740\n");//BPR values from fig 14.34a
//Saturate vapor pressure above the liquour
   determined from Table 7
//Saturated steam pressure in the following effect
   determined from Table 7

t1 = 274; // F
t2 = 147; // F
```

```
t = t1-t2; // F
printf("\tTotal temperature difference in the evaporator system = %.0f  F \n",t);
bpr1 = 1; // F
bpr2 = 2; // F
bpr3 = 2; // F
bpr4 = 4; // F
bpr5 = 7; // F
bpr = bpr1 + bpr2 + bpr3 + bpr4 + bpr5; // F
printf("\tThe sum of all the BPR(from effect 1B to the fifth effect inclusive) = %.0f  F \n",bpr);
tf = t-bpr; // F
printf("\tTotal EFFECTIVE  temperature difference = %.0f  F \n",tf);
lbh = 229000; //lb/hr
tp1=212; // F
tp2=184; // F
tp3=144; // F
tp4=82; // F
tj1=243; // F
tj2=220; // F
tj3=200; // F
Ud1=231;
Ud2=243;
Ud3=230;
Ud4=214;
Ud5=217;
printf("\n\t\t\tSUGAR-JUICE HEATERS\n");
printf("\tRaw-juice heaters\t\t\t\tClear=juice heaters\n\t--------------------------------------------------------------------------------n");
rj1=lbh*(tp1-tp2)*(0.91); //Btu/hr
printf("\t1.%.0f(%.0f-%.0f)(0.91) = %.2e Btu/hr",lbh,tp1,tp2,rj1);
rj2=lbh*(tj1-tj2)*(0.91); //Btu/hr
printf("\t1.%.0f(%.0f-%.0f)(0.91) = %.1e Btu/hr\n",lbh,tj1,tj2,rj2);
```

```
printf("\tVapor temp. = 227 F \tdelT=26.6 F \t\tVapor temp. = 250 F \tdelT=15.8 F \n");
printf("\tUD=%.0f\t\t\t\t\tUD=%.0f\n",Ud1,Ud2);
A1=rj1/(26.6*Ud1);//ft^2
A2=rj2/(15.8*Ud2);//ft^2
printf("\tSurface,A=%.0f ft^2\t\t\t\tSurface,A=%.0f ft^2\n\n",A1,A2);

rj3=lbh*(tp2-tp3)*(0.90);//Btu/hr
printf("\t2.%.0f(%.0f-%.0f)(0.91) = %.2e Btu/hr",lbh,tp2,tp3,rj3);
rj4=lbh*(tj2-tj3)*(0.90);//Btu/hr
printf("\t2.%.0f(%.0f-%.0f)(0.91) = %.2e Btu/hr\n",lbh,tj2,tj3,rj4);
printf("\tVapor temp. = 205 F \tdelT=37.6 F \t\tVapor temp. = 227 F \tdelT=14.8 F \n");
printf("\tUD=%.0f\t\t\t\t\tUD=%.0f\n",Ud3,Ud4);
A3=rj3/(37.6*Ud3);//ft^2
A4=rj4/(14.8*Ud4);//ft^2
printf("\tSurface,A=%.0f ft^2\t\t\t\tSurface,A=%.0f ft^2\n\n",A3,A4);

rj5=lbh*(tp3-tp4)*(0.90);//Btu/hr
printf("\t2.%.0f(%.0f-%.0f)(0.91) = %.2e Btu/hr",lbh,tp3,tp4,rj4);
printf("\t(Use 2 heaters at 1300 ft^2 each plus 1\n\t\t\t\t\t\t\theater at 1300 ft^2 as spare)\n");
A5=rj5/(62.2*Ud5);//ft^2
printf("\tVapor temp. = 181 F \tdelT=62.2 F \n\tSurface,A=%.0f\n",A5);
printf("\t(Use 3 heaters at 100 ft^2\n\teach plus 1 heater as spare)\n\n");

v1=42600;//lb/hr
tt1=251;// F
printf("\t\t\t\tHEAT BALANCE\n");
printf("\tEffect\t\tBtu/hr\t\tEvaporation,l/hr\n");
```

```
printf("\t------------------------------------------------------------\n");
hia=v1*929*0.97;//Btu/hr
printf("\t1A.Heat in steam........%.2e\n",hia);
hla=lbh*(tt1-tj1)*0.91;//Btu/hr
hh=hia-hla;//Btu/hr
lb1=946;//Btu/lb
dif=hh/lb1;//lb/hr
printf("\t    Heating liquor.......%.2e\n\t\t\t\t%.3e\t%.0f\n",hla,hh,dif);
ltob=lbh-dif;//lb/hr
printf("\t    Liqour to 1B\n\t    = %.0f lb/hr\n",ltob);
hia1=dif*929*0.97;//Btu/hr
printf("\t1B.Heat in steam........%.2e\n",hia1);
hla1=ltob*(tt1-tt1)*0.91;//Btu/hr
hh1=hia1;//Btu/hr
dif1=hh1/lb1;//lb/hr
printf("\t    Heating liquor........%.0f\n\t\t\t\t%.3e\t%.0f\n",hla1,hh1,dif1);
dif2=ltob-dif1;//lb/hr
printf("\t    Liqour to 2d \n\t    effect=%.0f lb/hr\n",dif2);
//Similarily the values in the table are calculated

printf("\t\t\t\t\t\t\tLb/hr\n");
aa=179400;//lb/hr
bb=145500;//lb/hr
cc=19700;//lb/hr
dd=30600;//lb/hr
ee=17900;//lb/hr
ff=13100;//lb/hr
tto=aa+bb+cc+dd+ee+ff;//lb/hr
printf("\t(a) Actual evaporation..................................%.0f\n",aa);
printf("\t(b) Equivalent evaporation from vapors of\n\t     1st effect used for vaccum pans
```

```
    .......................%.0f\n",bb);
printf("\t(c) Equivalent evaporation from 1st effect
     \n\t    vapors used for clarified-juice heaters
    ..............%.0f\n",cc);
printf("\t(d) Equivalent evaporation from 2d effect
    \n\t    vapors used for clarified-and raw-juice
    heaters.....%.0f\n",dd);
printf("\t(e) Equivalent evaporation from 3d effect
    \n\t    vapors used for raw-juice heaters
    .....................%.0f\n",ee)
printf("\t(f) Equivalent evaporation from 4th effect
     \n\t    vapors used for raw-juice heaters
    .....................%.0f\n",ff);
printf("\t

    -----\n")
printf("\t     Extrapolated evaporation
    ..............................%.0f\n",tto);
esq=tto/5;//lb/hr
printf("\t\tEstimated steam quantity = %.0f lb/hr\n"
    ,esq);
aesq=80600;//lb/hr
err = esq-aesq;//lb/hr
printf("\t\tActual steam required from final heat
    balance = %.0f lb/hr\n",aesq);
printf("\t\t\t\t\t\t\tError = %.0f lb/hr\n",err);
ta=15;
Q=14575000; //Btu/hr Total hourly evaporation
Gpm=Q/(500*(t2-tp4-ta));//From equation 14.4
printf("\tGallons per minute of Water required = %.0
    f gpm",Gpm);
```

Scilab code Exa 14.5 `unit calculation`

```
printf("\texample 14.5\n");
```

```
printf("\tapproximate values are mentioned in the
   book \n");
st1=280; // F
vt6=125; // F
odT=st1-vt6; // F
printf("\tOverall temperature difference = %.0f  F \
   n",odT); //corresponding to 35 psig and 26 in. Hg
bpr(1)=10; // F
bpr(2)=8; // F
bpr(3)=7; // F
bpr(4)=6; // F
bpr(5)=5; // F
bpr(6)=5; // F
i=1;
tbpr=0;
while(i<7)
    tbpr=tbpr+bpr(i);
    i=i+1;
end
printf("\tThe estimated total BPR = %.0f  F \n",tbpr
   ); //from fig. 14.36a
edT=odT-tbpr;
printf("\tEffective temperature difference = %.0f
    F \n",edT);
printf("\n\t\t\t\tEVAPORATOR SUMMARY\n\tAll bodies
   will consist of 300 2 in. OD, 10 BWG tubes 24
   long\n");
printf("\t
   ----------------------------------------------------------------
   n");
printf("\tItem\t\t\t\t\t\t\t\t\tEffects\n\t\t\t\t\t
   ----------------------------------------------------------------
   n\t\t\t\t\t1A\t\t1B\t\t2\t\t3\t\t4\t\t5\n");
printf("\t
   ----------------------------------------------------------------
   n");
printf("\t1.Steam  flow, lb/hr\t\t20000\n\t2.Steam
   pressure, psi/in.Hg\t35\t\t14.5\t\t4\t\t7\t\t16
```

```
    .5\t\t22\n\t3.Steam temp, F \t\t\t280\t\t249\t\
    t224\t\t199\t\t174\t\t151\n\t4.delT, F \t\t\t21\t
    \t17\t\t18\t\t19\t\t18\t\t21\n\t5.Liquor temp,
     F \t\t259\t\t232\t\t206\t\t180\t\t156\t\t130\n\
    t6.BPR,  F \t\t\t10\t\t8\t\t7\t\t6\t\t5\t\t5\n\t7
    .Vapor temp,  F \t\t259\t\t232\t\t206\t\t180\t\
    t156\t\t130\n\t8.Vapor pressure, pis/in.Hg\t14.5\
    t\t4\t\t7\t\t6\t\t5\t\t5\n\t9.Lambda, Btu/lb\t\
    t946\t\t962\t\t978\t\t994\t\t1008\t\t1022\n\t10.
    Liqour in, lb/hr\t\t73400\t\t88300\t\t101000\t\
    t113000\t\t72000\t\t72000\n\t11.Liqour out, lb/hr
    \t\t56200\t\t73400\t\t88300\t\t101100\t\t58300\t\
    t54700\n\t12.Evaporation,lb/hr\t\t17200\t\t14900\
    t\t12800\t\t11900\t\t13700\t\t17300\n\t13.Total
    solids, \t\t38.9\t\t29.8\t\t24.7\t\t21.6\t\t18.7\
    t\t20.0\n\t14.A,ft^2\t\t\t3250\t\t3250\t\t3250\t\
    t3250\t\t3250\t\t3250\n\t15.UD,Btu/(hr)*(ft^2)*(
     F )\t262\t\t295\t\t252\t\t251\t\t221\t\t221\n\
    t16.UD delT,Btu/(hr)*(ft^2)\t5510\t\t5000\t\
    t4530t\t\t4770\t\t3980\t\t4650\n");//BPR values
    from fig 14.36a
//Specific-heat data are given in Fig. 14.36b
ev(1)=17200; //lb/hr
ev(2)=14900; //lb/hr
ev(3)=12800; //lb/hr
ev(4)=11900; //lb/hr
ev(5)=13700; //lb/hr
ev(6)=17300; //lb/hr
i=1;
tev =0;
while(i<7)
    tev = tev+ev(i);
    i=i+1;
end
printf("\n\tTotal amount of water evaporated = %.0f
    lb/hr\n",tev);
ttev=tev/6;//lb/hr
printf("\tTheoretical amount of steam for a six-
```

```
effect evaporator = %.0f lb/hr\n",ttev);
tev2=tev/(6*0.75); //lb/hr . order of 75 percent of theoretical
printf("\tSteam used for trail balance = %.0f lb/hr\n",tev2);
lq=(tev/6);
lq=lq+(lq*0.15);
printf("\tEstimate of the amount of evaporation in the first effect = %.0f lb/hr\n",lq);
lout6=54000;//lb/hr
lq2=lout6+lq+2200;//lb/hr
printf("\tEstimated discharge from second effect = %.0f lb/hr\n",lq2);
printf("\n\t\t\tHEAT BALANCE\n");
cw = 17750000/(500*(125-15-60)); //gpm, values from table 14.6
printf("\t\tCooling water at 60  F  = %.0f gpm\n",cw);
printf("\t------------------------------------------------------------\n");
printf("\tEffect\t\t\tBtu/hr\t\tEvaporation,l/hr\n");
printf("\t------------------------------------------------------------\n");
sf=20000;//lb/hr
lqi=73400;//lb/hr
lqi2=88300
lt1=259;// F
lt2=232;// F
lt3=206;// F
ev=17200;//lb/hr
his=sf*924*0.97;//Btu/hr
printf("\t1.a.Heat in steam \t%.2e\n",his);
hl=lqi*(lt1-lt2)*0.82;//Btu/hr
printf("\t  b.Heating liquor \t%.2e\n",hl);
hh=his-hl;
```

```
ev1=(hh)/946;//lb/hr
printf(”\t  c.Evaporation\t\t\t\t%.0f\n”,ev1);
dif=lqi-ev1;
tft=(dif)*(lt1-209)*0.78;
printf(”\t  d.To flash tank\t%.1e”,tft);
ev2=tft/978;//lb/hr
printf(”\t\t%.0f\n”,ev2);
printf(”\t  e.Flashed vapor=%.0f\n”,ev2);
p=dif-ev2;
printf(”\t  f.product %.1e\n”,p);
printf(”\n\t2.a.Heat in 1st vapors\t%.3e\n”,hh);
hl2=lqi2*(lt2-lt3)*0.85;
printf(”\t  b.Heating liqour\t%.2e\n”,hl2);
ev3=(hh-hl2)/962;
printf(”\t  c.Evaporation=%.0f”,ev3);

printf(”\t\t\t%.0f\n”,ev3);
lto1=lqi2-ev3;
printf(”\t  d.Liquor to lb=%.0f\n”,lto1);
//end
```

Scilab code Exa 14.6 Evoparator specification

```
printf(”\texample 14.6\n”);
printf(”\tapproximate values are mentioned in the book \n”);
st1=274; // F
vt6=115; // F
odT=st1-vt6; // F
printf(”\tTotal temperature difference = %.0f  F \n”,odT); //corresponding to 35 psig
eb1=77;// F , From fig.14.38
eb2=26;// F , From fig.14.38
etd=odT-(eb1+eb2);// F
printf(”\tThe effective temperature difference is %
```

```
.0f  F \n”,etd);
printf(”\n\t\t\tCAUSTIC EVAPORATOR MATERIAL BALANCE\n”);
//Basis: 1 ton/hr NaOH
printf(”\tCell liquour at 120 F  \t\tWash at 80 F \n”);
printf(”\t————————————————————————————————————————————\n”);
l1=2000;//Lb
l2=3800;//Lb
l3=17050;//Lb
lq=l1+l2+l3;//Lb
w1=340;//Lb
w2=1020;//Lb
w=w1+w2;//Lb
printf(”\t8.75 prcnt NaOH = %.0f\n\t16.6 prcnt NaCl = %.0f\t\t25 prcnt NaCl = %.0f\n”,l1,l2,w1);
printf(”\t74.65 prcnt H20 = %.0f\t\t75 prcnt H20 = %.0f\n”,l3,w2);
printf(”\tTotal cell liquor = %.0f\tTotoal wash = %.0f\n”,lq,w);
printf(”\n\t
————————————————————————————————————————————————————————————————
n”);
printf(”\t\t\t\tNaOH\t\tNaCl\t\tH20,Lb\tTotal,Lb\n\t\t\t\tprcnt\tLb\tprcnt\tLb\n”);
printf(”\t
————————————————————————————————————————————————————————————————
n”);
printf(”\tOverall operation:\n\t  Cell liquor .......... 8.75\t”+string(l1)+”\t16.60\t”+string(l2)+”\t”+string(l3)+”\t”+string(lq)+”\n”);
printf(”\t  Wash.................  ....\t....\t25.00\t”+string(w1)+”\t”+string(w2)+”\t”+string(w)+”\n”);
wl1=l2+w1;//Lb
wl2=l3+w2;//Lb
```

```
wlt=lq+w;
printf("\t  Total in............. ....\t"+string(l1)
   +"\t....\t"+string(wl1)+"\t"+string(wl2)+"\t"+
   string(wlt)+"\n");
prn=110;//Lb
prh=1890;//Lb
prt=4000;//Lb
printf("\t  Product.............. 50.00\t"+string(l1
   )+"\t2.75\t"+string(prn)+"\t"+string(prh)+"\t"+
   string(prt)+"\n");
r1=wl1-prn;//Lb
r2=wl2-prh;//Lb
r3=wlt-prt;//Lb
gain=3200;//gpm
printf("\t  Removed.............. ....\t....\t....\
   t%.0f\t%.0f\t%.0f\n",r1,r2,r3);
//Rest of the table is calculated similarily
printf("\n\t\t\t\t\tCAUSTIC EVAPORATOR SUMMARY\n");
printf("\t
   ——————————————————————————————————————————————————————————————
   n");
printf("\tItem\t\t\t\t\tEffects\nt\t\t\t\t\t
   ——————————————————————\t\tFlash Tank\n\t\t\t\t\t\I\
   t\tII\n");
printf("\t
   ——————————————————————————————————————————————————————————————
   n");
printf("\t1.Steam pressure, psi/in.Hg\t30\n\t2.Steam
    temperature, F \t\t274\t\t169\n\t3.delT, F \t\t\
   t28\t\t28\n\t4.Liquor temperature,  F \t246\t\
   t141\t\t192\n\t5.BPR,  F \t\t\t77\t\t26\t\t77\n\
   t6.Vapor temperature,  F \t\t169\t\t115\t\t115\n\
   t7.Lambda, Btu/lb\t\t997\t\t1027\t\t1027\n\t8.
   Feed, lb/hr\t\t\t22788\t\t50602\t\t13367\n\t9.
   Product, lb/hr\t\t13367\t\t40352\t\t12813\n\t10.
   Evaporation,lb/hr\t\t9421\t\t10250\t\t554\n\t11.
   Heat flow, Btu/hr\t\t11890000\t11020000\n\t12.UD,
   Btu/((hr)*(ft^2)*( F ))\t700\n\t13.A,ft^2\t\t\
```

```
    t683\t\t683\n\t14.Tubes, OD, in. and BWG\t1,16\t\
    t1,16\n\t15.Tube length, ft\t\t7\t\t7\n\t16.No.
    tubes\t\t\t432\t\t432\n\t17.Circulating pump. gpm
    \t3200 at 20 ft\t3200 at 20ft\t167 at 45 ft\n\t18
    .Apparent efficiency, prcnt\t54\t\t64\n\t18.BHP\t
    \t\t\t38\t\t35\t\t8.2\n\t20.Motor,hp\t\t\t40\t\
    t40\t\t10.0\n");
printf("\t
    ------------------------------------------------------------------------
    n");
V=8;
s=1.5;
G=V*s*62.5*3600;//lb/((hr)*(ft^2))
printf("\tG = V(s*62.5*3600) = %.1e lb/((hr)*(ft^2))
    \n",G);
UD=700;//Btu/((hr)*(ft^2)*( F ))
//Combining with a steam film coefficient of
    approximately 1500
printf("\tUC or UD = %.0f Btu/((hr)*(ft^2)*( F ))\n"
    ,UD);
printf("\n\t
    ------------------------------------------------------------------------
    ");
printf("\n\ttx, F \tw,lb/hr\t\tdelT\tUC\tA,ft^2\tat,
    flow area\tGcalc\t\tUcalc\n\t\t\t\t\t\t\tper pass
    , ft^2\n");
printf("\t
    ------------------------------------------------------------------------
    n");
printf("\t251\t2970000\t\t25.4\t700\t670\t0.87\t\
    t3420000\n\t252\t2480000\t\t25.0\t700\t680\t0.88\
    t\t2820000\n\t252.5\t2290000\t\t24.7\t700\t685\t0
    .89\t\t2570000\t\t700\n\t253\t2120000\t\t24.5\
    t700\t695\t0.90\t\t2520000\n");
printf("\tThee gain per minute is %.0f gpm\n",gain);
printf("\n\t\t\t\tCAUSTIC EVAPORATION HEAT BALANCE\n
    ");
printf("\t\t\t\t(Basis = 1ton/hr NaOH)\n");
```

```
printf("\t
    ------------------------------------------------------------------------
    n");
printf("\t\tEFFECT\t\t\tBtu/hr\t\tEvaopration, lb/hr
    \n");
hi=10500*930*0.974;//Btu/hr
hl=18230*(246-150)*0.83;//Btu/hr
rh=hi-hl;//Btu/hr
hc=300000;//Btu/hr
hv=rh-hc;//Btu/hr
evv=hv/997;//lb/hr
printf("\t1.a.Heat in steam\t\t%.1e\n\t   b.Heating
    liquor\t\t%.2e\n\t   c.Resultant heat\t\t%.2ef\n\t
      d.Heat of concentrate\t\t%.0e\n\t   e.Heat of
    vapors\t\t%.2e\t%.0f\n",hi,hl,rh,hc,hv,evv);
s1=1.35;
G1=V*s1*62.5*3600;//lb/((hr)*(ft^2))
printf("\n\tG = V(s*62.5*3600) = %.2e lb/((hr)*(ft
    ^2))\n",G1);
UD1=700;//Btu/((hr)*(ft^2)*( F ))
//Using thermal characteristics for this solution
printf("\tUD = %.0f Btu/((hr)*(ft^2)*( F ))\n",UD1);
//As for effect I:
printf("\n\t
    ------------------------------------------------------------------------
    ");
printf("\n\ttx, F \tw,lb/hr\t\tdelT\tUC\tA,ft^2\tat,
    flow area\tGcalc\t\tUcalc\n\t\t\t\t\t\t\tper pass
    , ft^2\n");
printf("\t
    ------------------------------------------------------------------------
    n");
printf("\t146\t2400000\t\t25.4\t700\t620\t0.80\t\
    t2790000\t\t700\n\t146.5\t2160000\t\t25.2\t700\
    t683\t0.89\t\t2430000\n");
//end
```

Scilab code Exa 14.7 heat and steam

```
printf(”\texample 14.7\n”);
printf(”\tapproximate values are mentioned in the
   book \n”);
M2=14300;//From fig.14.43 and heat balance above
M1=32200-14300;//From fig.14.43 and heat balance
   above
printf(”\tM1 = %.0f lb\n”,M1);
printf(”\n\t\t\t\tEVAPORATOR SUMMARY\n”);
printf(”\t
   ------------------------------------------------------------------------
   n”);
printf(”\tEffects\t\t\t\t\tStraight triple effect\t\
   t\t\tThermocompression\nt\t\t\t\t\t
   ------------------------------------------------------------------------
   n\t\t\t\t\t1\t\t2\t\t3\t\t1\t\t2\t\t3\n”);
printf(”\t
   ------------------------------------------------------------------------
   n”);
printf(”\tSteam flow, lb/hr\t\t22400\t\t\t\t\t\
   t17900\n\tSteam pressure, psi in.Hg\t20\t\t9\t\t2
   \t\t20\t\t9\t\t2\n\tSteam temp, F \t\t\t258\t\
   t237\t\t217\t\t258\t\t237\t\t217\n\ttdelT, F \t\t
   \t20\t\t18\t\t22\t\t20\t\t18\t\t22\n\tLiquor temp
   , F \t\t\t238\t\t219\t\t195\t\t238\t\t219\t\t195
   \n\tBPR, F \t\t\t\t1\t\t2\t\t3\t\t1\t\t2\t\t3\n\
   tVapor temp, F \t\t\t237\t\t217\t\t192\t\t237\t\
   t215\t\t192\n\tVapor pressure, pis/in.Hg\t9\t\t2\
   t\t10\t\t9\t\t2\t\t10\n\tLambda, Btu/lb\t\t\t954\
   t\t965\t\t983\t\t954\t\t965\t\t983\n\tLiquor in,
   lb/hr\t\t100000\t\t79400\t\t56900\t\t109000\t\
   t70000\t\t52400\n\tLiqour out, lb/hr\t\t79400\t\
   t56900\t\t33300\t\t70000\t\t52400\t\t33300\n\
```

```
   tEvaporation ,lb/hr\t\t20600\t\t22500\t\t23500\t\
   t30000\t\t17600\t\t19100\n\t  Brix(out)\t\t\t\t\t
   \t\t\t\t\t\t\t\t30\n\tCondenser water, gpm\t\t\t\
   t455\t\t\t\t\t\t365\n”);
printf(”\n\t\t\t\tHEAT BALANCE–STRAIGHT TRIPLE
   EFFECT\n\t\t\t\tCondenser water = 455 gpm\n”);
printf(”\t
   ————————————————————————————————————————————————————————————————\
   n”);
printf(”\tEffect\t\t\tBtu/hr\t\tEvaporation ,l/hr\n”)
   ;
printf(”\t
   ————————————————————————————————————————————————————————————————\
   n”);
sf=22400;//lb/hr
lc=100000;//lb/hr
t1=238;// F
t2=230;// F
his=sf*940*0.97;//Btu/hr
hlq=lc*(t1-t2)*0.92;//Btu/hr
hd=his-hlq;//Btu/hr
eva=(hd)/954;//lb/hr
l2d=lc-eva;
printf(”\t1.a.Heat in steam\t%.2e\n\t   b.Heating
   liquor\t%.2e\n\t   c.Evaporation\t\t%.4e/954\t%.0f
   \n\t   d.Liquor to 2d = %.0f”,his,hlq,hd,eva,l2d);
//end
```

Chapter 15

Vaporizers Evoparators and Reboilers

Scilab code Exa 15.1 `Calculation of the Average Specific Volume`

```
printf("\t example 15.1 \n");
printf("\t approximate values are mentioned in the book \n");
ts=250;
T1=400;
T2=300;
w=10000; // lb/hr
W=150000; // lb/hr
l=945.3; // Btu/(lb) , table 7
Q=((w)*(l)); // Btu/hr
printf("\t total heat required for steam is : %.2e Btu/hr \n",Q);
C=0.63; // Btu/(lb)*(F)
Q=((W)*(C)*(T1-T2)); // Btu/hr
printf("\t total heat required for kerosene is : %.2e Btu/hr \n",Q);
delt1=T2-ts; //F
delt2=T1-ts; // F
printf("\t delt1 is : %.0f F \n",delt1);
```

```
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
UD=100;
A=(Q/(UD*LMTD));
printf("\t A : %.2e ft^2 \n",A);
WC=94500; // Btu/F
vl=0.017; // ft^3/lb, from table 7
vv=13.75; // ft^3/lb, from table 7
printf("\t By the law of mixtures \n");
// Assume 80 per cent of the outlet fluid is vapor
v2=(0.8*vv)+(.2*vl);
printf("\t v2 : %.0f ft^3/lb \n",v2);
vav=(WC*(v2-vl)/(UD*A))-((WC*(T2-ts)/(l*w))*(vv-vl))+vl;
printf("\t vav : %.2f ft^3/lb \n",vav);
printf("\t By the approximate method \n");
vav1=(vl+v2)/(2);
printf("\t vav : %.2f ft^3/lb \n",vav1);
row=62.5;
rowac=(1/vav);
s=(rowac/row);
printf("\t actual density : %.3f lb/ft^3 \n",rowac);
printf("\t s : %.4f \n",s);
rowap=(1/vav1);
s=(rowap/row);
printf("\t approximate density : %.3f lb/ft^3 \n",rowac);
printf("\t s : %.4f \n",s);
// end
```

Scilab code Exa 15.2 Vaporizer or Pump through Reboiler with Isothermal Boiling

```
printf("\t example 15.2 \n");
printf("\t approximate values are mentioned in the
```

```
book \n");
t1=108; // inlet cold fluid ,F
t2=235; // outlet cold fluid ,F
Ts=338;
Wp=24700; // lb/hr
Wv=19750; // lb/hr
w=4880; // lb/hr
printf("\t 1.for heat balance \n");
Ht1=162; // enthalpy at t1, Btu/lb, fig 9
Ht2=248; // enthalpy at t2, Btu/lb, fig 9
qp=(Wp*(Ht2-Ht1)); // for preheat
printf("\t total heat required for preheat of butane is : %.2e Btu/hr \n",qp);
Ht3=358; // enthalpy of vapour at t2, Btu/lb, fig 9
qv=Wv*(Ht3-Ht2);
printf("\t total heat required for vapourisation of butane is : %.2e Btu/hr \n",qv);
Q=qp+qv;
printf("\t total heat required for  butane is : %.2e Btu/hr \n",Q);
printf("\t for steam \n");
l=880.6; // Btu/(lb), table 7
Q=((w)*(l)); // Btu/hr
printf("\t total heat required for steam is : %.2e Btu/hr \n",Q);
deltp=158.5; // F, from eq 5.14
deltv=103; // F eq 5.14
Wp1=(qp/deltp);
printf("\t Wp1 is : %.2e lb/hr \n",Wp1);
Wv1=(qv/deltv);
printf("\t Wv1 is : %.2e lb/hr \n",Wv1);
W=(Wp1+Wv1);
printf("\t W is : %.2e lb/hr \n",W);
delt=(Q/W);
printf("\t weighted delt is : % .1f F \n",delt);
Tc=((Ts)+(Ts))/(2); // caloric temperature of hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.1
```

```
f F \n",Tc);
tc=((t1)+(t2))/(2); // caloric temperature of cold fluid,F
printf("\t caloric temperature of cold fluid is : %.1f F \n",tc);
printf("\t hot fluid:inner tube side,steam \n");
Nt=76;
n=2; // number of passes
L=16; //ft
at1=0.594; // flow area,table 10, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu1=0.0363; // at 338F, fig 15,lb/(ft)*(hr)
D=0.0725; // ft
Ret=((D)*(Gt)/mu1); // reynolds number
printf("\t reynolds number is : %.1e \n",Ret);
hio=1500; // condensing steam,Btu/(hr)*(ft^2)*(F)
printf("\t hio is : %.0f Btu/(hr)*(ft^2)*(F) \n",hio);
printf("\t cold fluid:shell side,butane \n");
printf("\t preheating \n");
ID=15.25; // in
C=0.25; // clearance
B=5; // baffle spacing,in
PT=1.25;
as=((ID*C*B)/(144*PT)); // flow area,ft^2
printf("\t flow area is : %.3f ft^2 \n",as);
Gs=(Wp/as); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gs);
mu2=0.278; // at 172F,lb/(ft)*(hr), from fig.14
De=0.0825; // from fig.28,ft
Res=((De)*(Gs)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
```

```
jH=159; // from fig.28
Z=0.12; // Z=k*((c)*(mu1)/k)^(1/3), fig 16
hop=((jH)*(1/De)*(Z)); //using eq.6.15b,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",hop);
Up=((hio)*(hop)/(hio+hop)); // clean overall coefficient ,eq 6.38,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient for preheating : %.0e Btu/(hr)*(ft^2)*(F) \n",Up);
Ap=(qp/(Up*deltp));
printf("\t clean surface required for preheating : %.0f ft^2 \n",Ap);
printf("\t for vapourisation \n");
mu2=0.242; // at 172F,lb/(ft)*(hr), from fig.14
Res=((De)*(Gs)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Res);
jH=170; // from fig.28
Z=0.115; // Z=k*((c)*(mu1)/k)^(1/3), fig 16
hov=((jH)*(1/De)*(Z)); //using eq.6.15b,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",hov);
Uv=((hio)*(hov)/(hio+hov)); // clean overall coefficient ,eq 6.38,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient for vapourisation : %.0f Btu/(hr)*(ft^2)*(F) \n",Uv);
Av=(qv/(Uv*deltv));
printf("\t clean surface required for vapourisation : %.0f ft^2 \n",Av);
Ac=Ap+Av;
printf("\t total clean surface : %.1e ft^2 \n",Ac);
UC=((Up*Ap)+(Uv*Av))/(Ac);
printf("\t weighted clean overall coefficient : %.0f Btu/(hr)*(ft^2)*(F) \n",UC);
A2=0.2618; // actual surface supplied for each tube, ft^2,from table 10
A=(Nt*L*A2); // ft^2
```

```
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
// A total of 170 ft2 are required of which 103 are to be used for vaporization. For the total surface required 318 ft2 will be provided. It can be assumed, then, that the surface provided for vaporization is 193ft^2
// then flux is Q/A=10700, which is with in satisfactory levels.
Rd=((UC-UD)/((UD)*(UC))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for inner pipe \n");
f=0.000165; // friction factor for reynolds number 62000, using fig.26
s=0.00413;
phyt=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt)))/(2); // using eq.7.45,psi
printf("\t delPt is : %.2f psi \n",delPt);
printf("\t allowable delPa is negligible \n");
printf("\t pressure drop  for annulus \n");
printf("\t preheating \n");
f=0.00145; // friction factor for reynolds number 69200, using fig.29
Lp=(L*Ap/Ac); //ft
printf("\t length of preheat zone : %.1f ft \n",Lp);
N=(12*Lp/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",N);
s=0.5; // for reynolds number 69200,using fig.6
Ds=1.27; // fig 28
phys=1;
delPsp=((f*(Gs^2)*(Ds)*(N))/(5.22*(10^10)*(De)*(s)*(phys))); // using eq.7.44,psi
printf("\t delPsp is : %.1f psi \n",delPsp);
printf("\t vapourisation \n");
```

```
f=0.00142;
Lv=9.7; // Lv=L-Lp
Nv=(12*Lv/B); // number of crosses,using eq.7.43
printf("\t number of crosses are : %.0f \n",Nv);
s=0.28;
delPsv=((f*(Gs^2)*(Ds)*(Nv))/(5.22*(10^10)*(De)*(s)*(1))); // using eq 12.47,psi
printf("\t delPsv is : %.1f psi \n",delPsv);
delPS=delPsp+delPsv;
printf("\t delPS is : %.1f psi \n",delPS);
printf("\t allowable delPa is 5 psi \n");
//end
```

Scilab code Exa 15.3 Calculation of a Kettle Reboiler

```
printf("\t example 15.3 \n");
printf("\t approximate values are mentioned in the book \n");
ts=400;
T1=575;
T2=475;
W=28100; // lb/hr
w=34700; // lb/hr
printf("\t 1.for heat balance \n");
HT1=290; // enthalpy at T1, Btu/lb, fig 11
HT2=385; // enthalpy at T2, Btu/lb, fig 11
Q=(W*(HT2-HT1)); // for preheat
printf("\t total heat required for gasoline is : %.2e Btu/hr \n",Q);
c=0.77; // Btu/(lb), table 7
Q=((w)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for gasoil is : %.2e Btu/hr \n",Q);
delt=118; // F eq 5.14
S=((T2-ts)/(T1-ts));
```

```
printf("\t S is : %.3f \n",S);
Kc=0.37; // fig 17
Fc=0.42;
Tc=(T2+(0.42*(T1-T2)));
printf("\t Tc is : %.0f F \n",Tc);
printf("\t hot fluid:inner tube side,gasoil \n");
Nt=68;
n=6; // number of passes
L=12; //ft
at1=0.546; // flow area,table 10, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu1=0.65; // at 517F, fig 14,lb/(ft)*(hr)
D=0.0694; // ft
Ret=((D)*(Gt)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
jH=220; // from fig.24
Z=0.118; // Z=k*((c)*(mu1)/k)^(1/3), fig 16
Hi=((jH)*(1/D)*(Z)); //hi/phyt, Hi=()using eq.6.15d, Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Hi);
Hio=((Hi)*(0.834/1)); //Hio=(hio/phyp), using eq.6.9
printf("\t Correct Hi0 to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",Hio);
// (mu1/muw)^(0.14) is negligible
printf("\t cold fluid:shell side,gasoline \n");
ho=300; // assumption
tw=(ts)+(((Hio)/(Hio+ho))*(Tc-ts)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
deltw=(tw-ts);
printf("\t deltw : %.0f F \n",deltw);
// from fig 15.11, ho>300
Uc=((Hio)*(ho)/(Hio+ho)); // clean overall
```

```
coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
A2=0.2618; // actual surface supplied for each tube,ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
// check for max. flux=Q/A=12500.(satisfactory)
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for inner pipe \n");
f=0.00015; // friction factor for reynolds number 85700, using fig.26
s=0.71;
phyt=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.09; // X1=((V^2)/(2*g)), for Gt 1060000,using fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPa is 10psi \n");
printf("\t delPs is negligible \n");
//end
```

Scilab code Exa 15.4 Calculation of a Once through Horizontal Thermosyphon Reboile

```
printf("\t example 15.4\n");
```

```
printf(”\t approximate values are mentioned in the book \n”);
t1=315; // inlet cold fluid ,F
t2=335; // outlet cold fluid ,F
T1=525;
T2=400;
Wv=29000; // lb/hr
Ws=38500; // lb/hr
w=51000; // lb/hr
printf(”\t 1.for heat balance \n”);
Ht1=238; // enthalpy at t1, Btu/lb, fig 9
Ht2=252; // enthalpy at t2, Btu/lb, fig 9
Ht3=378; // enthalpy of vapour at t2
qv=(Wv*(Ht3-Ht2)); // for preheat
printf(”\t qv is : %.2e Btu/hr \n”,qv);
qs=Ws*(Ht2-Ht1);
printf(”\t qs is : %.2e Btu/hr \n”,qs);
Q=qs+qv;
printf(”\t total heat required for naphtha is : %.2e Btu/hr \n”,Q);
c=0.66; // Btu/(lb)(F)
Q=((w)*(c)*(T1-T2)); // Btu/hr
printf(”\t total heat required for gasoil is : %.2e Btu/hr \n”,Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf(”\t delt1 is : %.0f F \n”,delt1);
printf(”\t delt2 is : %.0f F \n”,delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf(”\t LMTD is :%.0f F \n”,LMTD);
R=((T1-T2)/(t2-t1));
printf(”\t R is : %.2f \n”,R);
S=((t2-t1)/(T1-t1));
printf(”\t S is : %.3f \n”,S);
printf(”\t FT is 0.97 \n”); // from fig 18
delt=(0.97*LMTD); // F
printf(”\t delt is : %.0f F \n”,delt);
X=((delt1)/(delt2)); // fig 17
```

```
printf("\t ratio of two local temperature difference is : %.3f \n",X);
Fc=0.41; // from fig.17
Kc=0.42;
Tc=((T2)+((Fc)*(T1-T2))); // caloric temperature of hot fluid,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+((Fc)*(t2-t1))); // caloric temperature of cold fluid,F
printf("\t caloric temperature of cold fluid is : %.0f F \n",tc);
printf("\t hot fluid:inner tube side,steam \n");
Nt=116;
n=8; // number of passes
L=12; //ft
at1=0.546; // flow area,table 10, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu1=1.09; // at 451F, fig 14,lb/(ft)*(hr)
D=0.0695; // ft
Ret=((D)*(Gt)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
jH=168; // from fig.24
Z=0.142; // Z=k*((c)*(mu1)/k)^(1/3), fig 16
Hi=((jH)*(1/D)*(Z)); //, Hi=(hi/phyt)using eq.6.15d,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Hi);
Hio=((Hi)*(0.834/1)); //Hio=(hio/phyp), using eq.6.9
printf("\t Correct Hio to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",Hio);
printf("\t cold fluid:shell side,naphtha \n");
ho1=200; // assumption
```

```
tw=(tc)+(((Hio)/(Hio+ho1))*(Tc-tc)); // from eq.5.31, calculation mistake
printf("\t tw is : %.0f F \n",tw);
deltw=(tw-tc);
printf("\t deltw : %.0f F \n",deltw);
// from fig 15.11, hv>300, hs=60
Av=(qv/300);
As=qs/60;
printf("\t qv/hv : %.3e \n",Av);
printf("\t qs/hs : %.0e \n",As);
A1=As+Av;
printf("\t A : %.3e \n",A1);
ho=(Q/A1);
printf("\t ho : %.0f \n",ho);
Uc=((Hio)*(ho)/(Hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
A2=0.2618; // actual surface supplied for each tube,ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UD=((Q)/((A)*(delt)));
printf("\t actual design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
// check for max. flux=Q/A=11500.(satisfactory)
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t pressure drop  for inner pipe \n");
f=0.000168; // friction factor for reynolds number 59200, using fig.26
s=0.73;
phyt=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
X1=0.11; // X1=((V^2)/(2*g)), for Gt 1060000,using
```

```
    fig.27
delPr=((4*n*X1)/(s)); // using eq.7.46,psi
printf("\t delPr is : %.1f psi \n",delPr);
delPT=delPt+delPr; // using eq.7.47,psi
printf("\t delPT is : %.1f psi \n",delPT);
printf("\t allowable delPa is negligible \n");
printf("\t pressure drop  for annulus \n");
Af=(3.14*(21.25^2-(116))/8);
printf("\t flow area : %.0f in^2 \n",Af);
as=0.917; // ft^2
p=(3.14*21.25/2)+(3.14*1*116/2)+(21.25);
printf("\t wetted perimeter : %.1f in \n",p);
De=0.186; // ft
Gs=(Ws/(2*as)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.1e lb/(hr)*(ft^2) \n
    ",Gs);
mu2=0.435; // at 315F, fig 14,lb/(ft)*(hr)
Res=((De)*(Gs)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
f=0.00028; // using fig.26
row=0.337; // fig 13.14
// soutlet max=0.071,
s=0.35; // using fig.6
phys=1;
delPs=((f*(Gs^2)*(L))/(5.22*(10^10)*(De)*(s)*(phys))
    ); // using eq.7.44,psi
printf("\t delPs is : %.4f psi \n",delPs);
printf("\t allowable delPa is .25 psi \n");
//end
```

Scilab code Exa 15.5 Calculation of a Vertical Thennosyphon Reboiler

```
printf("\t example 15.5\n");
printf("\t approximate values are mentioned in the
    book \n");
```

```
W=40800; // lb/hr
w=4570; // lb/hr
printf(”\t 1.for heat balance \n”);
Ht1=241; // enthalpy of liquid at 228F, Btu/lb, fig 9
Ht2=338; // enthalpy of vapourat 228F, Btu/lb, fig 9
Q=(W*(Ht2-Ht1));
printf(”\t total heat required for butane is : %.2e Btu/hr \n”,Q);
l=868; // Btu/(lb), table 7
Q=((w)*(l)); // Btu/hr
printf(”\t total heat required for steam is : %.2e Btu/hr \n”,Q);
delt=125; // delt=LMTD, isothermal boiling, eq 5.14
// Tc and tc: Both streams are isuthermal
printf(”\t trail 1 \n”);
A1=((Q)/((12000))); // Q/A1 =12000, first trial should always be taken for the maximum allowable flux
printf(”\t A1 is : %.1e ft^2 \n”,A1);
a1=0.1963; // ft^2/lin ft
L=16;
N1=(A1/(L*a1)); // table 10
printf(”\t number of tubes are : %.0f \n”,N1);
N2=109; // assuming one tube passes, 13.25-in ID, from table 9
A2=(N2*L*a1); // ft^2
printf(”\t total surface area is : %.0f ft^2 \n”,A2);
UD=((Q)/((A2)*(delt)));
printf(”\t correct design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n”,UD);
// Assume 4: 1 recirculation ratio
rowv=(58/(359*(688/492)*(14.7/290))); // eq 15.18
printf(”\t vapour density : %.2f lb/ft^3 \n”,rowv);
Vv=0.44;
Vl=0.0372; // fig 6
W1=4*W;
```

```
printf("\t weight flow of recirculated liquid : %.3e lb/hr \n",W1);
VL=W1*Vl;
VV=W*Vv;
printf("\t volume of liquid : %.2e ft^3 \n",VL);
printf("\t volume of vapour : %.3e ft^3 \n",VV);
V=VL+VV;
printf("\t total volume out of reboiler : %.3e ft^3 \n",V);
vo=(V/(W1+W));
printf("\t vo is : %.4f ft^3/lb \n",vo);
Pl=((2.3*16)/(144*(vo-Vl)))*(log10(vo/Vl));
printf("\t pressure leg : %.1f psi \n",Pl);
printf("\t frictional resistance \n");
Nt=109;
n=1; // number of passes
at1=0.302; // flow area,table 10, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=((W1+W)/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu1=0.242; // at 228F, fig 14,lb/(ft)*(hr)
D=0.0517; // ft
Ret=((D)*(Gt)/mu1); // reynolds number
printf("\t reynolds number is : %.1e \n",Ret);
f=0.000127; // using fig.26
s=0.285;
phyt=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.2f psi \n",delPt);
P=Pl+delPt;
printf("\t total resisitance : %.2f psi \n",P);
F=(16*0.43*62.5/144);
printf("\t driving force : %.2f psi \n",F);
// The resistances are greater than the hydrostatic
```

```
   head can provide; hence the recirculation ratio will be less than 4: 1
printf("\t trial 2 \n"); // Assume 12'0" tubes and 4:1 recirculation ratio
A1=((Q)/((12000))); // Q/A1 =12000, first trial should always be taken for the maximum allowable flux
printf("\t A1 is : %.1e ft^2 \n",A1);
a1=0.1963; // ft^2/lin ft
L=12;
N1=(A1/(L*a1)); // table 10
printf("\t number of tubes are : %.0f \n",N1);
N2=151; // assuming one tube passes, 15.25-in ID, from table 9
A2=(N2*L*a1); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A2);
UD=((Q)/((A2)*(delt)));
printf("\t correct design overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
Pl=((2.3*12)/(144*(vo-Vl)))*(log10(vo/Vl));
printf("\t pressure leg : %.1f psi \n",Pl);
printf("\t frictional resistance \n");
Nt=151;
n=1; // number of passes
at1=0.302; // flow area,table 10, in^2
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf("\t flow area is : %.3f ft^2 \n",at);
Gt=((W1+W)/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu1=0.242; // at 228F, fig 14,lb/(ft)*(hr)
D=0.0517; // ft
Ret=((D)*(Gt)/mu1); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
f=0.000135; // using fig.26
s=0.285;
```

```
phyt=1;
delPt=((f*(Gt^2)*(12)*(n))/(5.22*(10^10)*(D)*(s)*(phyt))); // using eq.7.45,psi
printf("\t delPt is : %.2f psi \n",delPt);
P=Pl+delPt;
printf("\t total resisitance : %.2f psi \n",P);
F=(12*0.43*62.5/144);
printf("\t driving force : %.2f psi \n",F);
// Since the driving force is slightly greater than the resistances, a recirculation ratio better than 4:1 is assured.
printf("\t hot fluid : shell side,steam \n");
ho=1500; // condensing steam
printf("\t cold fluid:inner tube side, butane \n");
jH=330; // from fig.24
Z=0.115; // Z=k*((c)*(mu1)/k)^(1/3), fig 16
Hi=((jH)*(1/D)*(Z)); //, Hi=(hi/phyt)using eq.6.15d,Btu/(hr)*(ft^2)*(F)
printf("\t individual heat transfer coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Hi);
Hio=((300)*(0.62/0.75)); //Hio=(hio/phyp), using eq.6.9
printf("\t Correct Hio to the surface at the OD is : %.0f Btu/(hr)*(ft^2)*(F) \n",Hio);
Uc=((Hio)*(ho)/(Hio+ho)); // clean overall coefficient,Btu/(hr)*(ft^2)*(F)
printf("\t clean overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
UD=89;
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
// end
```

Scilab code Exa 15.6 Calculation of the Reboiler Duty

```
printf(”\t example 15.6\n”);
printf(”\t approximate values are mentioned in the book \n”);
//20000=WD+WB;
//0.99*WD+(0.05*WB)=(20000*.5);
// solving above two eq. we get WD and WB
WD=9570; // lb/hr
WB=10430; // lb/hr
HBl=108; // fig 3 and 12
HDl=85.8; //fig 3 and 12
HDv=253.8; // fig 3 and 12
HFl=92; // fig 3 and 12
l=153; // fig 3 and 12
QR=((2.54+1)*WD*(HDv))-(2.54*WD*HDl)+(WB*HBl)-(20000*HFl);
printf(”\t total heat duty : %.1e Btu/hr \n”,QR);
Q=QR/153;
printf(”\t total heat duty : %.2e lb/hr \n”,Q);
// end
```

Scilab code Exa 15.7 `Distillation of a Binary Mixture`

```
printf(”\t example 15.7 \n”);
printf(”\t approximate values are mentioned in the book \n”);

//Basis: One hour
//20000=WD+WB , material balance
//0.99*WD+(0.05*WB)=(20000*0.5) , Benzene balance
// solving above two eq. we get WD and WB
WD=9570; // lb/hr
WB=10430; // lb/hr

//Compositions and Boiling Points
//Feed
```

```
l1 = 10000; //Lb/hr , C6H4
l2 = 10000; //Lb/hr , C7H8
lb = l1+l2; //Lb/hr
printf(”\ttotal Lb/hr is %.0f\n”,lb);
mo1 = 78.1; //Mol. wt., C6H6
mo2 = 93.1; //Mol. wt , C7H8
mh1 = 128.0; //Mol/hr , C6H6
mh2 = 107.5; //Mol/hr , C7H8
mh = mh1 + mh2; // Mol/hr
printf(”\ttotal Mol/hr is %.1f\n”,mh);
x1 = mh1/mh;
printf(”\tx1 of C6H6 is %.3f\n”,x1);
x2 = mh2/mh;
printf(”\tx1 of C7H8 is %.3f\n”,x2);
x = x1+x2;
printf(”\tTotal x1 is %.3f\n”,x);
Pp1= 1380; // 214 F
Pp2=575; // 214 F
xp1 = x1*Pp1;
printf(”\tx1Pp1 of C6H6 is %.0f\n”,xp1);
xp2 = x2*Pp2;
printf(”\tx1Pp1 of C7H8 is %.0f\n”,xp2);
sxp = xp1 + xp2;
printf(”\tTotal x1Pp1 is %.0f\n”,sxp);
y1 = xp1/sxp;
printf(”\ty1 of C6H6 is %.3f\n”,y1);
y2 = xp2/sxp;
printf(”\ty1 of C7H8 is %.3f\n”,y2);
y = y1+y2;
printf(”\tTotal y1 is %.3f\n”,y);


w1 = 0.558; //from eq 15.42
printf(”\t(WR‘/V =((xD − yF)/.(xD − xF))) = %.3fmol/mol\n”,w1);
wD=1;
xD = 0.992;
//V = WR’ + WD
```

```
// WR'/V = 0.558
//Solving, WR' = (WR' * 0.558) + (0.558 * WD)
Wr = 1.27; // mol reflux/mol distillate
printf("\tWR` = %.2f (mol reflux)/(mol distillate)\n",Wr);
Wr1 = Wr * 2; // mol/ mol distillate
printf("\tAssumed 200 percent of the theoretical minimum reflux as economic\n\tWR = %.2f(mol)/(mil distillate)\n",Wr1);
in = (wD * xD)/(Wr1 + 1); //intercept for the upper operating line
printf("\tThe intercept for the upper operating line = %.3f\n",in);
p = 13; // From fig. 15.23, connecting the corresponding lines
printf("\tConnecting the corresponding line in Fig. 15.23, plates required: %.0f\n",p);
fp = 7; // From fig. 15.23, connecting the corresponding lines
printf("\tFeed plate is %.0fth(from top)\n",fp);
d=122.5;
tf = Wr1 * d;
printf("\tTotal reflux is %.1f\n",tf);
printf("\t\t\t\t\tHeat balances");

//Heat Balances
l1 = 33900;
l2 = 9570;
l3 = 24330;
b1 = 253.8;
b2 = 85.8;
b3 = 85.8;
bt1 = b1*l1;
bt2 = b2*l2;
bt3 = b3*l3;
bt4 = 5688000;
printf("\n\t\t\t\tMol/hr\tMol.wt.\tLb/hr\tTemp, F \tBtu/lb\tBtu/hr\n\
```

```
t-----------------------------------------------------------------\n\tHeat balance \n\taround condenser:\n");
printf("\t  Heat in:\n\t  Top plate vapor.......433\t87.3\t%.0f\t195\t%.1f\t%.0f\n",l1,b1,bt1);
printf("\t  Heat out:\n\t  Distillate............");
printf("122.5\t78.3\t%.0f\t195\t%.1f\t%.0f\n",l2,b2,bt2);
printf("\t  Reflux.................");
printf("310.5\t78.3\t%.0f\t195\t%.1f\t%.0f\n",l3,b3,bt3);
printf("\t  Condenser duty, by\n\t  difference ........... .....   ....    ......  ..");
printf(".     .....   5688000\n");
printf("\t\t\t\t\t\t\t\t\t_______\n\t\t\t\t\t\t\t\t\t8600000\n\n");

lam = 153; // At 246  F , Btu/hr
rv = 5800000/153; //Lb/hr
printf("\tReboiler vapor is %.2e lb/hr\n",rv);
to = rv + 10430; //Lb/hr
printf("\tTrapout is %.3e lb/hr\n",to);

printf("\n\t\t\t\tMol/hr\tMol.wt.\tLb/hr\tTemp, F \tBtu/lb\tBtu/hr\n\t-----------------------------------------------------------------\n");
printf("\tHeat in:\n\t  Trapout...............522\t92.8\t%.0f\t246\t108.0\t5230000\n",to);
printf("\t  Reboiler duty, \n\t  by difference ....... ....   ....   .....   ...    ..... 5800000\n");
printf("\t\t\t\t\t\t\t\t\t_______\n\t\t\t\t\t\t\t\t\t11030000\n\n");
printf("\n\tReboiler requirements are\n");
printf("\t\tTotal liquid to reboiler\t48330 lb/hr\n\t\tVaporization\t\t\t37900 lb/hr\n\t\tTemperature (nearly isothermal)\t246 F\n\t\tPressure\t\t\t5 psig\n\t\tHeat load\t\t\t5800000 Btu/hr\n")
```

```
//end
```

Scilab code Exa 15.8 The Reboiler Duty for a Multicomponent Mixture

```
printf("\t example 15.8 \n");
printf("\t approximate values are mentioned in the book \n");
//Dew point of Overhead
vc(1) = 6.4; // Mol/hr
vc(2) = 219.7; //Mol/hr
vc(3) = 2.3; //Mol/hr

K(1) = 2.8; //at 148 F  and 40 psia
K(2) = 1.01; //at 148 F  and 40 psia
K(3) = 0.34; //at 148 F  and 40 psia

i=1;
while(i<4)
    v(i)=vc(i)/K(i);
    i=i+1;
end

printf("\n\t\tDEW POINT OF OVERHEAD");
printf("\n\t\tMol/hr\t\tK(148 F ,40 psia)\tV/K\n");
printf("\t\t--------------------------------------------------------\n");
i=1;
while(i<4)
    printf("\tC"+string(i+3) + "\t%.1f\t\t%.1f\t\t\t%.1f\n",vc(i),K(i),v(i));
    i = i+1
end


bc(1)=4.1; //Mol/hr
```

```
bc(2)=49.3; //Mol/hr
bc(3)=71.9; //Mol/hr
bc(4)=52.5; //Mol/hr
bc(5)=54.7; //Mol/hr
bc(6)=82.5; //Mol/hr
bc(7)=76.6; //Mol/hr
bc(8)=22.4; //Mol/hr
tbc = 0;
i=1;
while(i<9)
    tbc = tbc+bc(i);
    i=i+1;
end

bK(1)=5.8; //at 330 F , 40 psia
bK(2)=3.0; //at 330 F , 40 psia
bK(3)=1.68; //at 330 F , 40 psia
bK(4)=0.98; //at 330 F , 40 psia
bK(5)=0.57; //at 330 F , 40 psia
bK(6)=0.35; //at 330 F , 40 psia
bK(7)=0.21; //at 330 F , 40 psia
bK(8)=0.13; //at 330 F , 40 psia

KL(1)=23.8;
KL(2)=148.0;
KL(3)=120.8;
KL(4)=51.4;
KL(5)=31.2;
KL(6)=28.9;
KL(7)=16.1;
KL(8)=2.9;
tk =0;
i=1;
while(i<9)
    tk = tk + KL(i);
    i=i+1;
end

```

```
l(1)=1700; //Lb/hr
l(2)=13900; //Lb/hr
l(3)=13030; //Lb/hr
l(4)=6260; //Lb/hr
l(5)=4240; //Lb/hr
l(6)=4330; //Lb/hr
l(7)=2640; //Lb/hr
l(8)=520; //Lb/hr

tl=0;
i=1;
while(i<9)
    tl = tl+l(i);
    i=i+1;
end

printf("\n\t\tBUBBLE POINTS OF BOTTOMS\n");
printf("\t\tMol/hr\t\tK(330 F ,40psia)\t\tKL\t\tLb/hr\n");
printf("\t\t---------------------------------------------------------------\n");
i=1;
while(i<9)
    printf("\tC"+string(i+4)+"\t%.1f\t\t%.2f\t\t\t%.1f\t\t%.0f\n",bc(i),bK(i),KL(i),l(i));
    i=i+1;
end
printf("\t\t____\t\t\t\t\t____\t\t____\n");
printf("\t\t%.1f\t\t\t\t\t%.1f\t\t%.0f\n",tbc,tk,tl);
av = tl/tk;
printf("\tAverage mol. wt. %.1f\n",av);

lh(1)=48894;//Lb/hr
lh(2)=16298;//Lb/hr
lh(3)=32596;//Lb/hr
bl(1)=286;//Btu/hr
```

```
bl(2)=129;//Btu/hr
bl(3)=129;//Btu/hr
i=1;
while(i<4)
    bh(i)=lh(i)*bl(i); //Btu/hr
    i=i+1;
end

//Heat Balances
printf("\n\n\t\t\t\t\t\tHEAT BALANCES:");
printf("\n\t\t\t\tMol/hr\t\tMol.wt.\t\tLb/hr\t\tTemp, F \t\tBtu/lb\t\tBtu/hr\n\t");
printf("\t\t\t--------------------------------------------------------------------------------");
printf("\n\tHeat Balance onCondeser\n\t  Heat in:\n\t   Top plate vapor......");
printf("685.2\t\t71.3\t\t" + string(lh(1)) + "\t\t148\t\t" +string(bl(1)) + "\t\t" + string(bh(1)) + "\n");
printf("\t  Heat out:\n\t   Distillate...........");
printf("228.4\t\t71.3\t\t" + string(lh(2)) + "\t\t124\t\t" +string(bl(2)) + "\t\t" + string(bh(2)) + "\n");
printf("\t   Reflux, (2-1)........");
printf("456.8\t\t71.3\t\t" + string(lh(3)) + "\t\t129\t\t" +string(bl(3)) + "\t\t" + string(bh(3)) + "\n");
printf("\t   Condenser duty, by\n\t    difference......... ");
printf(".....\t\t.....\t\t.....\t\t.....\t\t......\t\t7680000\n");
printf("\t\t\t\t\t\t\t\t\t\t\t\t\t\t________\n");
printf("\t\t\t\t\t\t\t\t\t\t\t\t\t\t"+string(bh(1))+"\n");
//Heat Balances on reboiler
//Assume 30  difference between reboiler and bottom plate giving bottom-plate temperature of 300 F
```

```
//Mol/hr from Eq. 15.47
rl(1)=78177;//Lb/hr
rl(2)=22700;//Lb/hr
rl(3)=55477;//Lb/hr
rb(1)=234;//Btu/lb
rb(2)=369;//Btu/lb
rb(3)=256;//Btu/lb

i=1;
while(i<4)
    rr(i)=rl(i)*rb(i);//Btu/hr
    i=i+1;
end
tt = rr(1)+4280000;// Btu/hr
printf("\t\t\t\t\t\tHEAT BALANCES on reboiler:");
printf("\n\tHeat in:\n\t  Trapout...............");
printf("619.7\t\t126.6\t\t"+string(rl(1))+"\t\t300\t\t"+string(rb(1))+"\t\t%.2e\n",rr(1));
printf("\t  Reboiler duty........");
printf(" .....\t\t.....\t\t......\t\t...\t\t...\t\t4280000\n");
printf("\t\t\t\t\t\t\t\t\t\t\t\t\t\t________\n");
printf("\t\t\t\t\t\t\t\t\t\t\t\t\t\t%.3e",tt);
printf("\n\tHeat out:\n\t  Reboiler vapor........");
printf("205.7\t\t110.3\t\t"+string(rl(2))+"\t\t330\t\t"+string(rb(2))+"\t\t%.2e\n",rr(2));
printf("\t  Reboiler vapor........");
printf("414.0\t\t134.0\t\t"+string(rl(3))+"\t\t330\t\t"+string(rb(3))+"\t\t%.2e\n",rr(3));
printf("\t\t\t\t\t\t\t\t\t\t\t\t\t\t________\n");
printf("\t\t\t\t\t\t\t\t\t\t\t\t\t\t%.3e",rr(2)+rr(3));

//y*
pc(1)=0.056;
pc(2)=0.350;
pc(3)=0.285;
pc(4)=0.122;
```

```
pc(5)=0.074;
pc(6)=0.068;
pc(7)=0.038;
pc(8)=0.007;

//K(300  F  ,40psia)
pK(1)=4.5;
pK(2)=2.25;
pK(3)=1.20;
pK(4)=0.66;
pK(5)=0.38;
pK(6)=0.22;
pK(7)=0.13;
pK(8)=0.07;

printf("\n\n\t\tCALCULATION OF BOTTOM PLATE
   TEMPERATURE\n");
printf("\t\ty*\t\t\tReboiler vapor\t\t\t\tK(300  F
   ,40psia)\tMol*K\n\t\t\t\tV = y*205.7 +\tBottoms\t
   =\tTrapout\n");
printf("\t\t
   ----------------------------------------------------------------------------------------
   n");

i=1;
pcs=0;
pc2=0;
bcs=0;
tcs=0;
gg=0;
while(i<9)
    temp = pc(i)*205.7;
    temp2 = temp + bc(i);
    printf("\tC"+ string(i+4)+ "\t" +string(pc(i))+
       "\t\t%.1f\t\t\t" + string(bc(i))+"\t\t%.1f\t\t\t"
       +string(pK(i))+"\t\t%.2f\n",temp,temp2,temp2*
       pK(i));

```

```
    pcs=pcs+pc(i);
    pc2=pc2+temp;
    bcs=bcs+bc(i);
    tcs=tcs+temp2;
    gg=gg+(temp2*pK(i));
    i=i+1;
end
printf("\t\t------------------------------------------------------------------------------------------\n");
printf("\t\t%.3f\t\t%.1f\t\t%.1f\t\t%.1f\t\t\t\t%.1f\n",pcs,pc2,bcs,tcs,gg);
printf("\n\tReboiler requirements are\n");
printf("\t\tVaporization\t\t\t22700 lb/hr\n\t\tTotal liquor to reboiler\t78177 lb/hr\n\t\tHeat load\t\t\t4280000 Btu/hr\n\t\tTemperature range\t\t300-330 F \n\t\tOperating pressure\t\t40psia")
//end
```

Chapter 16

Extended Surfaces

Scilab code Exa 16.1 `Calculation of the Fin Efficiency and a Weighted Efficiency C`

```
printf("\t example 16.1 \n");
printf("\t approximate values are mentioned in the book \n");
Af=(20*0.75*12*2)/(144);
Ao=((3.14*1.25)-(20*0.035))*(12/144);
printf("\t fin surface is : %.1f ft^2/lin ft \n",Af);
printf("\t bare tube surface is : %.3f ft^2/lin ft \n",Ao);
A=(Af+Ao);
printf("\t total outside surface : %.2f ft^2/lin ft \n",A);
Ai=(3.14*1.06*12)/(144);
printf("\t total inside surface : %.3f ft^2/lin ft \n",Ai);
printf("\t fin efficiencies \n");
b=0.0625; // ft
hf=4; // from table in solution
m=(5.24*(hf^(1/2))); // m=((hf*P)/(Kax))^(1/2), eq 16.8
n=(tanh(m*b))/(m*b); // efficiency , eq 16.26
```

```
printf("\n hf       m       n \n "+string(hf)+"       "+string(m)+"       "+string(n)+" \n");
// similarly efficiencies values are calculated at different hf values
printf("\t weighted efficiency curve \n");
hfi=((n*Af)+(Ao))*(hf/Ai); // eq 16.34
printf("\n hf       hfi \n "+string(hf)+"       "+string(hfi)+" \n");
// similarly efficiencies values are calculated at different hf values
hf=[4 16 36 100 400 625 900]; // from 2nd table in the solution
hfi=[35.4 110.8 193.5 370 935 1295 1700]; // from 2 nd table in the solution
plot2d("oll",hf,hfi);
xtitle("weighted fin efficiency curve","heat transfer coefficient to fin,Btu/(ft^2)*(hr)","coefficient hf referred to the tube ID");
//end
```

Scilab code Exa 16.2 `Calculation of a Heat transfer Curve from Experimental Data`

```
printf("\t example 16.2 \n");
printf("\t approximate values are mentioned in the book \n");
Ts=302; // F
t1=151;
t2=185;
w=15200; // lb/hr
// The dropwise condensation of steam was promoted with oil.
aa=(3.14*(3.068^2-1.25^2))/(4*144)-((20*0.035*0.75)/(144));
printf("\t annulus flow area : %.4f ft^2 \n",aa);
p=(3.14*(1.25/12))-(20*0.035/12)+(20*0.75*2/12);
```

```
printf("\t wetted perimeter : %.2f ft \n",p);
De=(4*aa/p);
printf("\t equivalent diameter : %.3f ft \n",De);
Q=w*0.523*(t2-t1);
printf("\t heat load : %.2e Btu/hr \n",Q);
delt1=Ts-t1; //F
delt2=Ts-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
Ai=0.277; // ft^2/ft
n=20; // number of fins
Ui=(Q/(Ai*n*LMTD));
printf("\t Ui : %.0f Btu/(hr)*(ft^2)*(F) \n",Ui);
hi=3000; // assumed value for dropwise condensation of steam
hfi=(Ui*hi)/(hi-Ui);
printf("\t hfi : %.0f Btu/(hr)*(ft^2)*(F) \n",hfi);
hf=120; // from fig 16.7 for hfi=418
mu=1.94; // lb/(ft*hr)
k=0.079;
Z=2.34; // Z=((c*mu)/k)^(1/3)
jf=(hf*De/(Z*k)); // eq 16.36
printf("\t jf : %.0f \n",jf);
Ga=(w/aa);
printf("\t Ga : %.2e lb/(hr)*(ft^2) \n",Ga);
Rea=(De*Ga/mu);
printf("\t Rea : %.2e \n",Rea);
// end
```

Scilab code Exa 16.3 Calculation of a Double Pipe Extended surface Gas Oil Cooler

```
printf("\t example 16.3 \n");
printf("\t approximate values are mentioned in the
```

```
   book \n");
T1=250; // inlet hot fluid ,F
T2=200; // outlet hot fluid ,F
t1=80; // inlet cold fluid ,F
t2=120; // outlet cold fluid ,F
W=18000; // lb/hr
w=11950; // lb/hr
printf("\t 1.for heat balance \n")
C=0.53; // Btu/(lb)*(F)
Q=((W)*(C)*(T1-T2)); // Btu/hr
printf("\t total heat required for gas oil is : %.2e
    Btu/hr \n",Q);
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for water is : %.2e
   Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
X=((delt1)/(delt2));
printf("\t ratio of two local temperature difference
    is : %.2f \n",X);
Fc=0.47; // from fig.17
Kc=0.27;
Tc=((T2)+((Fc)*(T1-T2))); // caloric temperature of
   hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.0
   f F \n",Tc);
tc=((t1)+((Fc)*(t2-t1))); // caloric temperature of
   cold fluid ,F
printf("\t caloric temperature of cold fluid is : %
   .0f F \n",tc);
printf("\t hot fluid:shell side,gas oil \n");
ID=3.068; // in, table 11
OD=1.9; // in, table 11
```

```
af=0.0175; // fin cross section,table 10
aa=((3.14*ID^2/(4))-(3.14*OD^2/(4))-(24*af))/(144);
printf(”\t flow area is : %.4f ft^2 \n”,aa);
p=(3.14*(OD))-(24*0.035)+(24*0.5*2);
printf(”\t wetted perimeter : %.2f in \n”,p);
De=(4*aa*12/(p));
printf(”\t De : %.4f ft \n”,De);
Ga=(W/aa); // mass velocity,lb/(hr)*(ft^2)
printf(”\t mass velocity is : %.2e lb/(hr)*(ft^2) \n”,Ga);
mu1=2.5*2.42; // at 224F,lb/(ft)*(hr), from fig.14
Rea=((De)*(Ga)/mu1); // reynolds number
printf(”\t reynolds number is : %.2e \n”,Rea);
jf=18.4; // from fig.16.10
Z=0.25; // Z=k*((c)*(mu1)/k)^(1/3), fig 16
Hf=((jf)*(1/De)*(Z)); // Hf=(hf/phya),using eq.6.15, Btu/(hr)*(ft^2)*(F)
printf(”\t individual heat transfer coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n”,Hf);
printf(”\t cold fluid:inner tube side,water \n”);
D=0.134; // ft
row=62.5;
at=(3.14*D^2/(4));
printf(”\t flow area is : %.4f ft^2 \n”,at);
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf(”\t mass velocity is : %.2e lb/(hr)*(ft^2) \n”,Gt);
V=(Gt/(3600*row));
printf(”\t V is : %.2f fps \n”,V);
mu2=0.72*2.42; // at 99F,lb/(ft)*(hr)
Ret=((D)*(Gt)/mu2); // reynolds number
printf(”\t reynolds number is : %.1e \n”,Ret);
hi=(970*0.82); // fig 25
printf(”\t hi : %.0f Btu/(hr)*(ft^2)*(F) \n”,hi);
printf(”\t calculation of tfw \n”);
// Tc-tfw=40F assumption from fig 14
tfw=184;
mufw=3.5; // cp, at 184F
```

```
phya=(2.5/mufw)^0.14;
printf(”\t phya is : %.2f \n”,phya); // from fig.24
hf=(Hf)*(phya); // from eq.6.36
printf(”\t Correct hf to the surface at the OD is : %.1f Btu/(hr)∗(ft^2)∗(F) \n”,hf);
Rdo=0.002;
Rf=(1/hf);
printf(”\t Rf : %.4f \n”,Rf);
hf1=(1/(Rdo+Rf)); // eq 16.37
printf(”\t hf1 : %.1f \n”,hf1);
hfi1=255; // fig 16.9
hfi2=(hf1*5.76); // eq 16.38 and fig 16.9,((Af+Ao)/(Ai))=5.76 from previous prblm
printf(”\t hfi2 : %.0f \n”,hfi2);
Rmetal=(hfi2-hfi1)/(hfi2*hfi1); // eq 16.39
printf(”\t Rmetal : %.5f \n”,Rmetal);
phyt=1; // for cooling water
Rdi=0.003;
Ri=(1/hi);
printf(”\t Ri : %.5f \n”,Ri);
hi1=(1/(Rdi+Ri)); // eq 16.40
printf(”\t hi1 : %.1f \n”,hi1);
UDi=(hi1*hfi1)/(hi1+hfi1); // eq 16.41
printf(”\t UDi : %.0f \n”,UDi);
// To obtain the true flux the heat load must be divided by the actual heat−transfer surface.For a 1}2−in. IPS pipe there are 0.422 ft2/lin foot, from table 11
// trial
Ai=(Q/(UDi*LMTD)); // LMTD=delt
printf(”\t Ai : %.1f ft^2 \n”,Ai);
L=(Ai/0.422);
printf(”\t length of pipe required : %.1f lin ft \n”,L);
// Use two 20−ft hairpins = 80 lin ft
Ai1=(80*0.422); // ft^2
r=(Q/Ai1);
printf(”\t Q/Ai1 : %.2e Btu/(hr)∗(ft^2) \n”,r);
```

```
deltf=(r/hfi2);
deltdo=(r*Rdo/5.76);
printf("\t annulus film : %.1f \n",deltf);
printf("\t annulus dirt : %.1f \n",deltdo);
d=deltf+deltdo; // d=Tc-tfw
deltmetal=(r*Rmetal);
deltdi=(r*Rdi);
delti=(r/hi);
printf("\t Tc-tfw : %.1f \n",d);
printf("\t fin and tube metal : %.1f \n",deltmetal);
printf("\t tube side dirt : %.1f \n",deltdi);
printf("\t tubeside film : %.1f \n",delti);
Td=deltf+deltdo+deltmetal+deltdi+delti;
printf("\t total temperature drop : %.1f F \n",Td);
printf("\t pressure drop  for annulus \n");
De1=0.0359; // ft
Rea1=(De1*Ga/mu1);
printf("\t reynolds number : %.2e \n",Rea1);
f=0.00036; // fig 16.10
s=0.82; //using fig.6
delPs=((f*(Ga^2)*(80))/(5.22*(10^10)*(De1)*(s)*(phya))); // using eq.7.44,psi
printf("\t delPs is : %.1f psi \n",delPs);
printf("\t allowable delPa is 10 psi \n");
printf("\t pressure drop  for inner pipe \n");
f=0.000192; // friction factor for reynolds number 65000, using fig.26
s=1;
delPt=((f*(Gt^2)*(80))/(5.22*(10^10)*(0.134)*(s)*(1))); // using eq.7.45,psi
printf("\t delPt is : %.1f psi \n",delPt);
printf("\t allowable delPa is 10 psi \n");
//end
```

Scilab code Exa 16.4 Calculation of a Longitudinal Fin Shell and tube Exchanger

```
printf("\t example 16.4 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=250; // inlet hot fluid ,F
T2=100; // outlet hot fluid ,F
t1=80; // inlet cold fluid ,F
t2=100; // outlet cold fluid ,F
W=30000; // lb/hr
w=50500; // lb/hr
printf("\t 1.for heat balance \n")
C=0.225; // Btu/(lb)*(F)
Q=((W)*(C)*(T1-T2)); // Btu/hr
printf("\t total heat required for oxygwn is : %.2e
   Btu/hr \n",Q);
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for water is : %.2e
   Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.1f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.4f \n",S);
printf("\t FT is 0.87 \n"); // from fig 18
delt=(0.87*LMTD); // F
printf("\t delt is : %.1f F \n",delt);
Tc=(T2+T1)/(2); // caloric temperature of hot fluid ,
   F
printf("\t caloric temperature of hot fluid is : %.0
   f F \n",Tc);
tc=((t1)+(t2))/(2); // caloric temperature of cold
   fluid ,F
printf("\t caloric temperature of cold fluid is : %
```

```
    .0f F \n",tc);
printf("\t hot fluid:shell side,oxygen \n");
ID=19.25; // in, table 11
OD=1; // in, table 11
as=((3.14*ID^2/(4))-(70*3.14*OD^2/(4))
    -(70*20*0.035*0.5))/(144);
printf("\t flow area is : %.2f ft^2 \n",as);
p=(70*3.14*(OD))-(70*20*0.035)+(70*20*0.5*2);
printf("\t wetted perimeter : %.2e in \n",p);
De=(4*as*12/(p));
printf("\t De : %.3f ft \n",De);
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n
    ",Gs);
mu1=0.0545; // at 175F,lb/(ft)*(hr), from fig.15
Res=((De)*(Gs)/mu1); // reynolds number
printf("\t reynolds number is : %.3e \n",Res);
jH=59.5; // from fig.16.10a
k=0.0175;
Z=0.89; // Z=((c)*(mu1)/k)^(1/3), fig
hf=((jH)*(k/De)*(Z)); //using eq.6.15,Btu/(hr)*(ft
    ^2)*(F)
printf("\t individual heat transfer coefficient is :
     %.1f Btu/(hr)*(ft^2)*(F) \n",hf);
Rdo=0.003;
hdo=(1/Rdo);
hf1=(hdo*hf)/(hdo+hf); // eq 16.37
printf("\t hf1 : %.1f \n",hf1);
hfi1=142; // fig 16.9
printf("\t cold fluid:inner tube side,water \n");
at1=0.479; // table 10
L=16;
Nt=70;
n=4;
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq
    .7.48
printf("\t flow area is : %.4f ft^2 \n",at);
D=0.0652; // ft
```

```
row=62.5;
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*row));
printf("\t V is : %.2f fps \n",V);
mu2=1.94; // at 90F,lb/(ft)*(hr)
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=(940*0.96); // fig 25
printf("\t hi : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
Rdi=0.003;
hdi=(1/Rdi);
hi1=(hdi*hi)/(hdi+hi);
printf("\t hi1 : %.0f Btu/(hr)*(ft^2)*(F) \n",hi1);
UDi=((hfi1)*(hi1)/(hi1+hfi1)); // eq 16.41,Btu/(hr)*(ft^2)*(F)
printf("\t overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UDi);
A2=0.2048; // actual surface supplied for each tube,ft^2,from table 10
A=(Nt*L*A2); // ft^2
printf("\t total surface area is : %.0f ft^2 \n",A);
UDi1=((Q)/((A)*(delt)));
printf("\t design overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UDi1);
Re=(1/UDi1)-(1/UDi);
printf("\t excess fouling factor : %.5f \n",Re);
Ro=9.27; //Adding to the outside fouling factor
Rdo1=Rdo+(Re*Ro);
printf("\t Rdo : %.4f \n",Rdo1);
hf2=(hf/(1+(hf*Rdo1)));
printf("\t hf2 : %.1f \n",hf2);
hfi2=113;
UDi2=((hfi2)*(hi1)/(hi1+hfi2)); // eq 16.41,Btu/(hr)*(ft^2)*(F)
printf("\t overall coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",UDi2);
```

```
printf(”\t pressure drop  for annulus \n”);
De1=0.0433; // ft
Res1=(De1*Gs/mu1);
printf(”\t reynolds number : %.2e \n”,Res1);
f=0.00025; // fig 16.10
s=0.00133;
delPs=((f*(Gs^2)*(L))/(5.22*(10^10)*(De1)*(s)*(1))); // using eq.7.44,psi
printf(”\t delPs is : %.1f psi \n”,delPs);
printf(”\t allowable delPa is 2 psi \n”);
printf(”\t pressure drop  for inner pipe \n”);
f=0.00021; // friction factor for reynolds number 29100, using fig.26
s=1;
delPt=((f*(Gt^2)*(L)*(n))/(5.22*(10^10)*(0.0625)*(s)*(1))); // using eq.7.45,psi
printf(”\t delPt is : %.0f psi \n”,delPt);
printf(”\t allowable delPa is 10 psi \n”);
//end
```

Scilab code Exa 16.5 `Calculation of a Transverse fin Air Cooler`

```
printf(”\t example 16.5 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=250; // inlet hot fluid,F
T2=200; // outlet hot fluid,F
t1=150; // inlet cold fluid,F
t2=190; // outlet cold fluid,F
W=100000; // lb/hr
w=31200; // lb/hr
printf(”\t 1.for heat balance \n”)
C=0.25; // Btu/(lb)*(F)
Q=((W)*(C)*(T1-T2)); // Btu/hr
printf(”\t total heat required for air is : %.2e Btu
```

```
/hr \n",Q);
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for water is : %.2e Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD);
R=((T1-T2)/(t2-t1));
printf("\t R is : %.1f \n",R);
S=((t2-t1)/(T1-t1));
printf("\t S is : %.4f \n",S);
printf("\t FT is 0.985 \n"); // from fig 18
delt=(0.985*LMTD); // F
printf("\t delt is : %.1f F \n",delt);
Tc=(T2+T1)/(2); // caloric temperature of hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+(t2))/(2); // caloric temperature of cold fluid ,F
printf("\t caloric temperature of cold fluid is : %.0f F \n",tc);
Af=(3.14*2*8*12*(1.75^2-1^2))/(4);
Ao=((3.14*1*12)-(3.14*1*8*0.035*12));
printf("\t fin surface is : %.0f in^2/lin ft \n",Af);
printf("\t bare tube surface is : %.1f in^2/lin ft \n",Ao);
A=(Af+Ao);
printf("\t total outside surface : %.1f ft^2/lin ft \n",A);
p=(2*3*2*8*12/8)+(((12)-(8*0.035*12))*(2));
printf("\t projected perimeter : %.1f in/ft \n",p);
De=(2*A/(3.14*p*12)); // eq 16.104
```

```
printf(”\t De : %.3f ft \n”,De);
// 21 tubes may be fit in one :vertical bank (Fig.16.19b) ,20 tubes in alternating banks for triangular pitch
as=((4^2*12^2)-(21*1*48)-((21)*(2*0.035*3*8*48/8)))/(144); // fig 16.19
printf(”\t flow area : %.1f ft^2 \n”,as);
printf(”\t hot fluid:shell side,oxygen \n”);
Gs=(W/as); // mass velocity,lb/(hr)*(ft^2)
printf(”\t mass velocity is : %.2e lb/(hr)*(ft^2) \n”,Gs);
mu1=0.052; // at 225F,lb/(ft)*(hr), from fig.15
Res=((De)*(Gs)/mu1); // reynolds number
printf(”\t reynolds number is : %.2e \n”,Res);
jf=157; // from fig.16.18a
k=0.0183;
Z=0.89; // Z=((c)*(mu1)/k)^(1/3), fig
phys=1;
hf=((jf)*(k/De)*(Z)); //using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf(”\t individual heat transfer coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n”,hf);
Rdo=0.003;
hdo=(1/Rdo);
hf1=(hdo*hf)/(hdo+hf); // eq 16.37
printf(”\t hf1 : %.1f \n”,hf1);
hfi1=142; // fig 16.9
printf(”\t cold fluid:inner tube side,water \n”);
at1=0.546; // table 10
L=4;
Nt=21;
n=1;
at=((Nt*at1)/(144*n)); // total area,ft^2,from eq.7.48
printf(”\t flow area is : %.4f ft^2 \n”,at);
D=0.0695; // ft
row=62.5;
Gt=(w/(at)); // mass velocity,lb/(hr)*(ft^2)
```

```
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*row));
printf("\t V is : %.2f fps \n",V);
mu2=0.895; // at 170F,lb/(ft)*(hr)
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
hi=(710*0.94); // fig 25
printf("\t hi : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
Rdi=0.003;
hdi=(1/Rdi);
hi1=(hdi*hi)/(hdi+hi); // 16.40
printf("\t hi1 : %.0f Btu/(hr)*(ft^2)*(F) \n",hi1);
k1=60; // table 3 , for brass
// yb=0.00146 ft
X=((0.875-0.5)/12)*(21.5/(60*0.00146))^(1/2);
printf("\t X :%.2f \n",X);
nf=0.91; // from fig 16.13a , by comparing X value
Ai=0.218; // ft^2/ft
hfi2=((nf*Af/144)+(Ao/144))*(hf1/Ai); // eq 16.34
printf("\t hfi2 : %.0f \n",hfi2);
UDi=((hfi2)*(hi1)/(hi1+hfi2)); // eq 16.41,Btu/(hr)*(ft^2)*(F)
printf("\t overall coefficient is : %.0f Btu/(hr)*(ft^2)*(F) \n",UDi);
A=(21*4*Ai); // ft^2
printf("\t inside surface per bank is : %.1f ft^2 \n",A);
Ai1=(Q/(UDi*delt));
printf("\t Ai1 : %.0f ft^2 \n",Ai1);
Nb=(Ai1/A);
printf("\t number of banks : %.0f \n",Nb);
Vn=(4*4*1.95/12)-(41*3.14*1*4/(2*4*144))-((41*3.14*0.035*8*4/(144*2*4))*(1.75^2-1^2)); // fig 16.19b
printf("\t net free volume : %.2f ft^3 \n",Vn);
Af1=(41*2.34*4/2);
printf("\t frictional surface : %.0f ft^2 \n",Af1);
```

```
printf(”\t pressure drop  for annulus \n”);
De1=(4*Vn/Af1); // ft
printf(”\t De1 : %.2f ft \n”,De1);
Res1=(De1*Gs/mu1);
printf(”\t reynolds number : %.2e \n”,Res1);
f=0.0024; // fig 16.18b
s=0.000928;
Lp=1.95;
R1=0.538; // R1=(De1/ST)^(0.4)
R2=1; // R2=(SL/ST)^0.6
delPs=((f*(Gs^2)*(Lp)*(R1)*(R2))/(5.22*(10^10)*(De1)*(s)*(1)));
printf(”\t delPs is : %.2f psi \n”,delPs);
printf(”\t pressure drop  for inner pipe \n”);
f=0.0002; // friction factor for reynolds number 30400, using fig.26
s=1;
delPt=((f*(Gt^2)*(L)*(Nb))/(5.22*(10^10)*(0.0695)*(s)*(1))); // using eq.7.45,psi
printf(”\t delPt is : %.2f psi \n”,delPt);
//end
```

Chapter 17

Cooling Towers

Scilab code Exa 17.1 `Calculation of the Enthalpy of Saturated Air`

```
printf("\t example 17.1 \n");
pw=0.4298; // psia , at 75F, table 7
pt=14.696; // psia
t=75;
Mw=18;
Ma=29;
X=(pw/(pt-pw))*(Mw/Ma);
printf("\t humidity is : %.4f lb water/lb air \n",X)
   ;
H=(X*t)+(1051.5*X)+(0.24*t); // eq 17.54
printf("\t enthalpy at 75F is : %.1f Btu/lb dry air
   \n",H);
// end
```

Scilab code Exa 17.2 `Calculation of the Number of Diffusion Units`

```
printf("\t example 17.2 \n");
printf("\t approximate values are mentioned in the
   book \n");
```

```
printf(”\t by numerical integration \n”);
T1=85;
T2=120;
A=576; // ground area, from fig 17.12
L=1500*(500/576);
G=1400;
R=(L/G);
printf(”\t R is : %.2f \n”,R);
H1=39.1; // fig 17.12
H2=H1+(R*(T2-T1));
printf(”\t H2 is : %.1f Btu \n”,H2);
// The area between the saturation line and the operating line represents the potential for heat transfer
// at T=85F
Hs=50; // fig 17.12
d1=(Hs-H1);
printf(”\t difference is : %.1f \n”,d1);
//at t=90
Hs=56.7; // fig 17.12
H=43.7; // fig 17.12
d2=Hs-H;
printf(”\t difference is : %.1f \n”,d2);
d=(d1+d2)/(2);
printf(”\t average of difference is : %.1f \n”,d);
dT=5; // F
nd1=(dT/d);
printf(”\t nd1 is : %.3f \n”,nd1);
// similarly calculating nd at each temperature and adding them will give you total nd value
nd=1.70;
printf(”\t number of diffusing units : %.2f \n”,nd);
printf(”\t log mean enthalpy difference \n”);
dt=49.9; // diff. of enthalpies at top of the tower, from table in solution
db=10.9; // diff of enthalpies at bottom of the tower,from table in solution
LME=(dt-db)/(2.3*log10(dt/db));
```

```
printf("\t log mean of enthalpy : %.1f Btu/lb \n",LME);
nd=(T2-T1)/(LME);
printf("\t number of diffusing units are : %.2f \n",nd);
// The error is naturally larger the greater the range
//end
```

Scilab code Exa 17.3 `Calculation of the Required Height of Fill`

```
printf("\t example 17.3 \n");
printf("\t approximate values are mentioned in the book \n");
// Since the loading is based on 1 ft2 of ground area
nd=1.7;
L=1302;
Kxa=115;
Z=(nd*L)/(Kxa);
printf("\t Z is : %.1f ft \n",Z);
HDU=(Z/nd);
printf("\t height of diffusion unit : %.1ff ft \n",HDU);
// end
```

Scilab code Exa 17.4 `Determination of a Cooling tower Guarantee`

```
printf("\t example 17.4 \n");
printf("\t approximate values are mentioned in the book \n");
```

```
// The area between the saturation line and the operating line represents the potential for heat transfer
// at T=79.3F
Hs=43.4; // fig 17.12
H=30.4; // fig 17.12
d1=(Hs-H);
printf("\t difference is : %.1f \n",d1);
//at t=85
Hs=50; // fig 17.12
H=35.7; // fig 17.12
d2=Hs-H;
printf("\t difference is : %.1f \n",d2);
d=(d1+d2)/(2);
printf("\t average of difference is : %.2f \n",d);
dT=(85-79.3); // F
nd1=(dT/d);
printf("\t nd1 is : %.3f \n",nd1);
// similarly calculating nd at each temperature and adding them will give you total nd value
nd=1.72;
printf("\t number of diffusing units : %.2f \n",nd);
// end
```

Scilab code Exa 17.5 `The Recalculation of Cooling tower Performance`

```
printf("\t example 17.5 \n");
printf("\t approximate values are mentioned in the book \n");
T1=85;
T2=120;
R=0.93; // R=(L/G), for 1500 gpm
printf("\t for 120percent of design \n");
R1=1.2*R;
printf("\t R is : %.3f \n",R1);
```

```
H1=39.1; // at 87.2F
H2=H1+(R1*(T2-T1));
printf(”\t H2 is : %.1f Btu \n”,H2);
// The area between the saturation line and the operating line represents the potential for heat transfer
// at T=87.2F
Hs=53.1; // from table in the solution
d1=(Hs-H1);
printf(”\t difference is : %.1f \n”,d1);
//at t=90
Hs=56.7; // fig 17.12
H=42; // fig 17.12
d2=Hs-H;
printf(”\t difference is : %.1f \n”,d2);
d=(d1+d2)/(2);
printf(”\t average of difference is : %.1f \n”,d);
dT=(90-87.2); // F
nd1=(dT/d);
printf(”\t nd1 is : %.3f \n”,nd1);
// similarly calculating nd at each temperature and adding them will give you total nd value
nd=1.53;
printf(”\t number of diffusing units : %.2f \n”,nd);
printf(”\t for 80 percent of design \n”);
R2=0.8*R;
printf(”\t R is : %.3f \n”,R2);
H1=39.1; // at 87.2F
H2=H1+(R2*(T2-T1));
printf(”\t H2 is : %.0f Btu \n”,H2);
// The area between the saturation line and the operating line represents the potential for heat transfer
// at T=82.5F
Hs=47.2; // from table in the solution
d1=(Hs-H1);
printf(”\t difference is : %.1f \n”,d1);
//at t=85
```

```
Hs=50; // fig 17.12
H=40.8; // fig 17.12
d2=Hs-H;
printf("\t difference is : %.1f \n",d2);
d=(d1+d2)/(2);
printf("\t average of difference is : %.1f \n",d);
dT=(85-82.5); // F
nd1=(dT/d);
printf("\t nd1 is : %.3f \n",nd1);
// similarly calculating nd at each temperature and
   adding them will give you total nd value
nd=1.92;
printf("\t number of diffusing units : %.2f \n",nd);
X=[1.115 0.93 0.74];
Y=[1.53 1.70 1.92];
plot2d(X,Y,style=3,rect=[0.7,1.4,1.3,2]);
xtitle("KxaV/L vs L/G","L/G","nd");
printf("\t trial 1 \n");
R3=1.1;
printf("\t R is : %.3f \n",R3);
H1=34.5; // at 87.2F
H2=H1+(R3*(T2-T1));
printf("\t H2 is : %.0f Btu \n",H2);
// The area between the saturation line and the
   operating line represents the potential for heat
   transfer
// at T=85F
Hs=50; // from table in the solution
d1=(Hs-H1);
printf("\t difference is : %.1f \n",d1);
//at t=90
Hs=56.7; // fig 17.12
H=40; // fig 17.12
d2=Hs-H;
printf("\t difference is : %.1f \n",d2);
d=(d1+d2)/(2);
printf("\t average of difference is : %.1f \n",d);
dT=(90-85); // F
```

```
nd1=(dT/d);
printf(”\t nd1 is : %.3f \n”,nd1);
// similarly calculating nd at each temperature and adding them will give you total nd value
nd=1.48;
printf(”\t number of diffusing units : %.2f \n”,nd);
R3=1.19; // from fig 17.14
printf(”\t L/G is : %.2f \n”,R3);
printf(”\t trial 2 \n”);
R4=1.2;
printf(”\t R4 is : %.3f \n”,R4);
H1=34.5; // at 87.2F
H2=H1+(R4*(T2-T1));
printf(”\t H2 is : %.1f Btu \n”,H2);
// The area between the saturation line and the operating line represents the potential for heat transfer
// at T=85F
Hs=50; // from table in the solution
d1=(Hs-H1);
printf(”\t difference is : %.1f \n”,d1);
//at t=90
Hs=56.7; // fig 17.12
H=40.5; // fig 17.12
d2=Hs-H;
printf(”\t difference is : %.1f \n”,d2);
d=(d1+d2)/(2);
printf(”\t average of difference is : %.1f \n”,d);
dT=(90-85); // F
nd1=(dT/d);
printf(”\t nd1 is : %.3f \n”,nd1);
// similarly calculating nd at each temperature and adding them will give you total nd value
nd=1.56;
printf(”\t number of diffusing units : %.2f \n”,nd);
R3=1.08; // from fig 17.14
printf(”\t L/G is : %.2f \n”,R3);
// end
```

Scilab code Exa 17.6 Calculation of a Direct contact Gas Cooler

```
printf("\t example 17.6 \n");
printf("\t approximate values are mentioned in the book \n");
// basis 1ft^2 ground area
//Assumption: 20 per cent of the initial vapor content of the gas enters the water body
X1=(1.69/(14.7-1.69))*(18/29);
printf("\t X1 : %.4f lb/lb \n",X1);
G=1500;
w1=G*X1;
printf("\t total water in inlet gas : %.2f lb/hr \n",w1);
// The inlet gas is at 300F and a 120F dew point. Use 0.25 Btu/(lb)( F ) for the specific heat of nitrogen
H1=(0.0807*120)+(0.0807*1025.8)+(0.45*0.0807*(300-120))+(0.25*300); // eq 17.55
printf("\t H1 : %.0f Btu/lb dry air \n",H1);
X2=(w1*(1-.2)/G);
printf("\t outlet gas humidity : %.5f lb/lb \n",X2);
pw=(X2*29*14.7/18)/(1+(X2*29/18));
printf("\t pw : %.3f psia \n",pw);
Tw=112.9; // F, from table 7 for above pw
// The outlet gas has a temperature of 200 F  and a 112.9 F  dew point
H2=(X2*Tw)+(X2*1029.8)+(X2*0.45*(200-Tw))+(0.25*200); // eq 17.55
printf("\t H2 : %.1f Btu/lb dry air \n",H2);
q=G*(H1-H2);
printf("\t total heat load : %.2e Btu/hr \n",q);
w2=q/(120-85);
printf("\t water loading : %.2e lb/hr \n",w2);
```

```
printf(”\t interval 1 \n”);
// (Kxa*delV/L)= 0 t0 0.05
nd=0.05; // nd=Kxa*V/L
Le=0.93; // fig 17.4 at 300F
C=(0.25)+(0.45*X1);
printf(”\t C : %.3f Btu/(lb)*(F) \n”,C);
haV=(nd*w2*Le*C);
printf(”\t haV : %.1f Btu/(hr)*(F) \n”,haV);
qc=(haV*(300-120));
printf(”\t qc : %.2e Btu/hr \n”,qc);
delT=(qc/(C*G));
printf(”\t delT : %.1f F \n”,delT);
T1=(300-delT);
printf(”\t T(0.05) : %.1f F \n”,T1);
delt=(qc/w2);
printf(”\t delt : %.2f F \n”,delt);
t1=(120-delt);
printf(”\t t(0.05) : %.1f F \n”,t1);
printf(”\t interval 2 \n”);
// (Kxa*delV/L)= 0.05 to 0.15
nd1=0.1;
haV1=(nd1*w2*Le*C);
printf(”\t haV1 : %.1f Btu/(hr)*(F) \n”,haV1);
qc1=(haV1*(T1-t1));
printf(”\t qc1 : %.1e Btu/hr \n”,qc1);
delT1=(qc1/(C*G));
printf(”\t delT1 : %.1f F \n”,delT1);
T2=(T1-delT1);
printf(”\t T(0.15) : %.2f F \n”,T2);
X3=0.0748; // at 117.6F
w3=(nd1*w2*(0.0807-X3));
printf(”\t water diffused during interval : %.3f lb/hr \n”,w3);
w4=(w1-w3);
printf(”\t water remaining : %.2f lb/hr \n”,w4);
l1=1027; // Btu/lb , l1= lamda at 117.6F
qd=(w3*l1);
printf(”\t qd : %.0f Btu/hr \n”,qd);
```

```
q1=(qd+qc1);
printf(”\t q1 : %.0f Btu/hr \n”,q1);
delt1=(q1/w2);
printf(”\t delt1 : %.2f F \n”,delt1);
t2=(t1-delt1);
printf(”\t t(0.15) : %.1f F \n”,t2);
X4=0.0640; // at 112.5
X5=(w4/G);
printf(”\t X(112.5F) : %.4f lb/lb \n”,X5);
printf(”\t interval 3 \n”);
// (Kxa*delV/L)= 0.15 to 0.25
nd1=0.1;
haV1=(nd1*w2*Le*C);
printf(”\t haV1 : %.1f Btu/(hr)*(F) \n”,haV1);
qc2=(haV1*(T2-t2));
printf(”\t qc2 : %.2e Btu/hr \n”,qc2);
delT2=(qc2/(C*G));
printf(”\t delT2 : %.1f F \n”,delT2);
T3=(T2-delT2);
printf(”\t T(0.25) : %.1f F \n”,T3);
w5=(nd1*w2*(X5-X4));
printf(”\t water diffused during interval : %.3f lb/hr \n”,w5);
w6=(w4-w5);
printf(”\t water remaining : %.2f lb/hr \n”,w6);
l2=1030; // Btu/lb , l1= lamda at 112.5F
qd1=(w5*l2);
printf(”\t qd1 : %.2e Btu/hr \n”,qd1);
q2=(qd1+qc2);
printf(”\t q2 : %.3e Btu/hr \n”,q2);
delt2=(q2/w2);
printf(”\t delt2 : %.2f F \n”,delt2);
t3=(t2-delt2);
printf(”\t t(0.25) : %.1f F \n”,t3);
X6=0.0533; // at 106.5
X7=(w6/G);
printf(”\t X(106.5F) : %.4f lb/lb \n”,X7);
// The calculations of the remaining intervals until
```

```
 a. gas temperature of 200 F  is reached are shown in Fig. 17.17
w7=21.92; // total water diffused from table in solution
d=(w7/w1)*100;
printf("\t calculated diffusion : %.0f \n",d);
printf("\t Using some standard low-pressure-drop data \n");
// For G = 1500, extrapolate to L = 2040 on logarithmic coordinates. Kxa = 510.
ndt=.54; // from 1st table in solution
Kxa=510; // from 2nd table in solution
Z=(ndt*w2/Kxa);
printf("\t tower height : %.2f ft \n",Z);
A=(50000/G);
printf("\t cross section : %.1f ft^2 \n",A);
// end
```

Scilab code Exa 17.7 Approximate Calculation of a Gas Cooler

```
printf("\t example 17.7 \n");
printf("\t approximate values are mentioned in the book \n");
C=0.28; // assumption
w=50000; // lb/hr
G=1500;
Qs=(w*C*(500-200));
Qd=(w/G)*(22685); // qd=22685, from previous prblm
printf("\t sensible heat : %.1e Btu/hr \n",Qs);
printf("\t approximate diffusion : %.2e Btu/hr \n",Qd);
Q=(Qs+Qd);
printf("\t total heat : %.3e Btu/hr \n",Q);
// an allowance as high as 30 per cent of the sensible load can be made and the excess water
```

```
compensated for by throttling when the tower is in operation
w1=(Q/(120-85));
printf("\t total water quantity : %.2e lb/hr \n",w1);
// If the maximum liquid loading is taken as 2040 lb/(hr)(ft'!), the required tower cross section
A=(w1/2040);
printf("\t tower cross section : %.1f ft^2 \n",A);
w3=(w/A);
printf("\t new gas rate : %.0f lb/(hr)(ft^2) \n",w3);
// The two terminal temperature differences are (200 - 85) and (500 - 120).
LMTD=((500-120)-(200-85))/(log((500-120)/(200-85)));
printf("\t LMTD : %.0f \n",LMTD);
dt=35;
N=(dt/LMTD); // eq 17.88
printf("\t haV/L : %.2f \n",N);
Le=0.93;
nd=(N/(C*Le));
printf("\t number diffusion units : %.2f \n",nd);
// By extrapolation for G = 718 and L = 2040,Kxa=215
L=2040;
Kxa=215;
Z=(nd*L/Kxa); // calculation mistake
printf("\t height of tower : %.1f ft \n",Z);
di=(A)^(1/2);
printf(" ground dimensions : %.1f ft \n",di);
// ground dimensions are 5.8*8.3*8.3 ft
// end
```

Chapter 18

Batch and unsteady process

Scilab code Exa 18.1 `Calculation of Batch Heating`

```
printf("\t example 18.1 \n");
// specific gravity of benzene is 0.88
// specific heat of benzene is 0.48 Btu/(lb)*(F)
U=50;
A=400;
T1=400;
t1=100;
t2=300;
c=0.48;
w=40000;
C=0.60;
W=10000;
printf("\t values are approximately mentioned in the
    book \n");
printf("\t for a \n");
M=(7500*8.33*0.88);
printf("\t weight of benzene is : %.1e lb \n",M);
Q1=(w*c);
printf("\t Q1 is : %.2e Btu/(hr)*(F) \n",Q1);
Q2=(W*C);
printf("\t Q2 is : %.0e Btu/(hr)*(F) \n",Q2);
```

```
Ks=((%e)^(U*A*((1/Q1)-(1/Q2)))); // eq 18.16
printf("\t Ks is : %.3f \n",Ks);
Z=log((T1-t1)/(T1-t2));
printf("\t Z is : %.3f \n",Z);
theta=((M*(Z)*(Ks*6000-(19200)))/((Ks-1)*40000*6000));
printf("\t theta is : %.1f hr \n",theta);
printf("\t for b \n");
R=(Q1/Q2);
printf("\t R is : %.1f \n",R);
KT=((%e)^(U*(A/Q1)*(1+R^2)^(1/2)));
printf("\t KT is : %.0f \n",KT);
S=((2*(KT-1))/((KT*(R+1+(1+R^2)^(1/2)))-(R+1-(1+R^2)^(1/2)))); // eq 18.24
printf("\t S is : %.3f \n",S);
theta1=((M*Z)/(0.266*40000)); // eq 18.25
printf("\t theta1 is : %.2f hr \n",theta1);
printf("\t for c \n");
U1=100;
A1=200;
K8=((%e)^(U*(A/(2*Q1))*(1+R^2)^(1/2))); // eq 18.32
S1=((2*(K8-1)*(1+((1-0.266)*(1-(3.2*0.266)))^(1/2)))/(((K8-1)*(3.2+1))+((K8+1)*(1+3.2^2)^(1/2)))); // eq 18.31
printf("\t K8 is : %.2f \n",K8);
printf("\t S1 is : %.3f \n",S1);
theta2=((M*Z)/(0.282*40000)); // eq 18.25
printf("\t theta2 is : %.2f hr \n",theta2);
printf("\t for d \n");
K9=((%e)^(U*(A/(Q1))*(R-1)));
S2=((K9-1)/((K9*R)-1)); // eq 18.36
printf("\t K9 is : %.2f \n",K9);
printf("\t S2 is : %.2f \n",S2);
t=100;
t1=t+(S2*(T1-t)); // 18.37
printf("\t t1 is : %.0f F \n",t1);
t2=t1+(S2*(T1-t1));
printf("\t t2 is : %.0f F \n",t2);
```

```
t3=t2+(S2*(T1-t2));
printf("\t t3 is : %.0f F \n",t3);
t4=t3+(S2*(T1-t3));
printf("\t t4 is : %.0f F \n",t4);
x=0.23;
printf("\t fractional circulation is : %.2f \n",x);
N=3+x;
printf("\t total fractional circulation : %.2f \n",N);
theta3=(N*(M/w));
printf("\t theta3 is : %.2f \n",theta3);
// end
```

Scilab code Exa 18.2 `Heat Flow through a Wall`

```
printf("\t example 18.2 \n");
tav=500; // F
Ts=1000;
t0=100;
c=0.12; // Btu/(lb)*(F)
k=24; // Btu/(hr)*(ft^2)*(F/ft)
row=488; // lb/ft^3
alpha=0.41; // alpha=(k/(c*row)), ft^2/hr
x=0.333; // ft
theta=4;
printf("\t values are approximately mentioned in the book \n");
X=(x/(2*(alpha*theta)^(1/2)));
printf("\t X is : %.2f \n",X);
Y=0.142; // Y=f1(X) from fig 18.7
t=Ts+(t0-Ts)*(Y); // eq 18.43
printf("\t t si : %.0f F \n",t);
q=((k*(Ts-t0))/(3.14*alpha*theta)^(1/2)); // q=(Q/A),from eq 18.47
printf("\t q is : %.0f Btu/(hr)*(ft^2) \n",q);
```

```
q1=(2*k*(Ts-t0)*(theta/(3.14*alpha))^(1/2)); // q=(
   Q1/A). eq 18.49
printf("\t The total heat which flowed through a
   square foot of wall in the 4 hr is : %.1e Btu/ft
   ^2 \n",q1);
// end
```

Scilab code Exa 18.3 `Center line Temperature of a Shaft`

```
printf("\t example 18.3 \n");
Ts=1000;
t0=100;
alpha=0.41; // alpha=(k/(c*row)), ft^2/hr
theta=15/60;
l=1; // ft
X=(4*alpha*theta)/(l^2);
printf("\t X is : %.2f \n",X);
Y=0.155; // Y=f3*(X)from fig 18.9 when L=infinity
t=Ts+(t0-Ts)*(Y); // eq 18.52
printf("\t t si : %.1e F \n",t);
// end
```

Scilab code Exa 18.4 `The Schack Chart`

```
printf("\t example 18.4 \n");
T1=1100; // F
T2=70; // F
t1=T1+460; // R
t2=T2+460; // R
k=27; // from appendix
c=0.14; // from appendix
row=490; // from appendix
alpha=0.394;
```

```
theta=4;
l=10/12; // ft
x=0.173*10^(-8); // stefan constant
e=0.7; // emmisivity
printf("\t values are approximately mentioned in the book \n");
printf("\t for a \n");
// Assume the temperature is 500 F  after 4 hr. The coefficient from plate to air is the  sum of the radiation and convection coefficients
hri=(e*x*(t1^4-t2^4))/(T1-T2);
printf("\t radiation coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",hri); // eq 4.32
hci=(0.3*(T1-T2)^(1/4)); // eq 10.10
printf("\t convection coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",hci);
hti=hri+hci;
printf("\t total intial coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",hti);
// For the 4-hr coefficient at 500 F
hr=2.2; // Btu/(hr)*(ft^2)*(F)
hc=1.35; // Btu/(hr)*(ft^2)*(F)
ht=hr+hc;
printf("\t total intial coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",ht);
h=(hti+ht)/2;
printf("\t mean coefficient is : %.1f Btu/(hr)*(ft^2)*(F) \n",h);
X=(4*alpha*theta)/(l^2);
Y=(h*l)/(2*k);
printf("\t X is : %.1f \n",X);
printf("\t Y is : %.3f \n",Y);
Z=0.42; // Z=f3(X,Y), from fig 18.10
t=T2+((T1-T2)*Z); // eq 18.53
printf("\t t is : %.0f F \n",t);
printf("\t for b \n");
Z1=0.43; // Z=f4(X,Y), from fig 18.11
t1=T2+((T1-T2)*Z1); // eq 18.53
```

```
printf("\t temperature of center plane is : %.0f F \n",t1);
// end
```

Scilab code Exa 18.5 `The Gumey Lurle Chart`

```
printf("\t example 18.5 \n");
Ts=400;
t0=200;
k=25; // from appendix
c=0.12; // from appendix
row=490; // from appendix
alpha=0.45; // alpha=(k/(c*row))
theta=15/60;
l=8/12; // ft
h=50;
X=(4*alpha*theta)/(l^2);
Z=(2*k)/(h*l);
printf("\t X is : %.2f \n",X);
printf("\t Z is : %.1f \n",Z);
Y=0.31; // Y=(Ts-t)/(Ts-t0), from fig 18.13
t=Ts+(t0-Ts)*(Y); // eq 18.43
printf("\t t is : %.0f F \n",t);
//end
```

Scilab code Exa 18.6 `The Application of Newmans Method to Heating a Brick`

```
printf("\t example 18.6 \n");
Ts=300;
t0=70;
c=0.25; // Btu/(lb)*(F)
k=0.3; // Btu/(hr)*(ft^2)*(F/ft)
row=103; // lb/ft^3
```

```
alpha=0.01164; // alpha=(k/(c*row)), ft^2/hr
theta=1;
lx=9/12;
ly=4.5/12;
lz=2.5/12;
h=4.1;
printf("\t values are approximately mentioned in the book \n")
X1=(4*alpha*theta)/(lx^2);
Z1=(2*k)/(h*lx);
printf("\t X1 is : %.4f \n",X1);
printf("\t Z1 is : %.3f \n",Z1);
X2=(4*alpha*theta)/(ly^2);
Z2=(2*k)/(h*ly);
printf("\t X2 is : %.4f \n",X2);
printf("\t Z2 is : %.3f \n",Z2);
X3=(4*alpha*theta)/(lz^2);
Z3=(2*k)/(h*lz);
printf("\t X3 is : %.3f \n",X3);
printf("\t Z3 is : %.3f \n",Z3);
printf("\t at centre (2*x/l) is zero \n");
Yx=0.98; // fig 18.12
Yy=0.75; // fig 18.12
Yz=0.43; // fig 18.12
printf("\t at surface (2*x/l) is one \n");
Yx1=0.325; // fig 18.12
Yy1=0.29; // fig 18.12
Yz1=0.245; // fig 18.12
printf("\t center of brick \n");
t1=Ts-(Yx*Yy*Yz*(Ts-t0));
printf("\t t1 is : %.1f F \n",t1);
printf("\t corner of brick \n");
t2=Ts-(Yx1*Yy1*Yz1*(Ts-t0));
printf("\t t2 is : %.1f F \n",t2);
printf("\t center of 9 by 4.5in face \n");
t3=Ts-(Yx*Yy*Yz1*(Ts-t0));
printf("\t t3 is : %.1f F \n",t3);
printf("\t center of 9 by 2.5in face \n");
```

```
t4=Ts-(Yx*Yy1*Yz*(Ts-t0));
printf("\t t4 is : %.0f F \n",t4);
printf("\t center of 4.5 by 2.5in face \n");
t5=Ts-(Yx1*Yy*Yz*(Ts-t0));
printf("\t t5 is : %.1f F \n",t5);
printf("\t middle of long edge \n");
t6=Ts-(Yx*Yy1*Yz1*(Ts-t0));
printf("\t t6 is : %.0f F \n",t6);
//end
```

Scilab code Exa 18.7 `The Graphical Determination of the Time Temperature Distribut`

```
printf("\t example 18.7 \n");
t=20; // min
alpha=0.40; // ft^2/hr
delx=0.167; // ft
// From the conditions of Eq. (18.61) take time increments such that alpha(deltheta/delx^2)=1/2
printf("\t approximate values are mentioned in the book \n");
deltheta=(delx^2/(2*alpha));
printf("\t deltheta is : %.3f hr \n",deltheta);
N=(t/(deltheta*60));
printf("\t number of steps required : %.1f \n",N);
// end
```

Scilab code Exa 18.8 `Calculations for a Wall with Periodic Temperature Variation`

```
printf("\t example 18.8 \n");
k=0.3;
row=103;
c=0.25;
alpha=0.01164;
```

```
f=1/24;
t1=120;
t2=60;
printf("\t approximate values are mentioned in the book \n");
printf("\t temperature lag 6in below the surface \n");
x=6/12;
theta=(x/2)*(1/(3.14*f*alpha))^(1/2); // eq 18.65
printf("\t theta is : %.2f hr \n",theta);
printf("\t amplitude \n");
deltom=(t1-t2)/2;
printf("\t deltom is : %.0f F \n",deltom);
delt=(deltom)*(%e)^(-x*(3.14*f/alpha)^(1/2)); // eq 18.67
printf("\t delt is : %.1f F \n",delt); // calculation mistake in book
printf("\t temperature deviation after 2 hr \n");
theta1=2; // hr
deltx=(deltom)*((%e)^(-x*(3.14*f/alpha)^(1/2)))*cos((2*3.14*f*theta1)-(x*(3.14*f/alpha)^(1/2))); // eq 18.69
printf("\t deltx is : %.1f F \n",deltx);
printf("\t heat flow during the half period \n");
q=(k*deltom*(2/(3.14*f*alpha))^(1/2)); // eq 18.70
printf("\t heat flow is : %.0f Btu/(hr)*(ft^2) \n",q);
// end
```

Scilab code Exa 18.9 Calculation of the Length of a Bed

```
printf("\t example 18.9 \n");
G=60; // lb/(hr)*(ft^2)
De=1/12; // ft
theta=6; // hr
```

```
cs=41.3; // Btu/(ft^3)*(F)
c=0.0191; // Btu/(ft^3)*(F)
f=0.45; // void fraction
T=90;
T1=200;
t0=50;
h=(0.79*(G/De)^0.7); // eq 18.90
printf("\t h is : %.1f \n",h);
X=(h*theta/(cs*(1-f)));
Y=(T-t0)/(T1-t0);
printf("\t X is : %.0f \n",X);
printf("\t Y is : %.3f \n",Y);
row=0.0807; // lb/(ft^3) air
Z=24.5; // Z=(h*x*row/(c*G)), by comparing X an Y in fig 18.21
x=24.5*(c*G/(h*row));
printf("\t x is : %.1f ft \n",x);
// end
```

Chapter 19

Furnace Calculations

Scilab code Exa 19.1 `Lobo and Evans`

```
printf("\t example 19.1 \n");
// For orientation purposes, one can make an estimate of the number of tubes required in the radiant section by assuming avg flux is 12000 Btu/(hr)*(ft^2)
// from Fig.19.14 it can be seen that with a tube temperature of 800"F, an exit-gas temperature of 1730 F will be required to effect such a flux.
printf("\t approxiate values are mentioned in the book \n");
Q=50000000; // Btu/hr
QF=(Q/0.75); // efficiency of tank is 75%
printf("\t heat liberated by the fuel : %.3e Btu/hr \n",QF);
w1=(QF/17130); // heating value of fuel is 17130Btu/lb
printf("\t fuel quantity : %.2e lb/hr \n",w1);
w2=(w1*17.44); // lb of fuel fired with 17.44lb of air
printf("\t air required : %.2e lb/hr \n",w2);
w3=(w1*0.3); // 0.3 lb of air is used for atomizing
```

```
lb of fuel
printf(”\t steam for atomizing : %.2e lb/hr \n”,w3);
QA=(w2*82); // heating value at 400F is 82Btu/lb
printf(”\t QA is : %.2e Btu/hr \n”,QA);
printf(”\t QS is negligible \n”);
QW=(0.02*QF);
printf(”\t QW is : %.2e Btu/hr \n”,QW);
Qnet=(QF+QA-QW);
printf(”\t Qnet is : %.2e Btu/hr \n”,Qnet);
//Heat out m gases at 1730 F , 25 per cent excess air , 476 Btu/lb of flue gas
QG=(476*(w1+w2+w3));
printf(”\t QG is : %.2e Btu/hr \n”,QG);
Q1=(Qnet-QG);
printf(”\t Q1 is : %.2e Btu/hr \n”,Q1); // calculation mistake in book
A=(3.14*38.5*(5/12)); // area of tube
printf(”\t area of tube is : %.1f ft^2 \n”,A);
Nt=(Q1/(12000*A)); // 12000 is avg flux
printf(”\t estimated number of tubes : %.0f \n”,Nt);
// The layout of the cross section of the furnace may be as shown m Fig. 19.16.
// center to center distance is 8(1/2)in
Acp=(8.5*38.5/12);
printf(”\t cold plane surface per tube : %.1f ft^2 \n”,Acp); // calculation mistake in book
a=0.937; // a=alpha, from fig 19.11 as Ratio of center-to-center/OD is 1.7
Acp1=(Acp*a);
printf(”\t Acp1 is : %.0f ft^2 \n”,Acp1);
Acpt=(Acp1*Nt);
printf(”\t total cold plane surface is : %.1e ft^2 \n”,Acpt);
A1=(2*20.46*14.92); // from fig 19.16
printf(”\t surface of end walls : %.0f ft^2 \n”,A1);
A2=(38.5*14.92); // from fig 19.16
printf(”\t surface of side wall : %.0f ft^2 \n”,A2);
A3=(38.5*9.79); // from fig 19.16
```

```
printf("\t surface of bridge walls : %.0f ft^2 \n",A3);
A4=(2*20.46*38.5); // from fig 19.16
printf("\t surface of floor and arch : %.0f ft^2 \n",A4);
AT=(A1+A2+A3+A4);
printf("\t AT is : %.0f ft^2 \n",AT);
AR=(AT-Acpt);
printf("\t AR is : %.0f ft^2 \n",AR);
Ar=(AR/Acpt);
printf("\t ratio of areas is : %.2f \n",Ar);
printf("\t dimension ratio is 3:2:1 \n");
L=((2/3)*(38.5*20.46*14.92)^(1/3));
printf("\t length is : %.0f ft \n",L);
printf("\t gas emissivity \n");
// From the analysis of the fuel, the steam quantity, and the assumption that the humidity of the air is 50 per cent of saturation at 60F, the partial pressures of CO2 and H2O in the combustion gases with 25 per cent excess air are
pCO2=0.1084;
pH2O=0.1248
pCO2L=1.63; // pCO2L=(pCO2*L)
pH2OL=1.87;
P=((pCO2)/(pCO2+pH2O));
printf("\t percentage correction at P : %.3f \n",P);
Pt=pCO2L+pH2OL;
printf("\t Pt is : %.2f \n",Pt);
// %correction estimated to be 8%
eG=(((6500+14500)-(650+1950))/(39000-4400))*((100-8)/100); // values from fig 19.12 and 19.13, eq 19.5
printf("\t eG is : %.3f \n",eG);
f=0.635; // from fig 19.15 as (AR/Acpt)=1.09 and eG=0.496
printf("\t overall exchange factor : %.3f \n",f);
Z=(Q1/(Acpt*f));
printf("\t Z is : %.2e \n",Z);
```

```
printf("\t TG required (at Ts = 800F) = 1670F compared with 1730 F  assumed in heat balance) \n");
// end
```

Scilab code Exa 19.2 `Calculation of a Furnace by the Method of Wilson and Lobo and`

```
printf("\t example 19.2 \n");
QF=50000000;
G=22.36;
Acpt=1500;
printf("\t approxiate values are mentioned in the book \n");
Q=(QF/(1+(G/4200)*(QF/Acpt)^(1/2))); // eq 19.15
printf("\t Q is : %.2e Btu/hr \n",Q);
printf("\t The radiant-section average rate will be 8350 Btu/(hr) (ft2), and the exit-flue-gas temperature 1540 F  by heat balance. \n");
// end
```

Scilab code Exa 19.3 `Calculation of Performance by the Orrok Hudson Equation`

```
printf("\t example 19.3 \n");
Qr=1.5; // Qr=(QF2/QF1)
Cr=1.5; // Cr=(CR2/CR1)
Gr=140/125; // Gr=(G2/G1)
Qr1=0.38; // Qr1=(Q1/QF1)
printf("\t approxiate values are mentioned in the book \n");
a1=1.63; // a1=(G1*(CR1/27)^(1/2)), from eq 19.17
printf("\t a1 is : %.2f \n",a1);
a2=1.37*(a1); // a2=(G2*(CR2/27)^(1/2))
printf("\t a2 is : %.2f \n",a2);
```

```
Qr2=(1/(1+a2)); // Qr2=(Q2/QF2),from eq 19.15
printf(”\t Qr2 is : %.2f \n”,Qr2);
Q21=(Qr2/Qr1)*(Qr); // Q21=(Q2/Q1)
printf(”\t ratio of heats is : %.2f \n”,Q21);
printf(”\t Hence the radiant absorption will be increased only 22 per cent for an increase of 50 per cent in the heat liberated. \n”);
// end
```

Scilab code Exa 19.4 Calculation of the Equivalent Radiant Coefficient

```
printf(”\t example 19.4 \n”);
eS=0.9; // assumed
TG=1500;
TS=650;
pCO2=0.1084;
pH2O=0.1248;
printf(”\t approxiate values are mentioned in the book \n”);
L=(0.4*8.5)-(0.567*5);  // table 19.1
printf(”\t L is : %.3f ft \n”,L);
pH2OL=0.1248*L;
pCO2L=0.1084*L;
printf(”\t pH2OL is : %.4f atm-ft \n”,pH2OL);
printf(”\t pCO2L is : %.4f atm-ft \n”,pCO2L);
qH2O=1050; // at TG, from fig 19.12 ana 19.13
qCO2=1700; // at TG, from fig 19.12 ana 19.13
qTG=(qH2O+qCO2);
printf(”\t qTG is : %.0f \n”,qTG);
qsH2O=165; // at TS, from fig 19.12 ana 19.13
qsCO2=160; // at TS, from fig 19.12 ana 19.13
qTS=(qsH2O+qsCO2);
printf(”\t qTG is : %.0f \n”,qTS);
q=(0.9*(qTG-qTS)); // q=(QRC/A)
printf(”\t q is : %.1f \n”,q);
```

```
P=((pCO2)/(pCO2+pH2O));
printf("\t percentage correction at P : %.3f \n",P);
Pt=pCO2L+pH2OL;
printf("\t Pt is : %.4f \n",Pt);
// %correction estimated to be 2%
q1=(q*0.98); // // q1=(QRC/A)
printf("\t q1 is : %.2e \n",q1);
hr=(q1/(TG-TS));
printf("\t radiation coefficient is : %.2f Btu/(hr)
    *(ft^2)*(F) \n",hr);
//end
```

Scilab code Exa 19.5 `Furnace calculation`

```
printf("\t example 19.5 \n");
Q=500000;
printf("\t approxiate values are mentioned in the
    book \n");
a=(3.5+(3.14*4*(120/360)))/(2); // a=(alpha*Acp)
    from fig 19.17
AR=(3+3.6+3);
printf("\t a is : %.2f ft^2/ft \n",a);
printf("\t AR is : %.1f ft^2/ft \n",AR);
// Arbitrarily neglecting end wa.lls and also .the
    side wall refractory over 3'0" above the floor
R=(AR/a);
printf("\t ratio of two areas is : %.2f \n",R);
eG=0.265;
TG=1174; // F
TS=500; // F
f=0.56; // from fig 19.15 as (AR/Acpt)=2.49 and eG
    =0.265
q=15300; // at TG and TS,q=(Q/(a*f))
// However, the convection coefficient is small, 1.0
        Btu/(hr)(ft2)("F), and AR/a is not 2.0 as in
```

```
the assumptions for the Lobo and Evans equation.
q1=(q)-(7*(TG-TS)); // q1=(Q/(a*f))
printf("\t q1 is : %.2e Btu/(hr)*(ft^2) \n",q1);
q2=(q1*f); // q2=(Q/(a))
printf("\t q2 is : %.2e Btu/(hr)*(ft^2) \n",q2);
printf("\t convection rate basis \n");
q3=(1*(TG-TS)*(4.2/a)); // q2=(Q/(a))
printf("\t q3 is : %.1e Btu/(hr)*(ft^2) \n",q3); // calculation mistake in book
qt=(q2+q3); // qt=(Q/(a))
printf("\t qt is : %.2e Btu/(hr)*(ft^2) \n",qt);
ar=(Q/qt);
printf("\t required a is : %.0f ft^2 \n",ar);
L=(ar/a);
printf("\t length required is : %.1f ft \n",L);
// end
```

Chapter 20

Additional applications

Scilab code Exa 20.1 `Calculation of a Jacketed Vessel`

```
printf("\t example 20.1 \n");
printf("\t approximate values are mentioned in the
   book \n");
T=150; // F
L=0.6; // ft
N=7500; // rev/hr
row=62.5; // lb/ft^3
mu=1.06; // at 150 F and from fig 14, lb/ft*hr
k=0.38; // Btu/(hr)*(ft^2)*(F/ft), from table 4
c=1; // Btu/(lb)*(F)
Rej=(L^2)*(N)*(row)/(mu);
printf("\t Rej is : %.1e \n",Rej);
Z=1; // Z=(mu/muw)^(0.14), regarded as 1 for water
Dj=1.01; // ft, from table 11
j=1100; // fig 20.2
hi=((j)*(k/Dj)*((c*mu/k)^(1/3))*(Z)^(0.14));
printf("\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
hoi=1500; // Btu/(hr)*(ft^2)*(F)
Uc=((hi*hoi)/(hi+hoi)); // from eq 6.38
printf("\t Uc is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=0.005;
```

```
hd=(1/Rd);
printf(”\t hd is : %.0f \n”,hd);
UD=((Uc*hd)/(Uc+hd));
printf(”\t UD is : %.0f Btu/(hr)∗(ft ˆ2)∗(F) \n”,UD);
A=3.43; // ft ˆ2
Q=32600;
delt=(Q/(UD*A));
printf(”\t temperature difference is : %.0f F \n”,delt);
Ts=(T+delt);
printf(”\t temperature of the steam : %.0f F \n”,Ts);
// end
```

Scilab code Exa 20.2 Calculation of a Tube Coil

```
printf(”\t example 20.2 \n”);
printf(”\t approximate values are mentioned in the book \n”);
T1=150; // F
T2=220; // F
L=0.6; // ft
N=7500; // rev/hr
row=62.5; // lb/ft ˆ3
mu=1.06; // at 150 F and from fig 14, lb/ft∗hr
k=0.38; // Btu/(hr)∗(ft ˆ2)∗(F/ft), from table 4
c=1; // Btu/(lb)∗(F)
Rej=(L^2)*(N)*(row)/(mu);
printf(”\t Rej is : %.1e \n”,Rej);
Z=1; // Z=(mu/muw)ˆ(0.14), regarded as 1 for water
Dj=1.01; // ft, from table 11
j=1700; // fig 20.2
hi=((j)*(k/Dj)*((c*mu/k)^(1/3))*(Z)^(0.14));
printf(”\t hi is : %.0f Btu/(hr)∗(ft ˆ2)∗(F) \n”,hi);
hoi=1500; // Btu/(hr)∗(ft ˆ2)∗(F)
```

```
Uc=((hi*hoi)/(hi+hoi)); // from eq 6.38
printf("\t Uc is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=0.005;
hd=(1/Rd);
printf("\t hd is : %.0f \n",hd);
UD=((Uc*hd)/(Uc+hd));
printf("\t UD is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
Q=32600;
A=(Q/(UD*(T2-T1)));
printf("\t Area is : %.2f ft^2 \n",A);
a=0.1309; // ft^2/ft
a1=(3.14*0.8*a);
printf("\t area per turn is : %.3f ft^2 \n",a1);
n=(A/a1);
printf("\t number of turns : %.1f \n",n);
// end
```

Scilab code Exa 20.3 Calculation of a Submerged pipe Coil Slurry Cooler

```
printf("\t example 20.3 \n");
printf("\t approximate values are mentioned in the
   book \n");
T1=675; // inlet hot fluid ,F
T2=200; // outlet hot fluid ,F
t1=120; // inlet cold fluid ,F
t2=140; // outlet cold fluid ,F
W=33100; // lb/hr
w=510000; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for oil \n");
c=0.64; // Btu/(lb)*(F)
Q=((W)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for oil is : %.2e Btu
   /hr \n",Q);
printf("\t for water \n");
```

```
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for water is : %.2e Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=230;
printf("\t LMTD is :%.0f F \n",LMTD);
Tc=((T2)+(T1))/(2); // caloric temperature of hot fluid ,F
printf("\t caloric temperature of hot fluid is : %.1f F \n",Tc);
tc=((t1)+(t2))/(2); // caloric temperature of cold fluid ,F
printf("\t caloric temperature of cold fluid is : %.0f \n",tc);
printf("\t hot fluid:inner tube side, oil \n");
at=0.0458; // flow area, ft^2, table 11
printf("\t flow area is : %.4f ft^2 \n",at);
Gt=(W/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu2=5.56; // at 400F,lb/(ft)*(hr)
D=0.242; // ft, table 11
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.2e \n",Ret);
jH=100; // from fig.24
Z=0.245; // Z=(k(c*mu/k)^(1/3)), Btu/(hr)*(ft)*(F/ft), fig 16
hi=((jH)*(Z/D)); //Hi=(hi/phyp),using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf("\t hi is : %.0f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=2.9; // ft
OD=3.5; // ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hio to the surface at the OD is :
```

```
    %.1f Btu/(hr)*(ft^2)*(F) \n",hio);
ho=150; // Btu/(hr)*(ft^2)
tw=(tc)+(((hio)/(hio+ho))*(Tc-tc)); // from eq.5.31
printf("\t tw is : %.0f F \n",tw);
tf=(tw+tc)/2;
printf("\t tf is : %.0f F \n",tf);
delt=110; // F
d0=3.5; // in, fig 10.4
Uc=((ho*hio)/(ho+hio)); // from eq 6.38
printf("\t Uc is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=0.01;
hd=(1/Rd);
printf("\t hd is : %.0f \n",hd);
UD=((Uc*hd)/(Uc+hd));
printf("\t UD is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
A=(Q/(UD*(LMTD)));
printf("\t Area is : %.0f ft^2 \n",A);
a=0.917; // ft^2/ft, table 11
L=(A/(a*24));
printf("\t pipe length : %.0f \n",L);
// end
```

Scilab code Exa 20.4 Calculation of a Trombone 2

```
printf("\t example 20.4 \n");
printf("\t approximate values are mentioned in the book \n");
T1=450; // inlet hot fluid,F
T2=150; // outlet hot fluid,F
t1=85; // inlet cold fluid,F
t2=100; // outlet cold fluid,F
W=3360; // lb/hr
w=11100; // lb/hr
printf("\t 1.for heat balance \n");
printf("\t for SO2 \n");
```

```
c=0.165; // Btu/(lb)*(F)
Q=((W)*(c)*(T1-T2)); // Btu/hr
printf("\t total heat required for SO2 is : %.3e Btu/hr \n",Q);
printf("\t for water \n");
c=1; // Btu/(lb)*(F)
Q=((w)*(c)*(t2-t1)); // Btu/hr
printf("\t total heat required for water is : %.3e Btu/hr \n",Q);
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
R=20;
S=0.0412;
FT=0.98; // fig 18
delt=(FT*LMTD);
printf("\t delt is : %.0f F \n",delt);
Tc=((T2)+(T1))/(2); // caloric temperature of hot fluid,F
printf("\t caloric temperature of hot fluid is : %.0f F \n",Tc);
tc=((t1)+(t2))/(2); // caloric temperature of cold fluid,F
printf("\t caloric temperature of cold fluid is : %.1f \n",tc);
printf("\t hot fluid:inner tube side, SO2 \n");
at=0.0512; // flow area, ft^2, table 11
printf("\t flow area is : %.4f ft^2 \n",at);
Gt=(W/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.2e lb/(hr)*(ft^2) \n",Gt);
mu2=0.041; // at 300F,lb/(ft)*(hr), fig 15
D=0.256; // ft, table 11
Ret=((D)*(Gt)/mu2); // reynolds number
printf("\t reynolds number is : %.1e \n",Ret);
```

```
jH=790; // from fig.24
Z=0.006831; // Z=(k(c*mu/k)^(1/3)), Btu/(hr)*(ft)*(F/ft)
hi=((jH)*(Z/D)); //Hi=(hi/phyp),using eq.6.15,Btu/(hr)*(ft^2)*(F)
printf("\t hi is : %.1f Btu/(hr)*(ft^2)*(F) \n",hi);
ID=3.068; // ft
OD=3.5; // ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hio to the surface at the OD is : %.1f Btu/(hr)*(ft^2)*(F) \n",hio);
printf("\t cold fluid water \n");
L=8; // ft
G=(w/(2*L));
printf("\t G : %.0f lb/(hr)*(ft) \n",G);
mu1=1.94; // at 92.5F, lb/(ft)*(hr)
Re=(4*G/mu1);
printf("\t Re is : %.2e \n",Re);
Do=0.292; // ft
ho=(65*(G/Do)^(1/3));
printf("\t ho is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
Uc=((ho*hio)/(ho+hio)); // from eq 6.38
printf("\t Uc is : %.1f Btu/(hr)*(ft^2)*(F) \n",Uc);
Rd=0.01;
hd=(1/Rd);
printf("\t hd is : %.0f \n",hd);
UD=((Uc*hd)/(Uc+hd));
printf("\t UD is : %.1f Btu/(hr)*(ft^2)*(F) \n",UD);
A=(Q/(UD*(LMTD)));
printf("\t Area is : %.1f ft^2 \n",A); // calculation mistake in book
a=0.917; // ft^2/ft, table 11
l=(A/(a*8));
printf("\t pipe length : %.2f \n",l);
// end
```

Scilab code Exa 20.5 `Calculation of an Atmospheric Jacket Water Cooler`

```
printf("\t example 20.5 \n");
printf("\t approximate values are mentioned in the book \n");
Nt=25; // number of tubes
A=50; // total projected area
Tav=100; // F
s=28; // assumption spray, lb/(min)*(ft^2)
Do=0.0833; // ft
PH=0.1562;
Y=0.874;
Z=0.466;
E=(0.171*(Do*Y*Z)^0.1); // (E/(Do*Y*Z)^0.1)=0.171, from fig 20.10
printf("\t evaporation percentage is : %.2f \n",E);
Q=(295*500*(143-130));
printf("\t heat load is : %.2e Btu/hr \n",Q);
Q1=(Q*(1-0.12));
printf("\t sensible heat is : %.2e Btu/hr \n",Q1);
t2=(90)+(Q1/(28*60*50));
printf("\t final spray temperature is : %.0f F \n",t2);
w=(s*60*50);
printf("\t total spray : %.1e lb/hr \n",w);
m=(w/(2*4*12));
printf("\t m is : %.0f lb/(hr)*(ft^2) \n",m);
mu=1.84; // lb/(ft)*(hr)
Z=((m^0.3)*Do*Y*Z/(mu*0.125));
printf("\t Z is : %.2f \n",Z);
N=3; // assume 3 horizontal rows
ho=300*(N^0.05); // (ho/(N^0.05))=300, from fig 20.11
printf("\t ho is : %.0f Btu/(hr)*(ft^2)*(F) \n",ho);
```

```
printf("\t tube side coefficient \n");
printf("\t assuming even number of passes and tube side velocity about 8fps \n");
at=0.0775; // ft^2
Gt=(295*500/(at)); // mass velocity,lb/(hr)*(ft^2)
printf("\t mass velocity is : %.1e lb/(hr)*(ft^2) \n",Gt);
V=(Gt/(3600*62.5));
printf("\t velocity is : %.2f fps \n",V);
hi=2140; // Btu/(hr)*(ft^2)*(F), fig 25
ID=0.87; // ft
OD=1; // ft
hio=((hi)*(ID/OD)); // using eq.6.5
printf("\t Correct hio to the surface at the OD is : %.2e Btu/(hr)*(ft^2)*(F) \n",hio);
Uc=((ho*hio)/(ho+hio)); // from eq 6.38
printf("\t Uc is : %.0f Btu/(hr)*(ft^2)*(F) \n",Uc);
a=0.2618; // ft^2, table 11
A1=(2*3*25*12*a);
printf("\t total surface is : %.0f ft^2 \n",A1);
T1=143; // inlet hot fluid,F
T2=130; // outlet hot fluid,F
t1=90; // inlet cold fluid,F
t2=110; // outlet cold fluid,F
delt1=T2-t1; //F
delt2=T1-t2; // F
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.1f F \n",LMTD); // calculation mistake in book
R=0.65;
S=0.377;
FT=0.97; // fig 18
delt=(FT*LMTD);
printf("\t delt is : %.1f F \n",delt);
UD=(Q/(A1*(delt)));
printf("\t UD is : %.0f Btu/(hr)*(ft^2)*(F) \n",UD);
```

```
Rd=((Uc-UD)/((UD)*(Uc))); // (hr)*(ft^2)*(F)/Btu
printf("\t actual Rd is : %.4f (hr)*(ft^2)*(F)/Btu \n",Rd);
printf("\t The assumption of three horizontal rows is satisfactory , since a dirt factor of 0.004 was required \n");
// end
```

Scilab code Exa 20.6 `Calculation of the True Temperature Difference`

```
printf("\t example 20.6 \n");
printf("\t approximate values are mentioned in the book \n");
T1=200; // inlet hot fluid ,F
T2=100; // outlet hot fluid ,F
t1=50; // inlet cold fluid ,F
t2=100; // outlet cold fluid ,F
R=((T1-T2)/(t2-t1));
printf("\t R is : %.0f \n",R);
V=((T1+T2-t1-t2)/(t2-t1))/(2);
printf("\t V is : %.1f \n",V);
u=120;
U=60;
F=((u*1)/(U*2));
printf("\t F is : %.0f \n",F);
E=1.1; // In Fig.20.18b for R = 2.0and F = 1.O,the abscissa and ordinate intersect at E =1.10.
Z=(E/V);
printf("\t Z is : %.3f \n",Z);
deltD=0.783*V; // deltD/V=0.783, from fig 20.17
printf("\t deltD is : %.3f \n",deltD);
delt=(deltD*(t2-t1));
printf("\t delt is : %.1f \n",delt);
delt1=T2-t1; //F
delt2=T1-t2; // F
```

```
printf("\t delt1 is : %.0f F \n",delt1);
printf("\t delt2 is : %.0f F \n",delt2);
LMTD=((delt2-delt1)/((2.3)*(log10(delt2/delt1))));
printf("\t LMTD is :%.0f F \n",LMTD);
// end
```

Scilab code Exa 20.7 calculation of Sand Cooling with Negllgible Resistance

```
printf("\t example 20.7 \n");
printf("\t approximate values are mentioned in the book \n");
T1=284; // inlet hot fluid,F
T2=104; // outlet hot fluid,F
t1=86; // inlet cold fluid,F
t2=104; // outlet cold fluid,F
W=1000; // lb/hr
k=0.15; // thermal conductivity
L=10;
Beta=((2*k)/(500*(2/12))); // hoi=500Btu/(hr)*(ft^2)*(F) for water
printf("\t beta is : %.4f \n",Beta);
printf("\t for sand \n");
C=0.2; // Btu/(lb)*(F)
Q=((W)*(C)*(T1-T2)); // Btu/hr
printf("\t total heat required for sand is : %.1e Btu/hr \n",Q);
c=1;
w=(Q/(t2-t1));
printf("\t w is : %.0e lb/hr \n",w);
R=((W*C)/(w*c));
printf("\t R is : %.1f \n",R);
S=((T2-T1)/(t1-T1));
printf("\t S is : %.2f \n",S);
W1=(8.33*(k*L)/C); // ((W1*C)/(k*L))=8.33 from fig 20.20b for Beta=0
```

```
printf(”\t rate per tube is : %.1f lb/hr \n”,W1);
N1=(W/W1);
printf(”\t number of tubes : %.0f \n”,N1);
printf(”\t for air assume hoi=9 and Beta=0.2 \n”);
c1=0.25;
w1=(Q/(c1*(t2-t1)));
printf(”\t w1 is : %.0e lb/hr \n”,w1);
W2=(5.23*(k*L)/C); // ((W1*C)/(k*L))=5.23 from fig 20.20b for Beta=0.2
printf(”\t rate per tube is : %.0f lb/hr \n”,W2);
N2=(W/W2);
printf(”\t number of tubes : %.0f \n”,N2);
// end
```

Scilab code Exa 20.8.1 Immersion Water Heater

```
printf(”\t example 20.8a \n”);
printf(”\t approximate values are mentioned in the book \n”);
L=3; // ft
B=2; // ft
h=18/12; // ft , height of water present in tank
printf(”\t unsteady state \n”);
m=(L*B*h*62.5);
printf(”\t Lb of water is : %.1f lb \n”,m);
t1=50;
t2=150;
c=1;
Q=(m*c*(t2-t1))/(2*3412); // kwhr
printf(”\t heat to be supplied : %.2f kwhr \n”,Q);
printf(”\t losses \n”);
Q1=(L*B*260)/(1000); // from fig 20.25c
printf(”\t from surface of water : %.2f kwhr \n”,Q1);
Q2=(5.5*((2*B*2)+(2*L*B))/(1000)); // from fig 20.25
```

```
c
printf("\t from sides of vessel : %.2f kwhr \n",Q2);
printf("\t losses from bottom are negigible \n");
Qt=(Q+Q1+Q2);
printf("\t total requirement : %.2f kwhr \n",Qt);
printf("\t steady state \n");
m1=8; // gal/hr
Qs=(m1*8.33*c*(t2-t1))/(3412); // kwhr
printf("\t heat to be supplied : %.2f kwhr \n",Qs);
Qts=(Qs+Q1+Q2);
printf("\t total requirement : %.2f kwhr \n",Qts);
// end
```

Scilab code Exa 20.8.2 `Strip Heater for Air Heating`

```
printf("\t example 20.8b \n");
printf("\t approximate values are mentioned in the book \n");
m=100; // lb
t1=70;
t2=370;
L=4;
B=3;
n=4; // number of air changers
c1=0.12
Q1=(m*c1*(t2-t1));
printf("\t heat to steel charge : %.1e Btu \n",Q1);
c2=0.25
Q2=(n*L*B*2*0.075*c2*(t2-t1));
printf("\t heat to air : %.1e Btu \n",Q2);
printf("\t From Fig. 20.25a for 52ft^2 of oven outside surface and a temperature rise of 300F the loss is 5kw for 1 in.thick insulations.For 2 in.thick insulation the loss is 2.5kw \n");
Qt=((Q1+Q2)/(3412))+(2.5);
```

```
printf("\t total requirement : %.2f kw \n",Qt);
// end
```

Scilab code Exa 20.8.3 `Finned strip Heater`

```
printf("\t example 20.8c \n");
printf("\t approximate values are mentioned in the book \n");
m=270; // cfm
t1=70;
t2=120;
L=1.5; // ft
B=1.5; // ft
c=0.25
row=0.075; // lb/ft^3
Q=(m*row*60*c*(t2-t1));
printf("\t heat : %.2e Btu \n",Q);
V=(m/(L*B*60)); // fps
printf("\t velocity is : %.0f fps \n",V);
printf("\t Refer to Fig.20.22a.The air is capable of  removing 33watts/in which is the maximum dissipation which may be expected. Any group of heaters providing 5 kw which do not require a dissipation of more than 33 w/in. and which will fit into the duct will be satisfactory \n");
printf("\t Thus in Table 20.3 elements of 350 watts with a total length each of 18 in. each are satisfactory \n");
// end
```

Scilab code Exa 20.8.4 `Clamp on Plastic Heating`

```
printf("\t example 20.8d \n");
```

```
printf("\t approximate values are mentioned in the book \n");
t1=70;
t2=300;
L=26; // in
B=12; // in
H=1; // in
c1=0.13
// specific gravity of cast iron is 7.2
printf("\t unsteady state \n");
m=(L*B*H*62.5*7.2/1728); // lb
printf("\t weight of plate : %.0f lb \n",m);
Q1=(m*c1*(t2-t1));
printf("\t heat : %.1e Btu \n",Q1);
printf("\t From Figure 20.25b for a black body the radiation is 1.5w/in^2.The radiation from the top is actually 110 per cent of this value, and from the bottom of the plate it is 55 per cent for an average of 82.5 per cent is taken \n");
Q2=(2*26*12*1.5*0.825/1000); // ke
printf("\t radiation loss : %.1f kw \n",Q2);
Qt=((Q1)/(3412))+(Q2);
printf("\t total requirement : %.1f kw \n",Qt);
printf("\t staedy state \n");
m2=70;
c2=0.22;
Qs=(m2*c2*(t2-t1));
printf("\t heat : %.2e Btu \n",Qs);
Ql=0.8; // kw
Qts=((Qs)/(3412))+(Ql);
printf("\t total requirement : %.2f kw \n",Qts);
printf("\t The steady state is controlling.The requirements are satisfied , by four 24-in. strip heaters , but the sheath temperature must now be checked. Since the temperature drop per unit flux density is 14 to 19F, assume an average of 16.5 F . For clamp-on strips 24 in. long the watts per square inch deliverable are 16 \n");
```

```
delt=(16*16.5);
printf("\t delt is : %.0f F \n",delt);
printf("\t The sheath temperature is then 300 + 264
   = 564 F , which is satisfactory for steel
   sheathed elements with a 750F maximum. \n");
// end
```