Scilab Textbook Companion for Basic Environmental Technology - Water Supply, Waste Management And Pollution Control
by Jerry A. Nathanson[1]

Created by
Jitendra Singh
PhD
Others
IIT Bombay
College Teacher
None
Cross-Checked by
None

July 19, 2018

[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: Basic Environmental Technology - Water Supply, Waste Management And Pollution Control

Author: Jerry A. Nathanson

Publisher: Phi Learning

Edition: 5

Year: 2008

ISBN: 978-81-203-3836-4

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

Hydraulics

Scilab code Exa 2.1 1

```
clc; clear;
// Page 29
// Example 2.1
// Given data
ha=25; //depth at point A in ft
h=15; //height from the bottom in ft
// estimation
// Pressure at the tank bottom
Pa=0.43*ha; //psi, using equation 2-2b
hb=ha-h; // Depth of point B below water surface
// Pressure at point B
Pb=0.43*hb; //psi, using equation 2.2b
printf('Pressure at the point A = %f psi\n',Pa)
printf('Pressure at the point B = %f psi', Pb)
```

Scilab code Exa 2.2 2

```
clc; clear;
```

```
// Example 2.2
// Page 30
// Given data
ha=100-95; //height of water above point A in m
hb=100-70; //height of water above point B in m
hc=hb;  //The pressure at the pint C is equal to the
    pressure at B
hd=100-55; // height of water above point D in m
// estimation
Pa=9.8*ha; // kPa, using equation 2-2a
Pb=9.8*hb; // kPa
Pc=9.8*hc; // kPa
Pd=9.8*hd; // kPa
printf('The hydrostatic pressure at the point A = %f
    kPa\n',Pa)
printf('The hydrostatic pressure at the point B = %f
    kPa\n', Pb)
printf('The hydrostatic pressure at the point C = %f
    kPa\n', Pc)
printf('The hydrostatic pressure at the point D = %f
    kPa\n', Pd)
disp('The hydrostatic pressure at the point E can
    not be calculated.')
clear
```

Scilab code Exa 2.3 3

```
clc; clear;
// Example 2.3
// Page 31
// Given data
P=13; //pressure at tank in psi
// estimation
h=2.3*P // ft, using euqation 2-3b
printf('The pressure head at that point = %f ft\n',h
```

```
)
printf('The total depth of water in the tank = %f ft\n',h+5)
clear
```

Scilab code Exa 2.4 4

```
clc; clear;
// Example 2.4
// Page 31
// Given data
P= 30; // pressure at tank in kPa
// estimation
// part (a)
h=0.10*P; // m, using equation 2-3a
printf('The pressure head of water at tank bottom = %f m\n',h)
// part (b)
printf('The water would rise in vertical tube = %f m\n',h)
// part (c)
printf('The pressure in the trapped air = %f kPa\n',P-9.8)
clear
```

Scilab code Exa 2.5 5

```
clc; clear;
// Example 2.5
// Page 33
// Given data
f=50; // flow in m^3/s
// estimation
```

```
f1=f*1000; // in L/s
f2=f1*3600*24*(1/10^6); // in ML/d
printf('The converted flow = %f L/s\n',f1)
printf('The converted flow = %f ML/d',f2)
clear;
```

Scilab code Exa 2.6 6

```
clc; clear;
// Example 2.6
// Page 33
// Given data
f=50; // flow in ft^3
// estimation
f1=f*7.48*60; // flow in gpm
f2=f1*60*24; // flow in mgd
printf('The converted flow = %f gpm\n',f1)
printf('The converted flow = %f mgd\n',f2)
clear
```

Scilab code Exa 2.7 7

```
clc; clear;
// Example 2.7
// Page 34
// Given data
V=4.0; // average velocity in ft/s
D=18/12; // diameter in ft
// estimation
A=%pi*1.5^2/4; // flow area in ft^2
Q=A*V; // flow rate in cfs, using equation 2-4
printf('The flow rate = %f cfs\n',Q)
clear
```

Scilab code Exa 2.8 8

```
clc; clear;
// Example 2.8
// Page 34
// Given data
Q=50; // flow in ML/d
V=3.0; // velocity in m/s
// estimation
Q=Q*10^6*(1/24)*(1/3600)*(1/1000); // flow in m^3/s
A=Q/V; // flow area in m^2
D=(4*A/%pi)^(1/2); // diameter of pipe in m
printf('The diameter of pipe = %f m\n',D)
clear
```

Scilab code Exa 2.9 9

```
clc; clear;
// Example 2.9
// Page 35
// Given data
Q=1000; // flow in L/s
A1=0.50; // area at section 1 in m^2
A2=0.25; // arear at section 2 in m^2
// estimation
Q=Q/1000; // flow in m^3/s
V1=Q/A1; // velocity at section 1 in m/s
V2=Q/A2; // velocity at section 2 in m/s
printf('Velocity at section 1= %f m/s\n',V1)
printf('Velocity at section 2 = %f m/s',V2)
clear
```

Scilab code Exa 2.10 10

```
clc; clear;
// Example 2.9
// Page 35
// Given data
Da=1; // diameter at branch A in ft
Db=1/3; // diameter at branch B in ft
Dc=0.5; // diameter at branch C in ft
Va=1.0; // velocity at branch A in ft/s
Vb=5.0; // velocity at branch B in ft/s
// estimation
Aa=%pi*(Da)^2/4; // flow area at branch A in ft^2
Ab=%pi*(Db)^2/4; // flow area at branch B in ft^2
Ac=%pi*(Dc)^2/4; // flow area at branch C in ft^2
Qa=Aa*Va; // flow rate at branch A in ft^3/s
Qb=Ab*Vb; // flow rate at branch B in ft^3/s
Qc=Qa-Qb; // flow rate at branch C is diff of Qa &
    Qb, ft^3/s
Vc=Qc/Ac; // velocity at branch C in ft/s
printf('Velocity at branch C = %f ft/s',Vc)
clear
```

Scilab code Exa 2.11 11

```
clc; clear;
// Example 2.11
// Page 36
// Given data
D1=1; // diameter at section 1 in ft
D2=0.333; // diameter at section 2 in ft
Q1=2.0; // flow rate at section 1 in cfs
```

```
Q2=2.0; // flow rate at section 2 in cdf
P1=100; // pressure at section 1, psi
// estimation
A1=%pi*(D1)^2/4; // flow area at section 1 in ft^2
A2=%pi*(D2)^2/4; // flow area at section 2 in ft^2
V1=Q1/A1; // velocity at section 1, ft/s
V2=Q2/A2; // vel at section 2, ft/s
P1=100*144; // pressure at section 1, lb/ft^2
P2=((P1/62.4)+(V1^2/(2*32.2))-(V2^2/(2*32.2)))*(62.4/144); // pressure at section 2, psi, from equation 2-8b
printf('The pressure at section 2 = %f psi',P2)
clear
```

Scilab code Exa 2.12 12

```
clc; clear;
// Example 2.12
// Page 38
// Given data
hl=10; // head loss, ft per 1000 ft of pipeline
D=12; // diameter, inch
C=100; // from Hazen/Williams equation
// estimation
S=hl/1000; // unitless quantity
Q=0.28*C*D^2.63*S^0.54; // flow rate in pipe, from equation 2-9
printf('The flow rate in pipe = %f gpm',Q)
clear
```

Scilab code Exa 2.13 13

```
clc; clear;
```

```
// Example 2.13
// Page 38
// Given data
Q=1.0; // flow rate in cfs
D=8; // diameter in inch
L=1; //mile
//estimation
Q=Q*7.48*60; // flow rate, gpm
S=0.0064; // from nomograpch for given D and Q
    values
L=5280; // ft
hl=S*L; // ft
P=0.43*hl; // pressure drop, psi/mi, using equation
    2-2b
printf('The pressure drop P = %f psi/mi',P)
clear
```

Scilab code Exa 2.14 14

```
clc; clear;
// Example 2.14
// Page 39
// Given data
Q=30; // flow rate, L/s
P=10; // pressure drop, kPa
L=1; // in km
// estimation
hl=0.10*P; // m/km, using equation 2-3a
L=1000; // in m
S=hl/L; // unitless quantity
D=310; // pipe diameter, mm, using nomograph for
    given Q and S values
printf('The diameter of the pipe D = %f mm',D)
clear
```

Scilab code Exa 2.15 15

```
clc; clear;
// Example 2.15
// Page 39
// Given data
D1=100; //mm
D2=50; //mm
P=75; //kPa
C=0.98; // discharge coefficient
//estimation
D1=D1/1000; //m
D2=D2/1000; //m
A1=%pi*(D1)^2/4; //m^2
A2=%pi*(D2)^2/4; //m^2
Q=C*A2*((2*9.8*P/9.8)/(1-(A2/A1)^2))^(1/2); // flow
    rate, m^3/s, equation 2-10
printf('The flow rate in pipe Q = %f m^3/s',Q)
clear
```

Scilab code Exa 2.16 16

```
clc; clear;
// Example 2.16
// Page 41
// Given data
W=3; //width of channel, ft
D=1.5; // depth of water, ft
slope=0.5; //ft per 100 ft
//estimation
A=W*D; // flow area, ft^2
P=D+W+D; // wetted perimeter, ft
```

```
R=A/P; // hydrolic radius, ft
S=slope/100; // unitless quantity
Q=1.5/0.013*A*R^(2/3)*S^(1/2); // from Manning's
    formula
printf('The flow rate in pipe Q = %f cfs',Q)
clear
```

Scilab code Exa 2.17 17

```
clc; clear;
// Example 2.17
// Page 42
// Given data
D=12; // dia in inch
S=0.01; // unitless quantity
// estimation
D=D/12; /// dia in ft
A=%pi*(D)^2/4; // flow area, ft^2
R=D/4; // for circular pipe, Manning's formula
Q=1.5/0.013*A*R^(2/3)*S^(1/2); // from Manning's
    formula
V=Q/A; // flow velocity, ft/s
printf('The flow rate in pipe Q = %f cfs\n',Q)
printf('The flow velocity in pipe V = %f ft/s',V)
clear
```

Scilab code Exa 2.18 18

```
clc; clear;
// Example 2.18
// Page 42
// Given data
D=450; // dia in mm
```

```
S=0.02; // unitless quantity
// estimation
D=D/1000; /// dia in m
A=%pi*(D)^2/4; // flow area, ft^2
R=D/4; // for circular pipe, Manning's formula
Q=1.0/0.013*A*R^(2/3)*S^(1/2); // from Manning's formula
V=Q/A; // flow velocity, ft/s
printf('The discharge capacity of the pipe = %f m^3/s\n',Q)
printf('The flow velocity in pipe V = %f ft/s',V)
clear
```

Scilab code Exa 2.19 19

```
clc; clear;
// Example 2.19
// Page 44
// Given data
Q=500; // flow rate in L/s
S=0.25; // percent
// estimation
Q=Q*(1/10^3); // flow rate in m^3/s
S=S/100;
D=73; // required diameter, cm, from nomograph for given Q and S values
printf('The required diameter D = %f cm\n',D)
clear
```

Scilab code Exa 2.20 20

```
clc; clear;
// Example 2.20
```

```
// Page 44
// Given data
D=16; // dia in inch
Q=750; // flow rate , gpm
V=2; // velocity , ft/s
// estimation
S=0.00048; // from nomograph for givel Q and D vlaues
S=0.0014; // to maiantan the given velocity V of flow with given Q and D
printf('The required slope S = %f ',S)
clear
```

Scilab code Exa 2.21 21

```
clc; clear;
// Example 2.21
// Page 44
// Given data
D=300; // dia in mm
S=0.02; // slope
V=1.9; // full floe velocity , m/s , from nomograph
// estimation
Q=135; // flow rate , L/s , from a nomograph for a given D and S values
q=96; // partial flow , from nomograph
d=D*0.62; // for q/Q value from partial flow diagram
v=V*1.08; // partial flow velocity , m/s , from velocity curve
printf('The depth of water d = %f mm\n',d)
printf('The velocity of flow at depth d = %f m/s',v)
clear
```

Scilab code Exa 2.22 22

```
clc; clear;
// Example 2.22
// Page 45
// Given data
D=900; // pipe diameter, mm
S=0.1; // slope
// estimation
Q=550; // full flow capacity of pipe, L/s, from
   nomograpch
q=1.08*Q; // maximum discharge, L/s
printf('The maximum possible discharge q = %f L/s',q
   )
clear
```

Scilab code Exa 2.23 23

```
clc; clear;
// Example 2.23
// Page 45
// Given data
D=18; // pipe diameter, inch
S=1.6; // slope over 400 ft distance
d=6; //depth of flow, inch
// estimation
S=1.6/400; // slope
Q=2900; // flow rate in gpm from nomograph
V=3.8; // velocity, fps
depth_ratio=d/D;
q=0.22*Q; // from partial flow diagram, for a given
   depth_ratio values
v=0.82*V; // from partial flow velocity diagram, for
    a given depth_ration vlaue
printf('The discharge q = %f gpm\n',q)
```

```
printf('The velocity in pipe v = %f ft/s\n',v)
clear
```

Scilab code Exa 2.24 24

```
clc; clear;
// Example 2.24
// Page 45
// Given data
D=10; // the travel distance, m
T=20; // the travel time, second
w=1.5; // width of ditch, m
d=0.5; // depth of flow, m
// estimation
V=D/T; // flow velocity, m/s
A=w*d; // area of channel, m^2
Q=A*V; // the discharge, m^3/s
printf('The discharge Q = %f m^3/s\n',Q)
clear
```

Scilab code Exa 2.25 25

```
clc; clear;
// Example 2.25
// Page 46
// Given data
H=100; // head on the weir, mm
// estimation
H=H*(1/25.4)*(1/12); // head in ft
Q=2.5*H^2.5; // discharge, in ft^3/s, from equation
    2-12
Q=Q*28.32; // discharge in L/s
printf('The discharge Q = %f L/s\n',Q)
```

```
clear
```

Scilab code Exa 2.26 26

```
clc; clear;
// Example 2.26
// Page 47
// Given data
b=18; // base width of weir in inch
h=9; // head on the weir in inch
// estimation
b=b/12; // width of wier in ft
h=h/12; // head on the weir in ft
Q=3.4*b*h^1.5; // discharge over Cipoletti wier from
    equation 2-13
printf('The discharge Q = %f ft^3/s\n',Q)
clear
```

Chapter 3

Hydrology

Scilab code Exa 3.1 1

```
clc; clear;
// Example 3.1
// Page 61
// Given data
depth=25.0; // rainfall depth in mm
A=2.5; // Area in ha
//estimation
depth=depth/1000; // rainfall depth in m
A=A*10000; // area in m^2
V=depth*A; // total volume of water, from equation
    3-1
printf('The total volume of water V = %f m^3',V)
clear
```

Scilab code Exa 3.2 2

```
clc; clear;
// Example 3.2
```

```
// Page 61
// Given data
depth=25.0; // rainfall depth in mm
T=20; // duration of rainfall storm in minute
//estimation
I=depth/T*60; // rainfall intensity
printf('The rainfall intensity I = %f mm/h',I)
clear
```

Scilab code Exa 3.3 3

```
clc; clear;
// Example 3.3
// Page 62
// Given data
depth=4.0; // rainfall depth in inch
T=8.0; // duration of rainfall storm in h
A=120; // area , ac.
//estimation
I=depth/T; // rainfall intensity, inch/h
V=A*depth*(1/12); // volume of rain, acre-feet, from euqtion 3-1
V=V*43560; // volume of rain , ft^3
printf('The average rainfall intensity I = %f inch/h\n',I)
printf('The volume of rain V = %f ft^3',V)
clear
```

Scilab code Exa 3.4 4

```
clc; clear;
// Example 3.4
// Page 64
```

```
// Given data
depth=50.0; // rainfall depth in mm
T=40; // duration of rainfall in min
//estimation
I=depth/T*60; // rainfall intensity , mm/h
P=1/7.5; // probability of observing similar
    rainfall events , from figure 3.6
printf('The probability of observing similar
    rainfall events = %f',P)
clear
```

Scilab code Exa 3.5 5

```
clc; clear;
// Example 3.3
// Page 62
// Given data
RP=10.0; //return period of rainfaal events , year
T=60.0; // duration of rainfall in min
//estimation
A_cal=1520; // constant values for California , for
    RP=10-y
B_cal= 13; // constant value for western states , for
     RP=10-y
A_del=5840; // constant value for Delware , fro RP
    =10-y
B_del=29; // contant value for Middle Atlantic
    states , for RP=10-y
i_cal=A_cal/(T+B_cal); // expected rainfall
    intensity for Calfornia , mm/h
i_del=A_del/(T+B_del); // expected rainfall
    intensity for Delware , mm/h
printf('The expected rainfall intensity in Calfornia
     = %f mm/h\n',i_cal)
printf('The expected rainfall intensity in Delware =
```

```
    %f mm/h\n',i_del)
clear
```

Scilab code Exa 3.6 6

```
clc; clear;
// Example 3.6
// Page 66
// Given data
disp('This problem is completely theoritical')
clear
```

Scilab code Exa 3.7 7

```
clc; clear;
// Example 3.7
// Page 67
// Given data
RP1=5.0; // return period of first event, y
T1=24.0; // duration of first storm, hrs
RP2=100.0; // return period of second event, y
T2=20.0; // duration of second storm, min
//estimation
disp('5-year storn could could do more damage than
   the 100-year storm, from figure 3.8 and table 3.1
   ')
clear
```

Scilab code Exa 3.8 8

```
clc; clear;
// Example 3.8
// Page 72
// Given data
disp('Thus problem is completely theoritical and
    graphical')
clear
```

Scilab code Exa 3.9 9

```
clc; clear;
// Example 3.9
// Page 74
// Given data
disp('Thus problem is completely theoritical and
    graphical')
clear
```

Scilab code Exa 3.10 10

```
clc; clear;
// Example 3.10
// Page 80
// Given data
S=1.0; // slope of water table over a distance of
    200 m, in m
K=0.1; // coefficient of permeability , mm/s
//estimation
S=S/200; // slope of water table
V=K*S; // velocity of groundwater flow in aquifer ,
    mm/s
printf('The velocity of groundwater flow in aquifer
    V = %f mm/s',V)
```

```
clear
```

Chapter 4

Water Quality

Scilab code Exa 4.1 1

```
clc; clear;
// Example 4.1
// Page 97
// Given data
sol=500; // aqueous sal solution , mL
salt=125; // salt dissolved in solution , mg
//estimation
C_a=salt/sol*1000; // concentration of solution , in
    mg/L
C_b=C_a; // concentration of solution , in ppm
C_c=C_a*(1/17.1); // concentration of solution , in
    gpg
C_d=(salt/1000)/sol*100; // concentration of
    solution , in percent
C_e=C_a*8.34; // concentration of solution , in lb/
    mil gal
printf('The concentration of solution C= %f mg/L\n',
    C_a)
printf('The concentration of solution C= %f ppm\n',
    C_b)
printf('The concentration of solution C= %f gpg\n',
```

```
    C_c)
printf('The concentration of solution C= %f percent\
    n',C_d)
printf('The concentration of solution C= %f lb/mil
    gal',C_e)
clear
```

Scilab code Exa 4.2 2

```
clc; clear;
// Example 4.2
// Page 97
// Given data
w=8.0; // water, mil gal
C=0.2; // required concentration, mg/L
//estimation
C=C*8.34; // required concentration in lb/mil gal
chl=C*w; // chlorine, pounds
printf('The required qunatity of chlorine= %f lb',
    chl)
clear
```

Scilab code Exa 4.3 3

```
clc; clear;
// Example 4.3
// Page 104
// Given data
BOD5=180; // 5 days BOD mg/L
t=5; // time, d
k=0.1; // rate of BOD reaction
//estimation
BODL=BOD5/(1-10^(-k*t))
```

```
printf('The ultimate BOD = %f mg/L',BODL)
clear
```

Scilab code Exa 4.4 4

```
clc; clear;
// Example 4.4
// Page 105
// Given data
DO0=8.5; // initial DO, mg/L
DO5=5.0; // Do after 5 days, mg/L
V=6; // volume of sample, mL
t=5; // time, d
k=0.1; // rate of reaction
//estimation
BOD5=(DO0-DO5)*300/V; // BOD5 mg/L, from eq 4-3
BODL=BOD5/(1-10^(-k*t)); // ultimate BOD, mg/L, from
    eq 4-2
printf('The 5-days BOD = %f mg/L \n',BOD5)
printf('The ultimate BOD = %f mg/L',BODL)

clear
```

Scilab code Exa 4.5 5

```
clc; clear;
// Example 4.5
// Page 106
// Given data
A=40.545; // weight of dish plus residue, g
B=40.525; // weight of empty dish, g
V=100; //  volume of the sample, mL
//estimation
```

```
A=A*1000; // weight of dish plus residue , mg
B=B*1000; // weight of empty dish , mg
TDS=(A-B)*1000/V; // total dissolved solids , mg/L
printf('The TDS = %f mg/L',TDS)
clear
```

Scilab code Exa 4.6 6

```
clc; clear;
// Example 4.6
// Page 114
// Given data
N=16; // no of colonies in sample
V=4; // volume of sample
//estimation
C=N*100/V; // colifomrs per 100 mL, from eq 4−5
printf('The coliforms per 100 mL = %f',C)
clear
```

Chapter 5

Water Pollution

Scilab code Exa 5.1 1

```
clc; clear;
// Example 5.1
// Page 129
// Given data
cs=2; // original stream concentration , mg/L
Qs=40; // stream discharge , ML/d
cw=25; // waste concentration , mg/L
Qw=4.0; // waste discharge , ML/d
//estimation
cd=(cs*Qs+cw*Qw)/(Qs+Qw); // diluted 5 days BOD
   concen from eq 5-1, mg/L
printf('The diluted 5 days BOD= %f mg/L',cd)
clear
```

Scilab code Exa 5.2 2

```
clc; clear;
// Example 5.2
```

```
// Page 129
// Given data
cs=25; // temp, cel
Qs=100; // stream discharge, cfs
cw=65; // temp, cel
cd=27; // max temp, cel
//estimation
Qw=(cs-cd)*Qs/(cd-cw) // dischage of warm water,
    from eq 5-1, cfs
printf('The discharge of warm water = %f cfs',Qw)
clear
```

Scilab code Exa 5.3 3

```
clc; clear;
// Example 5.3
// Page 131
// Given data
// There is error with this example in the text book
    , 11-4.4=6.6 while, in the text book its given
    5.6
cs=3; // BOD con in stream, mg/L
DO_stream=9; // DO con, mg/L
Qs=15; // stream discharge, mgd
cw=50; // BOD con in effluent, mg/L
DO_sewage=2; // DO con in sewage, mg/L
Qw=5; // discharge with BOD, mgd
DO_sat=11; // saturated DO, mg/L
k1=0.2; // deoxygenation rate conctant, d^-1
k2=0.5; // reaction rate conctant, d^-1
//estimation
BODL=(cs*Qs+cw*Qw)/(Qs+Qw); // diluted 5 days BOD
    concen from eq 5-1, mg/L
DO=(DO_stream*Qs+DO_sewage*Qw)/(Qs+Qw); // DO con,
    mg/L, from eq 5-1
```

```
Di=DO_sat-DO; // initial oxygen deficit , mg/L

tc= (1/(k2-k1))*log10((k2/k1)*(1-Di*((k2-k1)/(k1*
    BODL)))); // time for critical oxygen dificit ,
    days
Dc=(k1*BODL/(k2-k1))*(10^(-k1*tc)-10^(-k2*tc))+ (Di
    *10^(-k2*tc)); // critical oxygen dificit , mg/L,
    eq. 5-3

min_DO=DO_sat-Dc; // minimum DO in stream , mg/L

printf('The minimum DO in the stream = %f mg/L',
    min_DO)
//clear
```

Chapter 6

Drinking Water Purification

Scilab code Exa 6.1 1

```
clc; clear;
// Example 6.1
// Page 157
// Given data
V=15000; // volume of tank, m^3
Q=120; // flow rate, ML/d
//estimation
V=V/1000; // Volume of tank, ML
TD=V/Q; // detection time, day
printf('The detention time = %f day',TD)
clear
```

Scilab code Exa 6.2 2

```
clc; clear;
// Example 6.2
// Page 157
// Given data
```

```
TD=3; // detention time, hour
Q=6; // flow rate, mgd
A=10000; // surface area of tank, ft^2
//estimation
Q=Q*10^6*(1/24); // floq rate, gal/h
V=TD*Q; // Volume of tank, gal
V=V/7.5; // volume of tank, ft^3
D=V/A; // depth of water in tank, ft
printf('The required volume of tank = %f ft^3\n',V)
printf('The depth of water in tank = %f ft',D)
clear
```

Scilab code Exa 6.3 3

```
clc; clear;
// Example 6.3
// Page 158
// Given data
Q=700; // over flow rate, gpd/ft^2
//estimation
V=Q*(1/7.5)*(1/24); // settling velocity, ft/h
printf('The settling velocity = %f ft/h',V)
clear
```

Scilab code Exa 6.4 4

```
clc; clear;
// Example 6.4
// Page 158
// Given data
TD=4; // detention time, hour
Q=6; // average flow rate, ML/d
Vo=20; // maximum overflow rate, m^3/m/^2-d
```

```
//estimation
Q=Q/24; // average flow rate, ML/h
V=Q*TD; // required volume, ML, eq 6-1
V=V*10^6; // required volume, L
V=V/1000; // required volume, m^3
Q=6*10^6/10^3; // average flow rate, m^3/d
As=Q/Vo; // surface area, m^2, eq 6-2
D=sqrt(4*As/%pi); // dia of tank, m
SWD=V/As; // SWD of tank, m
printf('The required dia of tank = %f m\n',D)
printf('The SWD of tank = %f m',SWD)
clear
```

Scilab code Exa 6.5 5

```
clc; clear;
// Example 6.5
// Page 166
// Given data
Q=0.25; // flow rate, mgd
A=50; // surface area of tank, ft^2
//estimation
Q=Q*10^6*(1/24)*(1/60); // flow rate, gal/min
R=Q/A; // filtration rate, gpm/ft^2
V=R/7.5; // velocity of flow, ft/min
printf('The filtration rate = %f gpm/ft^2\n',R)
printf('The velocity of flow = %f ft/min',V)
clear
```

Scilab code Exa 6.6 6

```
clc; clear;
// Example 6.6
```

```
// Page 166
// Given data
Q=6; // flow rate , ML/d
R=2.8; // filtration rate , L/m^2
//estimation
Q=Q*10^6*(1/24)*(1/3600); // flow rate , L/s
A=Q/R; // require surface area , m^2
S=sqrt(A); // side dimension , m
printf('The required surface area = %f m^2\n',A)
printf('The sie dimensin of box = %f m',S)
clear
```

Scilab code Exa 6.7 7

```
clc; clear;
// Example 6.7
// Page 167
// Given data
T=12; // backwash time , min
Q_backwash=10; // flow rate for backwash , L/m^2
Q=6; // flow rate , ML/d
A=25; // surface area , m^2
//estimation
V=Q_backwash*A*T*60/10^3; // volume of water used in
    backwash , m^3
Q=Q*10^3; // flow rate in m^3
P=V/Q*100;
printf('The percentage of water used in backwash =
   %f percent',P)
clear
```

Scilab code Exa 6.8 8

```
clc; clear;
// Example 6.8
// Page 170
// Given data
Q=7.5; // flow, mgd
chl=0.5; // chlorine dose, mg/L
d=1; // duration, month
//estimation
lbd=8.34*Q*chl; // lb/d, eq 6-3b
N=lbd*30/100; // no of cylinder required
N=ceil(N)
printf('The number of cylinder required = %f per
    month',N)
clear
```

Scilab code Exa 6.9 9

```
clc; clear;
// Example 6.9
// Page 170
// Given data
chl=15; // total chlorine, kg
V=50; // volume of water, ML
//estimation
C=chl/V; // chlorine dose, mg/L
printf('The chlorine dose = %f mg/L',C)
clear
```

Chapter 7

Water Distribution Systems

Scilab code Exa 7.1 1

```
clc; clear;
// Example 7.1
// Page 191
// Given data
L=300; // length of pipe, m
l=6; // length of pipe section, m
D=305; // diameter of pipe, mm
P=1000; // test pressure, kPa
OL=10; // observed leakage, L/h
//estimation
N=L/l; //  number of joints
QL=N*D*P^(1/2)/32600; // allowable leakage
if OL<QL then
    disp('the pipe is sufficiently watertight')

else
  disp('the pipe is not suffieciently watertight')
end
clear
```

Scilab code Exa 7.2 2

```
clc; clear;
// Example 7.2
// Page 191
// Given data
D=18; // diameter of pipe, inch
delta=90; // chnage in the direction of pipe, degress
P=80; // water pressure, lb/ft^2
S=3000; // bearig stress, lb/ft^2
//estimation
A=%pi*D^2/4; // area of pipe section, in^2
F=2*A*P*sin(delta*%pi/(180*2)); // statis thrust, lb, from eq 7-2
AT=F/S; // area of thrust, ft^2
printf('The static thrust = %f lb\n',F)
printf('The thrust area = %f ft^2',AT)
clear
```

Scilab code Exa 7.3 3

```
clc; clear;
// Example 7.3
// Page 195
// Given data
N1=2000; // impeller speed, rpm
N2=1500; // impeller speed, rpm
Q2=100; // discharge, L/s
H2=25; // discharge pressure, m
//estimation
```

```
Q1=Q2*(N1/N2); // expected pump discharge, L/s, eq 7-3
H1=H2*(N1^2/N2^2); // expected discharge pressure, m, eq 7-4
printf('The expected pump discharge = %f L/s\n',Q1)
printf('The expected discharge pressure = %f m',H1)
clear
```

Scilab code Exa 7.4 4

```
clc; clear;
// Example 7.4
// Page 197
// Given data
disp('this problem is completely graphical.')
clear
```

Scilab code Exa 7.5 5

```
clc; clear;
// Example 7.5
// Page 199
// Given data
disp('this problem is completely graphical.')
clear
```

Scilab code Exa 7.6 6

```
clc; clear;
// Example 7.6
```

```
// Page 200
// Given data
disp('this problem is completely graphical.')
clear
```

Scilab code Exa 7.7 7

```
clc; clear;
// Example 7.7
// Page 201
// Given data
Q=500; // discharge , L/s
TDH=25; // total dynamic head , m
P_in=150; // power to pump, kW
//estimation
Q=Q/1000; // discharge , m^3/s
P_out=9.8*Q*TDH; // power , kW
e=(P_out/P_in)*100; // efficiency of pump, percent
printf('The pump efficiency = %f percent',e)
clear
```

Scilab code Exa 7.8 8

```
clc; clear;
// Example 7.8
// Page 201
// Given data
Q=1500; // discharge , gpm
D=10; // diameter of pipe , inch
L=3000; // length , ft
h=100; // static head , ft
e=65; // efficiency of pump, percent
//estimation
```

```
S=0.024; // slope from nomograph for Q AND D
hL=S*L; // head loss , ft
TDH=h+hL; // total dynamic head , ft
P_out=(Q*TDH)/3960; // hp , eq 7−7
P_in=(P_out/e)*100; // required brake horsepower , hp
printf('The required brake horsepower = %f hp',P_in)
clear
```

Scilab code Exa 7.9 9

```
clc; clear;
// Example 7.9
// Page 201
// Given data
e=50; // efficiency , percent
P_in=150; // required power , kW
C=0.15; // power cost , USD
T=8; // duration , h
//estimation
P=P_in/e*100; // power consumption , kW
E=P*T; // energy consumed in 8 hrs , kW−h
C_op=E*C; // cost of operation , USD
printf('The cost of operation = %f USD',C_op)
clear
```

Scilab code Exa 7.10 10

```
clc; clear;
// Example 7.10
// Page 205
// Given data
Q=100; // flow rate , L/s
P=150; // pressure , kPa
```

```
D=250; // diameter , mm, from figure
L=2000; // length , m, from figure
S=0.024; // from Hazen−Williams nomograph
//estimation
hL=S*L; // head loss , m
P=0.1*P; // meter of pressure head
Ph=P+48; // pressure head , m, from figure 7.26
rP=9.8*Ph; // required pressure , kPa
printf('The required pressure = %f kPa',rP)
clear
```

Scilab code Exa 7.11 11

```
clc; clear;
// Example 7.11
// Page 207
// Given data
Q_AB=450; // flow rate in section AB, L/s
D_AB=8; // diameter in section AB, inch
S_AB=0.0064; // from Hazen−Williams nomograph , for
    section AB
L_AB=1500; // length of section AB, m, from figure

Q_BC=450; // flow rate in section BC, L/s
D_BC=12; // diameter of section BC, inch
S_BC=0.001; // from Hazen−Williams nomograph for
    section BC
L_BC=2500; // length of section BC, m, from figure
//estimation
hL_AB=S_AB*L_AB; // head loss in section AB, ft
hL_BC=S_BC*L_BC; // head loss in section BC, ft
HL=hL_AB+hL_BC; // total head loss from A to C, ft
S=HL/(L_AB+L_BC); // overall hydraulic gradient
D=9.5; // diameter for Q=450 and S=0.003, from
    monograph
```

```
printf('The equavalent diameter = %f inch',D)
clear
```

Scilab code Exa 7.12 12

```
clc; clear;
// Example 7.12
// Page 208
// Given data
hL=10; // head loss , m
L_AIB=500; // length of branch AIB, m
L_AIIB=1500; // length of branch AIIB, m
D_AIB=300; // diameter of branch AIB, mm
D_AIIB=200; // diameter of branch AIIB, mm
//estimation
S_AIB=hL/L_AIB; // hydraulic gradient on branch AIB,
Q_AIB=143; // flow in branch AIB for S_AIB=0.02 and
    D_AIB=300 mm, L/s
S_AIIB=hL/L_AIIB; // hydraulic gradient on branch
    AIIB,
Q_AIIB=27; // flow in branch AIB for S_AIIB=0.0067
    and D_AIB=200 mm, L/s
Q=Q_AIB+Q_AIIB; // total flow into junction A
S=hL/L_AIB; // overall gradient
D=320; // equivalent diameter for Q=170 L/s and S
    =0.02 from monograph, mm
printf('The equavalent diameter = %f mm',D)
clear
```

Scilab code Exa 7.13 13

```
clc; clear;
// Example 7.13
```

```
// Page 209
// Given data
disp('this problem is graphical and required tabular
    calculation')
clear
```

Scilab code Exa 7.14 14

```
clc; clear;
// Example 7.14
// Page 210
// Given data
disp('this problem is graphical and required tabular
    calculation')
clear
```

Chapter 8

Sanitary Sewer Systems

Scilab code Exa 8.1 1

```
clc; clear;
// Example 8.1
// Page 227
// Given data
// all input information provided in figure 8.8
printf('(a). Manhole B-4 is on the westmost end of
    line B, at station = %f \n',8+44)
printf('(b). The groud elevation at B-2 is = %f ft\n
    ',90.62)
printf('(c). The invert out elevation at B-3 is = %f
     ft\n',84.42)
printf('(d). The steepest slope between B-3 and B-4
   = %f \n',0.006)
printf('(e). The distance between manhole B-1 and
    line B = %f ft\n',275.78)
printf('(a). The depth of manhole B-3 = %f ft'
    ,92.44-84.47)
clear
```

Scilab code Exa 8.2 2

```
clc; clear;
// Example 8.2
// Page 229
// Given data
Q=400; // sewage flow, L/d
d=25; // population density, people/ha
A=100; // area, ha
//estimation
P=4*Q; // peak flow, L/d per person
DQ=A*d*P; // design flow rate for the reach, L/d
DQ=DQ/(24*3600); // design flow rate for the reach,
    L/s
printf(' The design flow rate for the reach = %f L/s
    ',DQ)
clear
```

Scilab code Exa 8.3 3

```
clc; clear;
// Example 8.3
// Page 231
// Given data
disp('This example is completely graphical.')
clear
```

Scilab code Exa 8.4 4

```
clc; clear;
// Example 8.4
// Page 232
// Given data
```

```
E1=1100; // elevation at manhole 1, ft
E2=1093; // elevation at manhole 2, ft
E3=1090; // elevation at manhole 3, ft
Q1=1; // design flow at reach 1, mgd
Q2=2; // design flow at reach 2, mgd
C=8; // minimum cover, ft
L1=300; // length of reach 1, ft
L2=440; // length of reach 2, ft
D1=8; // diameter of reach 1, inch, for Q1 and S1
D2=12; // diameter of reach 2, inch for S2 and Q2
//estimation
S1=(E1-E2)/L1; // ground slope in reach 1
Q1=(Q1*10^6)/(24*60); // desing flow ar reach 1, gal
   /min
UIE1=E1-D1-(D1/D2); // upper invert elevation in
   reach 1, ft
FS1=S1*L1; // fall of sever at reach 1, ft
LIE1=UIE1-FS1; // lower invert elevation at reach 1,
    ft

S2=(E2-E3)/L2; // ground slope in reach 2
SIE=LIE1+(D1/D2); // sum of invert elevation, ft
UIE2=SIE-1; // upper invert elevation in reach 2, ft
FS2=S2*L2; // fall of sever at reach 2, ft
LIE2=UIE2-FS2; // lower invert elevation at reach 2,
    ft
printf('The upper invert elevation at reach 1 = %f
   ft\n',UIE1)
printf('The lower invert elevation at reach 1 = %f
   ft\n',LIE1)
printf('The upper invert elevation at reach 2 = %f
   ft\n',UIE2)
printf('The lower invert elevation at reach 2 = %f
   ft',LIE2)
clear
```

Scilab code Exa 8.5 5

```
clc; clear;
// Example 8.5
// Page 239
// Given data
d=300; // diameter of pipe, mm
D=3; // depth of pipe, m
B=0.60; // trench width, m
w=18.8; // clay unit weight, kN/m^3
//estimation
H=D-(d/100); // pipe cover, m
H_B=H/B; // ratio pf cover to width
C=2.6; // coefficient from figure 8.19
W=C*w*B^2; // dead load due to backfill, kN/m
printf(' The dead load due to backfill = %f kN/m',W)
clear
```

Scilab code Exa 8.6 6

```
clc; clear;
// Example 8.6
// Page 239
// Given data
SSS=18; // safe supporting srength, kN/m
CS= 26.3; // crushing strength, kN/m
SF=1.5; // safety factor
//estimation
LF=(SF*SSS)/CS; // load factor
printf('The load factor = %f \n',LF)
disp(' according to load factor, the class D bedding
    will be adequate.')
```

```
clear
```

Scilab code Exa 8.7 7

```
clc; clear;
// Example 8.7
// Page 239
// Given data
H=11; // cover, ft
B=2; // trench width, ft
SF=1.5; // safety factor
//estimation
H_B=H/B; // ratio pf cover to width
C=3.2; // coefficient, from figure 8.19
w=130; // unit weight of backfill, lb/ft^3, from table 8.2
W=C*w*B^2; // dead load due to backfill, lb/ft, from Marston's formula
CS1=1400; // crushing strength for standard-strength 8-inch VCP, lb/ft
LF1=(SF*W)/CS1; // load factor, for standard strength VCP
CS2=2200; // crushing strength for extra-strength VCP, lb/ft
LF2=(SF*W)/CS2; // load factor, for extra strength VCP
printf('The load factor for standard strength VCP= %f',LF1)
disp(' according to load factor, the class B bedding will be adequate.')

printf('The load factor for extra strength VCP= %f',LF2)
disp(' according to load factor, the class C bedding will be adequate.')
```

```
clear
```

Scilab code Exa 8.8 8

```
clc; clear;
// Example 8.8
// Page 242
// Given data
I=45; // infiltration rate, L/d/mm/km
d=200; // diameter of lateral sewer, mm
L=1500; // length of sewer, m
//estimation
I=I*d*(L/1000); // infiltration, L/d
Q=19; // discharge, L/s, from manning's nomograph
Q=Q*3600*24; // discharge, L/d
RI=(I/Q)*100; // rate of infiltration, percent
printf('The rate of infiltration = %f percent',RI)
clear
```

Scilab code Exa 8.9 9

```
clc; clear;
// Example 8.9
// Page 243
// Given data
L=400; // sewer length, ft
Ds=12; // sewe diameter, inch
Dm=4; // manhole diameter, ft
d=2; // drop, inch in 1 hr
//estimation
A=(%pi*Dm^2)/4; // area, ft^2
d=d/12; // drop, ft
V=A*d; // volume of manhole, ft^3
```

```
V=V*7.5; // volume of manhole, gal
L=L/5280; // length of sewer, mi
R=V*24; // leackge rate, gal/d
ex=R/Ds/L; // exfiltration , gal/d/in/mi
printf(' The exfiltration = %f gal/d/in/mi',ex)
clear
```

Chapter 9

Stormwater Management

Scilab code Exa 9.1 1

```
clc; clear;
// Example 9.1
// Page 256
// Given data
A=5; // drainage basin area, ac
i=3; // rainfall intensity, inch/hr
C=0.4; // runoff coefficient
//estimation
Q=C*i*A; // peak rate, ft^3/s, from eq 9-1
printf('The peak rate of runoff = %f ft^3/s',Q)
clear
```

Scilab code Exa 9.2 2

```
clc; clear;
// Example 9.2
// Page 257
// Given data
```

```
A=2; // drainage basin area, ha
i=75; // rainfall intensity, mm/hr
C=0.4; // runoff coefficient
//estimation
A=A*10000; // drainage basin area, m^2
i=0.075; // rainfall intensity, m^2/h
Q=C*i*A; // peak rate, ft^3/s, from eq 9-1
printf('The peak rate of runoff = %f m^3/h',Q)
clear
```

Scilab code Exa 9.3 3

```
clc; clear;
// Example 9.3
// Page 258
// Given data
AT=15; // drainage basin area, ha
A1=6.5; // flat drainage area, ha
A2=6.0; // lawn drainage area, ha
A3=2.5; //paved drainage area, ha
C1=0.01; // runnoff coefficient for flat
C2=0.2; // runnoff coefficient for lawn
C3=0.95; // runnoff coefficient for paved
//estimation
C=1/(AT)*(A1*C1+A2*C2+A3*C3); // composite runoff coefficient, from eq 9-2 and table 9.1
printf('The composite runoff coefficient = %f ',C)
clear
```

Scilab code Exa 9.4 4

```
clc; clear;
// Example 9.2
```

```
// Page 257
// Given data
A=0.25; // drainage basin area , km^2
C=0.25; // runoff coefficient
d=150; // overland flow distance , in m
s=7; // slope in percent
//estimation
A=250000; // drainage basin area , m^2
i_5=0.070; // 5-year storm intensity for 30 min duration , m/h
i_100=0.15; // 100-year storm intensity for 30 min duration , m/h
Q_5=C*i_5*A; // peak rate for 5-year strom , m^3/h
Q_100=C*i_100*A; // peak rate for 5-year strom , m^3/h
printf('The peak rate of for 5-year storm = %f m^3/h \n',Q_5)
printf('The peak rate of for 100-year storm = %f m^3/h',Q_100)
clear
```

Scilab code Exa 9.5 5

```
clc; clear;
// Example 9.5
// Page 261
// Given data
A=1000; // drainage basin area , ac
Tc=66; // time of concentration , min
I=7.5; // rainfall intensity , inch
//estimation
R=4; // for I=7.5, from fig 9.5
Tc=Tc/60; // in hr
A=A/640; // drainage basin area , mi^2
q=300; // unit peak discharge , csm/in , for Tc=1.1
```

```
    from fig 9.6
Q=q*A*R; // peak rate of runoff, ft^3/s, from eq 9.3
printf('The peak rate of runoff = %f ft^3/s',Q)
clear
```

Scilab code Exa 9.6 6

```
clc; clear;
// Example 9.6
// Page 263
// Given data
A=0.25; // drainage basin area, km^2
C=0.6; // runoff coefficient
d=150; // overland flow distance, in m
s=7; // slope in percent
//estimation
A=250000; // drainage basin area, m^2
i_5=0.09; // 5-year storm intensity for 30 min
    duration, m/h
i_100=0.19; // 100-year storm intensity for 30 min
    duration, m/h
Q_5=C*i_5*A; // peak rate for 5-year strom, m^3/h
Q_100=C*i_100*A; // peak rate for 5-year strom, m^3/
    h
printf('The peak rate of for 5-year storm = %f m^3/h
    \n',Q_5)
printf('The peak rate of for 100-year storm = %f m
    ^3/h',Q_100)
clear
```

Scilab code Exa 9.7 7

```
clc; clear;
```

```
// Example 9.7
// Page 266
// Given data
A1=1; // drainage area of chatchment 1, ha
C1=0.4; // runoff coefficient for catchment 1
A2=1.5; // dainage area of catchment 2
C2=0.3; // runoff coefficient for catchment 2
A3=2; // dainage area of catchment 3
C3=0.2; // runoff coefficient for catchment 3
Tc=5; // time of concentration, min
S1=0.0035; // slope for catchment 1
S2=0.002; //slope for catchment 2
S3=0.0015; //slope for catchment 3
D2=120; // distance from catchment 1 to catchment 2
D3=180; // distance from catchment 2 to catchment 3
//estimation
//A1=1*10000; // drainage area of catchment 1, m^2
I1=0.15; // intensity, m/h for Tc=5
Q1=C1*I1*A1*10000; // peak flow in catchment 1, m^3/
    h
Q1=Q1/3600; // peak flow in catchment 1, m^3/s
D1=45; // diameter in catchment 1, in cm for Q1 and
    S1 from mannings monograph
V1=1.05; // velocity in catchment 1, in m/s from
    mannings monograph
C=1/(A1+A2)*(A1*C1+A2*C2); // composite runoff
    coefficient
T=D2/V1; // total time of flow to inlet 2
Tc=Tc+T; // time of concentration to inlet 2
I2=0.145; // intensity, m/h for given Tc
A2=A1+A2; // in m^2
Q2=C*I2*A2*10000; // peak flow at inlet 2, m^3/h
Q2=Q2/3600; // peak flow at inlet 2, m^3/s
D2=65; // diameter in catchment 2, in cm for Q2 and
    S2 from mannings monograph
V2=1.02; // velocity at inlet 2, in m/s from
    mannings monograph

```

```
C=1/(A1+A2+A3)*(A1*C1+A2*C2+A3*C3); // composite
    runoff coefficient
T=D3/V2; // total time of flow to inlet 3
Tc=Tc+T; // time of concentration to inlet 3
I3=0.135; // intensity , m/h for given Tc
A3=A2+A3; // in m^2
Q3=C*I3*A3*10000; // peak flow at inlet 3, m^3/h
Q3=Q3/3600; // peak flow at inlet 3, m^3/s
D3=80; // diameter in catchment 3, in cm for Q3 and
    S3 from mannings monograph
V3=1; // velocity at inlet 3, in m/s from mannings
    monograph
printf('The design flow at reach 1 = %f m^3/s\n',Q1)
printf('The design flow at reach 2 = %f m^3/s\n',Q2)
printf('The design flow at reach 3 = %f m^3/s\n',Q3)
printf('The required pipe diameter at reach 1 = %f
    cm\n',D1)
printf('The required pipe diameter at reach 2 = %f
    cm\n',D2)
printf('The required pipe diameter at reach 3 = %f
    cm',D3)
clear
```

Scilab code Exa 9.8 8

```
clc; clear;
// Example 9.8
// Page 270
// Given data
Tc=30; // time of concentration , min
Q=5; // peak runoff rate , m^3/s
A=10000; // basin detention , m^3
//estimation
Tc=Tc*60; // time of concentratio , s
TRV=1.5*Tc*Q; // total rainfall volume , m^3, from eq
```

```
    9-6
SF=A/TRV; // storage factor
FF=1-SF; // flow factor
P=FF*Q; // peak outflow , m^3/s
printf('The peak outflow rate = %f m^3/s',P)

clear
```

Scilab code Exa 9.9 9

```
clc; clear;
// Example 9.9
// Page 270
// Given data
P=100; //peak outflow , cfs
Pin=150; // peak inflow , cfs
TRV=300000; // total rainfall volume , ft^3
//estimation
FF=P/Pin; // flow factor
SF=1-FF; // storage factor
V=SF*TRV; // storage volume , ft^3
printf('The storage volume = %f ft^3',V)
clear
```

Scilab code Exa 9.10 10

```
clc; clear;
// Example 9.10
// Page 271
// Given data
Tc=15; // time of concentration , min
Pin_5=14000; //peak inflow for 5-year storm , m^3/h
Pout_5=4400; // peak outflow for 5-year storm , m^3/h
```

```
Pin_100=30000; //peak inflow for 100-year storm, m^3/h
Pout_100=9400; // peak outflow for 100-year storm, m^3/h
D=2; // depth of water, m
//estimation
FF=Pout_5/Pin_5; // flow factor for 5-year storm
SF=1-FF; // storage factor
Tc=Tc/60; // time of concentration, hr
TRV=1.5*Tc*Pin_5; // total rainfall volume, m^3
SV=SF*TRV; // storage volume, m^3
A_5=SV/D; // basin area for 5-year storm, m^2
// for 100-year storm
FF=Pout_100/Pin_100; // flow factor for 100-year storm
SF=1-FF; // storage factor
TRV=1.5*Tc*Pin_100; // total rainfall volume, m^3
SV=SF*TRV; // storage volume, m^3
A_100=SV/D; // basin area for 100-year storm, m^2
printf('The basin area required for 5-year storm = %f m^2\n',A_5)
printf('The basin area required for 100-year storm = %f m^2',A_100)
clear
```

Chapter 10

wastewater Treatment and Disposal

Scilab code Exa 10.1 1

```
clc; clear;
// Example 10.1
// Page 289
// Given data
P_in=200; // conc of BOD flowing in, mg/L
e=85; // treatment efficiency, %
Q=5; //flow rate, mgd
//estimation
P_out=P_in-((e*P_in)/100); // allowed maximum conc
    of BOD, mg/L, eq 10-1
BOD=8.34*Q*P_out; // discharged BOD, lb/d, eq 6-3b
printf('The maximum conc of BOD allowed = %f mg/L\n'
    ,P_out)
printf('The discharged BOD will be = %f lb/d',BOD)
clear
```

Scilab code Exa 10.2 2

```
clc; clear;
// Example 10.2
// Page 290
// Given data
P_in=250; // conc of BOD flowing in, mg/L
P_out=20; // conc of BOD flowing out, mg/L
Q=5; //flow rate, ML/D
//estimation
e=(P_in-P_out)/P_in*100; // treatment efficiency, %,
    eq 10-1
SS=Q*P_out; // suspended solid present in diacharge,
    kg


printf('The treatment efficiency = %f percent\n',e)
printf('The suspended solid present in discharge =
   %f kg',SS)
clear
```

Scilab code Exa 10.3 3

```
clc; clear;
// Example 10.3
// Page 293
// Given data
P_in=250; // conc of BOD flowing in, mg/L
e=60; // removal efficiency, %
//estimation
P_out=P_in-((e*P_in)/100); // expected average
   effluent TSS conc, mg/L, eq 10-1
printf('The expected average effluent TSS
   concentration = %f mg/L',P_out)
clear
```

Scilab code Exa 10.4 4

```
clc; clear;
// Example 10.4
// Page 295
// Given data
D=18; // dia of filter , m
d=2; // depth of filter , m
R= 1.5; // recirculation ratio
Q=2.5; // flow , ML/D
P_in=210; // conc of BOD flowing in , mg/L
e=30; // removal efficiency , %
//estimation
As=%pi*D^2/4; // surface area of filter , m^2
V=As*d; // volume , m^3
QR=R*Q; // recirculated flow , ML/D, eq 10-2
hl=(Q+QR)/As*10^3; // hydraulic load , m^/m^2-d
P_out=P_in-((e*P_in)/100); // conc of BOD flowing
    out , mg/L
OG=Q*P_out/V; // organic load , kg/m^3-d, eq 10-4a
printf('The hydraulic load on the filter = %f m^3/m
    ^2-d\n',hl)
printf('The organic load on the filter = %f kg/m^3-d
    ',OG)
clear
```

Scilab code Exa 10.5 5

```
clc; clear;
// Example 10.5
// Page 296
```

```
// Given data
P_in=200; // conc of BOD flowing in, mg/L
e1=79; // removal efficiency of trickling filter, %
e2=35; // removal efficiency of primary treatmet, %
//estimation
C1=((100-e2)*200)/100; // remaining BOD after
    primary treatment,mg/L
P_out=(100-e1)*C1/100; // conc of BOD flowing out
    after filter removal, mg/L
e=((P_in-P_out)/P_in)*100
printf('The overall plant efficiency = %f percent',e
    )
clear
```

Scilab code Exa 10.6 6

```
clc; clear;
// Example 10.6
// Page 297
// Given data
L=30; // length of sludge tank, m
W=10; // width of sludge tank, m
D=4; // depth of sludge tank, 3
P_in=200; // conc of BOD flowing in, mg/L
e2=35; // removal efficiency of primary treatmet, %
BOD=200; // BOD, mg/L
Q=4; // wastewater flow, ML/day
MLSS=2000; // MLSS conc, mg/L
//estimation
C1=((100-e2)*200)/100; // remaining BOD after
    primary treatment,mg/L
V=W*L*D; // volume of tank, m^3
V=V/10^3; // volume of tank, ML
F_M=(Q*C1)/(MLSS*V); // F/M ratio, eq 10-5
printf('The food-to-microorganism (F/M) ratio = %f',
```

```
    F_M)
clear
```

Scilab code Exa 10.7 7

```
clc; clear;
// Example 10.7
// Page 298
// Given data
Q=800000; // flow , gpd
BOD=125; // BOD, ppm
F_M=0.4; // food-to-microorganism (F/M) ratio
MLSS=1800; // MLSS conc , ppm
SWD=15; // side wwater depth , ft
//estimation
V=(Q*BOD)/(F_M*MLSS); // volume , gal , eq 10-5
V=V/7.5; // volume , ft^3;
W=(V/(3*SWD))^(1/2);
L=3*W; // required length of tank , ft
printf('The required length of tank = %f ft',L)
clear
```

Scilab code Exa 10.8 8

```
clc; clear;
// Example 10.8
// Page 298
// Given data
V=150; // sludge volume , mL
MLSS=2000; // MLSS conc , mg/L
//estimation
SVI=V*1000/MLSS; // sludge volume index , mL/g, from
    eq 10-6
```

```
printf('The sludge volume index SVI = %f mL/g',SVI)
clear
```

Scilab code Exa 10.9 9

```
clc; clear;
// Example 10.9
// Page 317
// Given data
T=10; // time interval, min
d=3/4; // drop at steady state, inch
//estimation
PR=T/d; // perc rate, min/inch
printf('The perc rate = %f min/inch',PR)
clear
```

Scilab code Exa 10.10 10

```
clc; clear;
// Example 10.10
// Page 321
// Given data
NB=4; // number of bedroom
PR=30; // perc rate, min/inch
Tr=2; // width of trench, ft
AA=250; // absorption area required for per bedroom,
    from Fig 10.34
//estimation
TA=NB*AA; // total area required for four bedroom
T=TA/Tr; // required trench, ft
L=T/100; // number of lateral
WAF=(L-1)*6; // wodth of absorption field
```

```
LF=WAF*100; // the area will be occupy by leaching
    field , ft^2
printf('The dimension of leaching filed = %f ft^2',
    LF)
clear
```

Scilab code Exa 10.11 11

```
clc; clear;
// Example 10.11
// Page 327
// Given data
S1=6; // sludge solids conc , percent
V1=300; // volume , m^3
V2=200, // volume after dewatering , m^3
//estimation
WC=100-S1; // water content of the sludge , percent
V1=V1*1000; // volume , L
M=S1*V1/100; //mass of sludge solids , kg of dry
    solids . from eq 10-7
V2=V2*1000; // volume after dewatering , L
S2=(M/V2)*100; // sludge solids conc after
    dewatering , percent solids , from eq 10-7
WC1=100-S2; // water content after dewatering ,
    percent
printf('The water content = %f percent\n',WC)
printf('The mass of dry solids = %f kg of dry solids
    \n',M)
printf('The sludge solids conc after dewatering = %f
    percent\n',S2)
printf('The water content dewatering = %f percent\n'
    ,WC1)
clear
```

Scilab code Exa 10.12 12

```
clc; clear;
// Example 10.12
// Page 327
// Given data
V=5000000; // volume of sludge, gal
W=1000000; // dry sludge solids,lb
C=4; // conc of solids in sludge, percent
//estimation
S=(W/(8.34*V))*100; // solids content in sludge,
    percent
V=(W/(C*8.34))*100; // volume of sludge, gal
printf('The solid content in sludge = %f percent\n',
    S)
printf('The volume of sludge = %f gal\n',V)
clear
```

Scilab code Exa 10.13 13

```
clc; clear;
// Example 10.13
// Page 328
// Given data
E=0.5; // TSS removal efficiency, in decimal
TSS=240; // raw TSS, mg/L
Q=4; // Flow, ML/d
E_BOD=30; // removal efficiency of BOD, percent
bod=220; //raw BOD
K=0.25; //proportion of BOD convertred, from
    equation 10-9a
// estimation
```

```
m1=E*TSS*Q; // mass of primary sludge solids , kg/d,
    from equation 10−8a
BOD=(1-E_BOD/100)*bod; // BOD applied to the
    secondary system , mg/L
m2=K*BOD*Q; // mass of secondary sludge , kg/d, from
    equation , 10−9a
M=m1+m2; // total mass of combind primary and
    secondary sludge , kg/d
V=M/Q*100; // total volume of sludge , L/d, from
    equation 10−7a
V=V/10^3; // total volume of sludge , m^3/d

printf('Mass of primary sludge solids = %f kg/d\n',
    m1)
printf('Mass of secondary sludge solids = %f kg/d\n'
    , m2)
printf('Total volume of sludge solids = %f m^3/d', V
    )
clear
```

Chapter 11

Municipal Solid Waste

Scilab code Exa 11.1 1

```
clc; clear;
// Example 11.1
// Page 355
// Given data
Vi=15; // initial volume, m^3
Vf=3; // final volume, m^3
r=90; // required reduction in percent
//estimation part a
PVR=(Vi-Vf)/Vi*100; // percent volume reduction, in
    precent, from eq 11-1
CR=Vi/Vf; // compaction ratio, from eq 11-2
// part b
Vf=Vi-(r*Vi/100); // final volume, m^3
CR_b=Vi/Vf; // compaction ratio ofr part b
printf('The percent volume reduction = %f percent\n'
    ,PVR)
printf('The compaction ration for part a= %f \n',CR)
printf('The compaction ration for part b= %f',CR_b)
clear
```

Scilab code Exa 11.2 2

```
clc; clear;
// Example 11.2
// Page 369
// Given data
P=15000; // total population
R=5; // rate of waste generation per person per day, lb
A=25; // landfill area, ac
D=20; // depth of refuse, ft
f=25; // addtional volume of cpver material, percent
w=1000; //  unit weight of refuse, lb/yd^3
//estimation
W=R*365*P; // total weight of refuse  generated per year, lb/year
V=W/w; // total yearly volume of refuse, yd^3/yr
delta_V=0.25*V; // additional volume for cover material, yd^3/yr
TV=V+delta_V; // total lanfill volume required, yd^3/yr
AV=A*43560*D*(1/27); // available volume for landill, yd^3
L=AV/TV; // useful life of landfill, year
printf('The useful life of landfill= %f year',L)
clear
```

Scilab code Exa 11.3 3

```
clc; clear;
// Example 11.3
// Page 369
```

```
// Given data
L=30; // design life of landfill , in year
R=25; // waste generation rate , N/person/d
w=5; // waste compact unit weight , kN/m^3
D=10; // average depth , m
P=50000; // community population
C=25; // cover ratio , percent of volume for cover
//estimation
W=R*365*P; // total weight of refuse  generated per year , N/year
W=W/1000; // total weight of refuse  generated per year , kN/year
V=W/w; // total yearly volume of compacted refuse , m^3/yr
delta_V=(C/100)*V; // additional volume for cover material , yd^3/yr
TV=V+delta_V; // total required volume required , m^3/yr
A=TV/D; // required area , m^2/yr
A=A*(1/10000)*30; // required area for landill , m^2
printf('The required area for landfill = %f year',A)
clear
```

Chapter 13

Air Pollution and Control

Scilab code Exa 13.1 1

```
clc; clear;
// Example 13.1
// Page 440
// Given data
cs=0.12; // hourly ozone concentration , mg/L
T=25; // tempreature , degree celcius
P=1; // pressure , atm
AT=48; //atomic weight of ozone
//estimation
T=T+273; //tempreature in kelvin
CD=(273*cs*AT*P)/(22.4*T); // Ozone concentration in
    mg/m^3 (from equation 13.1)
CD=CD*1000; //  Ozone concentration in microg/m^3
printf('ozone concentration = %f microgram/m^3',CD)
clear
```

Scilab code Exa 13.2 2

```
clc; clear;
// Example 13.2
// Page 440
// Given data
CD=365; // sulfer dioxide concentration, microgram/m^3
T=25; // tempreature, degree celcius
P=1; // pressure, atm
AT=64; //atomic weight of osulfer dioxide

//estimation
CD=365/1000; // sulfer dioxide concentration, in ppm
T=T+273; //tempreature in kelvin
cs=(CD*22.4*T)/(273*AT*P); // sulfer dioxide in ppm (from euation 13.1)
cs_perc=(cs/10^6)*100; //sulfer dioxide concentration in percentage
printf('sulfer dioxide concentration = %f ppm\n',cs)
printf('sulfer dioxide concentration = %f percentage',cs_perc)
clear
```

Scilab code Exa 13.3 3

```
clc; clear;
// Example 13.3
// Page 440
// Given data
cs_perc=2.2; // carbon monooxide percentage
T=82; // tempreature, degree celcius
P=1; // pressure, atm
AT=28; //atomic weight of osulfer dioxide

//estimation
CD_ppm=cs_perc*1*(10^6)/100; // sulfer dioxide
```

```
concentration , in ppm
T=T+273; //tempreature in kelvin
CD=(273*CD_ppm*AT*P)/(22.4*T); // carbon monooxide in mg/m^3 (from euation 13.1)
//CD_ppm=CD/1000 // carbon monooxide in ppm

printf('carbon monooxide concentration = %f ppm\n',CD_ppm)
printf('carbon monooxide concentration  = %f mg/m^3',CD)
clear
```

Scilab code Exa 13.4 4

```
clc; clear;
// Example 13.4
// Page 441
// Given data
M=120; // mass of dustfall bucket when empty
m=120.30; //mass of dustfall bucket after exposure
d= 6; // diameter of bucket in inch
D=0.5; //diameter in feet
E=1; // exposure is 1 month

//estimation
A=3.14*(D^2)/4; // area of the bucket in square feet
M_diff= (m-M); // mass of particulates
D_unit1=(((M_diff)/E)*(1/454)*(1/2000))/(A*(1/5280)^2); //dustfall in tons/square mi/month
D_unit2=(((M_diff)/E)*(1/1000))/(A*(0.305^2)*(1/10000)); //dustfall in kg/ha/month
printf('dustfall  = %f tons/square mi/month\n',D_unit1)
printf('dustfall = %f kg/ha/month',D_unit2)
clear
```

Scilab code Exa 13.5 5

```
clc; clear;
// Example 13.5
// Page 441
// Given data
AF_ini= 55; // initial airflow in ft^3/min
AF_aft= 35; // after sampling in ft^3/min
T= 24; // duration in hours
m= 10; // filter weight before sampling in gram
M= 10.2; //filter weight after sampling in gram

//estimation
M_diff=M-m; //weight of particulates in gram
AF_avg=(AF_ini+AF_aft)/2; // average airflow
T=24*60; //duration in minutes
V=AF_avg*T; // air volume in cubic feet
V_m3= V*0.02832; // air volume in cubic meter
M_microgram= M_diff*(10^6); //weight of particulates
    in microgram
TSP=M_microgram/V_m3; // total suspended particulate
    concentration in microgram/m^3
printf('total suspended particulate concentration =
    %f microgram/m^3\',TSP)
clear
```

Scilab code Exa 13.6 6

```
clc; clear;
// Example 13.6
// Page 452
```

```
// Given data

A1= 5000; // total collector plate area in square
    meter
F=150; // flow rate in cubic meter per second
V=0.1; // drift velocity in m/s
A2=7500; // area in m^2
A3= 10000; // area in m^2

//estimation
E_5000=100*(1-(exp(-V*A1/F))); // efficieny
    corresponds to area 5000 square meter from
    equation 13.2
E_7500=100*(1-exp(-V*A2/F)); // efficieny
    corresponds to area 7500 square meter
E_10000=100*(1-exp(-V*A3/F)); // efficieny
    corresponds to area 10000 square meter
printf('Efficiency corresponds to area 5000m^2 = %f
    percentage\n',E_5000)
printf('Efficiency corresponds to area 7500m^2 = %f
    percentage\n',E_7500)
printf('Efficiency corresponds to area 10000m^2 = %f
     percentage\n',E_10000)
clear
```

Chapter 14

Noise Pollution and Control

Scilab code Exa 14.1 1

```
clc; clear;
// Example 14.1
// Page 465
// Given data
D=5; // distance , in km
s=344; // speed of sound waves , m/s
//estimation
D=D*1000; // distance in m
T=D/s; // time in s
printf('Required time to hear = %f second',T)
clear
```

Scilab code Exa 14.2 2

```
clc; clear;
// Example 14.2
// Page 465
// Given data
```

```
f=500; // frequency of sound, in Hz
v=5000; // speed of sound waves, m/s
//estimation
lambda=v/f; // wavelength of sound, m, from eq 14-1
printf('The wavelength of sound = %f meter',lambda)
clear
```

Scilab code Exa 14.3 3

```
clc; clear;
// Example 14.3
// Page 466
// Given data
P=200; // sound pressure, mubar
P0=0.0002; // reference pressure, mubar
//estimation
SPL=20*log10(P/P0); // sound pressure level, dB,
    from eq 14-2

printf('The sound pressure level = %f dB',SPL)
clear
```

Scilab code Exa 14.4 4

```
clc; clear;
// Example 14.4
// Page 467
// Given data
SPL=90; // sound pressure level for each dozers, in
    dB
//estimation
d=SPL-SPL; // numerical difference between two SPL
    values
```

```
SPL1=SPL+3; // for d=0, from Figure 14.4
d1=SPL1-SPL; // numerical difference between two SPL
    values
SPL2=SPL1+1.7; // for d1=3, from figure 14.4
d2=SPL2-SPL; // numerical difference between two SPL
    values
SPL3=SPL2+1.3; // final sound pressure level, for d2
   =4.7, from figure 14.4
printf('The sound pressulre level = %f dB',SPL3)
clear
```

Scilab code Exa 14.5 5

```
clc; clear;
// Example 14.5
// Page 465
// Given data
dBA=[72 76 79 81 84 76 75 75 74]; // sound readings,
    dBA
T=10:10:90; // time interval, in min
//estimation
[dBA1 idx]=gsort(dBA);
idx_T=find(T==50);
idx1=find(dBA1>dBA1(idx_T));
SPL=dBA1(length(idx1)); // sound pressure level
   exceed 50% of time
L10=dBA1(find(T==10)); // L10 value in dB
L90=dBA1(find(T==90)); // L90 value in dB
printf('sound pressure level exceed 50 percent of
   time = %f dB \n',SPL)
printf('L10 = %f dB\n',L10)
printf('L90 = %f dB',L90)
clear
```

Scilab code Exa 14.6 6

```
clc; clear;
// Example 14.6
// Page 474
// Given data
Sa=85; // sound level from centerline , in dBA
Db=12; // distance , in m
Da=4; // distance of sound measurement , in m
SL_79=79; // sound level form centerline , in dBA
//estimation
SL12=Sa-(10*log10(Db/Da)); // sound level ate distance 12 m
D=Da*(10^((SL_79-Sa)/(-10))); // distance for sound level 79 dBA
printf('The sound level ate distance 12 m = %f dBA \n',SL12)
printf('The distance from the road centerline for sound level 79 dBA= %f meter',D)
clear
```