

Scilab Textbook Companion for
Principles Of Geotechnical Engineering
by B. M. Das¹

Created by
Madhusudhan Reddy
C++
Civil Engineering
IIT BOMBAY
College Teacher
D. B. Phatak
Cross-Checked by
Chaitanya

January 15, 2015

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

Book Description

Title: Principles Of Geotechnical Engineering

Author: B. M. Das

Publisher: Cenage & learning, New Delhi

Edition: 7

Year: 2013

ISBN: 978-1133108665

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	5
2 weight volume relationships	8
3 weight volume relationships	9
4 plasticity and structure of soil	12
6 soil compaction	13
7 permeability	15
8 seepage	21
9 in situ stresses	25
10 stress in a soil mass	28
11 Compressibility of Soil	30
12 Shear Strength of Soil	37
13 Lateral Earth Pressure	41
14 Lateral Earth Pressure Curved Failure Surface	48
15 Slope Stability	50
16 Soil Bearing Capacity for Shallow Foundations	56

List of Scilab Codes

Exa 2.2	solved	8
Exa 3.2	solved	9
Exa 3.3	solved	10
Exa 3.4	solved	10
Exa 3.5	solved	11
Exa 4.1	solved	12
Exa 6.2	solved	13
Exa 6.3	solved	13
Exa 6.4	solved	14
Exa 7.1	solved	15
Exa 7.2	solved	15
Exa 7.3	solved	16
Exa 7.4	solved	16
Exa 7.5	solved	17
Exa 7.6	solved	17
Exa 7.8	solved	18
Exa 7.9	solved	18
Exa 7.10	solved	18
Exa 7.11	solved	19
Exa 7.12	solved	19
Exa 8.1	solved	21
Exa 8.2	solved	21
Exa 8.3	solved	22
Exa 8.4	solved	23
Exa 8.5	solved	23
Exa 9.1	solved	25
Exa 9.2	solved	26
Exa 9.3	solved	26

Exa 9.4	solved	27
Exa 10.1	solved	28
Exa 10.3	solved	28
Exa 11.1	solved	30
Exa 11.2	solved	31
Exa 11.3	solved	31
Exa 11.4	solved	32
Exa 11.5	solved	32
Exa 11.6	solved	33
Exa 11.7	solved	33
Exa 11.8	solved	34
Exa 11.9	solved	34
Exa 11.10	solved	34
Exa 11.11	solved	35
Exa 11.12	solved	35
Exa 11.13	solved	36
Exa 12.1	solved	37
Exa 12.2	solved	37
Exa 12.3	solved	38
Exa 12.4	solved	38
Exa 12.5	solved	39
Exa 12.6	solved	39
Exa 12.7	solved	40
Exa 12.8	solved	40
Exa 13.1	solved	41
Exa 13.2	solved	42
Exa 13.3	solved	43
Exa 13.4	solved	43
Exa 13.5	solved	44
Exa 13.6	solved	45
Exa 13.7	solved	45
Exa 13.9	solved	46
Exa 13.10	solved	46
Exa 14.1	solved	48
Exa 14.2	solved	49
Exa 15.1	solved	50
Exa 15.2	solved	50
Exa 15.3	solved	51

Exa 15.4	solved	51
Exa 15.5	solved	52
Exa 15.6	solved	52
Exa 15.7	solved	53
Exa 15.8	solved	53
Exa 15.9	solved	54
Exa 15.11	solved	54
Exa 16.1	solved	56
Exa 16.2	solved	57
Exa 16.3	solved	57
Exa 16.4	solved	58
Exa 16.5	solved	59
Exa 16.6	solved	59
Exa 16.7	solved	60
Exa 18.1	solved	61
Exa 18.2	solved	61
Exa 18.3	solved	62

Chapter 2

weight volume relationships

Scilab code Exa 2.2 solved

```
1 clc
2 //initialisation of variables
3 //from graph
4 d= 0.15 //mm
5 w= 0.17 //mm
6 a= 0.27 //mm
7 //calculations
8 C= a/d
9 c= w^2/(a*d)
10 //results
11 printf ('uniformity coefficient = % f ',C)
12 printf ('coefficient of gradation = % 2f ',c)
```

Chapter 3

weight volume relationships

Scilab code Exa 3.2 solved

```
1  clc
2  //initialisation of variables
3  V= 1.2 //m^3
4  M= 2350 //Kg
5  w= 0.086
6  G= 2.71
7  W= 1000 //kg/m^3
8  //calculations
9  R= M/V
10 D= M/((1+w)*V)
11 e= (G*W/D)-1
12 n= e/(e+1)
13 S= (w*G/e)*100
14 v= (M-(M/(1+w)))/W
15 //results
16 printf ('moist density = % f kg/m^3 ',R)
17 printf ('dry density = % f kg/m^3 ',D)
18 printf ('void ratio = % 3f ',e)
19 printf ('porosity = % 3f ',n)
20 printf ('Degree of saturation = % 3f ',S)
21 printf ('volume of water in soil sample = % 3f m^3 '
```

,v)

Scilab code Exa 3.3 solved

```
1 clc
2 //initialisation of variables
3 n= 0.4
4 G= 2.68
5 w= 0.12
6 R= 1000 //kg/m^3
7 V= 10 //m^3
8 //calculations
9 d= G*R*(1-n)*(1+w)
10 s= ((1-n)*G+n)*R
11 M= s-d
12 m= M*V
13 //results
14 printf ('mass of water to be added for full
    saturation = % f kg ',m)
```

Scilab code Exa 3.4 solved

```
1 clc
2 //initialisation of variables
3 d= 16.19 //kN/m^3
4 w= 0.23
5 W= 9.81 //kN/m^3
6 //calculations
7 R= d*(1+w)
8 G= d/(W-d*w)
9 e= w*G
10 //results
11 printf ('saturated unit weight = % 2f kN/m^3 ',R)
```

```
12 printf ('specific gravity = % 2f ',G)
13 printf ('void ratio = % 2f ',e)
```

Scilab code Exa 3.5 solved

```
1 clc
2 //initialisation of variables
3 G= 2.68
4 w= 0.12
5 d= 1794.4 //kg/m^3
6 W= 1000 //kg/m^3
7 emax= 0.75
8 emin= 0.4
9 //calculation
10 e= (G*W*(1+w)/d)-1
11 D= ((emax-e)/(emax-emin))*100
12 //results
13 printf ('relative density of compaction in
percentage = % f ',D)
```

Chapter 4

plasticity and structure of soil

Scilab code Exa 4.1 solved

```
1 clc
2 //initialisation of variables
3 V1= 24.6 //cm^3
4 V2= 15.9 //cm^3
5 M1= 44 //g
6 M2= 30.1 //g
7 W= 1 //g/cm^3
8 //calculations
9 SL= (((M1-M2)/M2)*100) - (((V1-V2)/M2)*W*100)
10 //results
11 printf ('shrinkage limit of the soil = % 2f ',SL)
```

Chapter 6

soil compaction

Scilab code Exa 6.2 solved

```
1 clc
2 //initialisation of variables
3 G= 2.6
4 LL= 20
5 P= 20
6 //calclations
7 R= (4804574*G-195.55*(LL)^2+156971*(P)^0.5-9527830)
      ^0.5
8 n= (1.195e-4)*((LL)^2)-1.964*G-(6.617e-5)*(P)+7.651
9 w= %e^n
10 //results
11 printf ('maximum dry density = % f kg/m^3 ',R)
12 printf ('optimum moisture content = % 2f ',w)
```

Scilab code Exa 6.3 solved

```
1 clc
2 //initialisation of variables
```

```

3 do= 1570 //kg/m^3
4 mo= 0.545 //kg
5 M1= 7.59 //kg
6 M2= 4.78 //kg
7 M3= 3.007 //kg
8 w= 0.102 //
9 dmax= 19 //KN/m^3
10 //calculations
11 Ms= M1-M2
12 Mc= Ms-mo
13 Vh= Mc/do
14 Dc= M3/Vh
15 Du= Dc*9.81/1000
16 f= Du/(1+w)
17 Rc= f*100/dmax
18 //results
19 printf ('dry unit weight of compaction in the field
    = % 2f kN/m^3 ',f)
20 printf ('relative compaction in the field = % f ',Rc
    )

```

Scilab code Exa 6.4 solved

```

1 clc
2 //initialisation of variables
3 D1= 0.36 //mm
4 D2= 0.52 //mm
5 D5= 1.42 //mm
6 //calculations
7 Sn= 1.7*(sqrt((3/(D5)^2)+(1/(D2)^2)+(1/(D1)^2)))
8 //results
9 printf ('sustainabilty number = % f ',Sn)

```

Chapter 7

permiability

Scilab code Exa 7.1 solved

```
1 clc
2 //initialisation of variables
3 L= 30 //cm
4 A= 177 //cm^2
5 h= 50 //cm
6 Q= 350 //cm^3
7 t= 300 //sec
8 //claculations
9 k=Q*L/(A*h*t)
10 //results
11 printf ('hydraulic conductivity = %3f cm/sec ',k)
```

Scilab code Exa 7.2 solved

```
1 clc
2 //initialisation of variables
3 L= 203 //mm
4 A= 10.3 //cm^2
```



```

5 a= 0.39 //cm^2
6 h0= 508 //mm
7 h180= 305 //mm
8 t= 180 //sec
9 //calculations
10 k= 2.303*a*L*log10(h0/h180)/(A*t)
11 //results
12 printf ('hydraulic conductivity of sand = % 2f in/
        sec ',k)

```

Scilab code Exa 7.3 solved

```

1 clc
2 //initialisation of variables
3 k= 3e-7 //cm/sec
4 n= 0.0911e-4 //g*sec/cm^2
5 dw= 1 //g/cc
6 //calculations
7 K= k*n/dw
8 //results
9 printf ('absolute permeability = % 4f cm^2 ',K)

```

Scilab code Exa 7.4 solved

```

1 clc
2 //initialisation of variables
3 k= 5.3e-5 //m/sec
4 H= 3 //m
5 a= 0.139 //radians
6 //calculations
7 A= H*cos(a)
8 i= sin(a)
9 q= k*i*A*3600

```

```
10 //results
11 printf ('rate of seepage = % 4f m^3/hr/m ',q)
```

Scilab code Exa 7.5 solved

```
1 clc
2 //initialisation of variables
3 L= 50 //m
4 k= 0.08e-2//m/sec
5 h= 4 //m
6 H1= 3 //m
7 H= 8 //m
8 a= 0.139 //radians
9 //calculations
10 i= h*cos(a)/L
11 A= H1*cos(a)
12 q= k*i*A
13 //results
14 printf ('flow rate = % 2f m^3/sec/m ',q)
```

Scilab code Exa 7.6 solved

```
1 clc
2 //initialisation of variables
3 k1= 0.02 //cm/sec
4 e1= 0.5
5 e2= 0.65
6 //calculations
7 k2= k1*(e2^3/(1+e2))/(e1^3/(1+e1))
8 //results
9 printf ('hydraulic conductivity at void ratio of
10 0.65 = % 2f cm/sec ',k2)
```

Scilab code Exa 7.8 solved

```
1 clc
2 //initialisation of variables
3 e= 0.6
4 D10= 0.09 //mm
5 //calculations
6 k= 2.4622*(D10^2*(e^3/(1+e)))^0.7825
7 //results
8 printf ('hydraulic conductivity = % 4f cm/sec ',k)
```

Scilab code Exa 7.9 solved

```
1 clc
2 //initialisation of variables
3 e= 0.6
4 D10= 0.09 //mm
5 D60= 0.16 //mm
6 //calculations
7 Cu=D60/D10
8 k= 35*(e^3/(1+e))*(Cu^0.6)*(D10^2.32)
9 //results
10 printf ('hydraulic conductivity = % 3f cm/sec ',k)
```

Scilab code Exa 7.10 solved

```
1 clc
2 //initialisation of variables
3 k1= 0.302e-7 //cm/sec
```

```

4 k2= 0.12e-7 //cm/sec
5 e1= 1.1
6 e2= 0.9
7 e= 0.75
8 //calculations
9 n= (log10((k1/k2)*((1+e1)/(1+e2))))/log10(e1/e2)
10 C= k1/(e1^n/(1+e1))
11 k= C*(e^n/(1+e))
12 //results
13 printf ('hydraulic conductivity = %e cm/sec ',k')

```

Scilab code Exa 7.11 solved

```

1 clc
2 //initialisation of variables
3 H1= 2 //m
4 H2= 3 //m
5 H3= 4 //m
6 k1= 1e-4 //cm/sec
7 k2= 3.2e-2 //cm/sec
8 k3= 4.1e-5 //cm/sec
9 //calculations
10 H= H1+H2+H3
11 Kh= (1/H)*((k1*H1)+(k2*H2)+(k3*H3))
12 Kv= H/((H1/k1)+(H2/k2)+(H3/k3))
13 P= Kh/Kv
14 //results
15 printf ('ration of equivalent hydraulic conductivity
        = % 2f ',P)

```

Scilab code Exa 7.12 solved

```

1 clc

```

```
2 //initialisation of variables
3 H= 450 //mm
4 h= 150 //mm
5 k1= 1e-2 //cm/sec
6 k2= 3e-3 //cm/sec
7 k3= 4.9e-4 //cm/sec
8 h1= 300 //mm
9 //calculations
10 Kv= H/(h*(1/k1+1/k2+1/k3))
11 i= h1/H
12 q= Kv*i*100*3600
13 //results
14 printf ('rate of water supply = %2f cm/hr ',q)
```

Chapter 8

seepage

Scilab code Exa 8.1 solved

```
1  clc
2  //initialisation of variables
3  H1= 12 //in
4  H2= 20 //in
5  z= 8 //in
6  h1= 24 //in
7  h= 20 //in
8  k1= 0.026 //in/sec
9  D= 3 //in
10 //calculations
11 k2= H2*k1/((z/(1-h/h1))-H1)
12 i= h1/(H1+H2)
13 A= %pi/4*D^2
14 keq= (H1+H2)/((H1/k1)+(H2/k2))
15 q= keq*A*i*3600
16 //results
17 printf ('rate of water flow = %2f in^3/hr ',q)
```

Scilab code Exa 8.2 solved

```

1  clc
2  //initialisation of variables
3  Nd= 6
4  H1= 5.6 //m
5  H2= 2.2 //m
6  k= 5e-5 //cm/sec
7  dL= 4.1 //m
8  //calculations
9  H= (H1-H2)/Nd
10 h1= 5.61-H
11 h2= 5.61-5*H
12 q= 2.38*(H1-H2)*k/Nd
13 i= H/dL
14 //results
15 printf ('at point a, water will rise to height of = %
        3f m ',h1)
16 printf ('at point b, water will rise to height of = %
        3f m ',h2)
17 printf ('total rate of seepage per unit length = %e
        m^3/sec/m ',q)
18 printf ('average hydraulic gradient at c = % 3f ',i)

```

Scilab code Exa 8.3 solved

```

1  clc
2  //initialisation of variables
3  k1= 5.67 //ft/day
4  k2= 11.34 //ft/day
5  //from graph
6  Nd= 8
7  Nf= 2.5
8  H= 20
9  //calculations
10 q= sqrt(k1*k2)*H*Nf/Nd
11 //results

```

```
12 printf ( 'average rate of flow = % 2f ft^3/day/ft ',q
    )
```

Scilab code Exa 8.4 solved

```
1 clc
2 //initialisation of variables
3 B= 6 //m
4 L= 120 //m
5 s= 3 //m
6 T= 6 //m
7 x= 2.4 //m
8 H= 5 //m
9 k= 0.008 //cm/sec
10 //calculations
11 b=B/2
12 a1= b/T
13 a2= s/T
14 a3= x/b
15 Q= 0.378*k*H*L*36*24
16 //results
17 printf ( 'seepage under the dam = % 2f m^3/day ',Q)
```

Scilab code Exa 8.5 solved

```
1 clc
2 //initialisation of variables
3 b= %pi/4 //degrees
4 a= %pi/6. //degrees
5 B= 10 //ft
6 H= 20 //ft
7 h= 25 //ft
8 k= 2e-4 //ft/min
```



```
9 //calculations
10 r= H/tan(b)
11 d= 0.3*r+(h-H)/tan(b)+B+h/tan(a)
12 L= d/cos(a)-sqrt((d/cos(a))^2-(H/sin(a))^2)
13 q= k*L*tan(a)*sin(a)*24*60
14 //results
15 printf ('seepage rate = % 4f ft^3/day/ft ',q)
```

Chapter 9

in situ stresses

Scilab code Exa 9.1 solved

```
1  clc
2  //initialisation of variables
3  Ds= 16.5 //kN/m3
4  S= 19.25 //kN/m3
5  g= 9.8 //kN/m3
6  h1= 6 //m
7  h2= 13 //m
8  //at point A
9  Sa= 0
10  Ua= 0
11  Sa1= 0
12  //at point B
13  Sb= h1*Ds
14  Ub= 0
15  Sb1= Sb-Ub
16  //at point C
17  Sc= h1*Ds+h2*S
18  Uc= h2*g
19  Sc1= Sc-Uc
20  //results
21  printf ('total pressure at C= %2f kN/m3 ',Sc)
```

```

22 printf ('pore water pressure at C = % 2f kN/m^3 ',Uc
    )
23 printf ('effective stress at point C= % 2f kN/m^3 ',
    Sc1)

```

Scilab code Exa 9.2 solved

```

1  clc
2  //initialisation of variables
3  h= 20 //ft
4  g= 120 //kg/ft^3
5  h1= 12 //ft
6  w= 62.4 //kg/ft^3
7  //calculations
8  H= h-(h1*w/g)
9  //results
10 printf ('maximu depth that can be made in clay = % 2
    f ft ',H)

```

Scilab code Exa 9.3 solved

```

1  clc
2  //initialisation of variables
3  G= 2.68
4  e= 0.52
5  g= 9.81 //kN/m^3
6  h1= 0.7 //m
7  h2= 1 //m
8  h3= 1.5 //m
9  h4= 2 //m
10 //calculations
11 //for soil A
12 sa= (G+e)*g/(1+e)

```

```

13 //point a
14 Sa= h1*g+h2*sa
15 u= (h2+h1+h3/2)*g
16 Es= Sa-u
17 //point b
18 sb= h1*g+h4*sa
19 ub= (h4+h1+h3)*g
20 Eb= sb-ub
21 i= h3/2
22 s= i*g
23 //results
24 printf ('effective stress at point a= % 2f kN/m^2 ',
        Es)
25 printf ('effective stress at point b= % 2f kN/m^2 ',
        Eb)
26 printf ('seepage force per unit voume = % 2f kN/m^3
        ',s)

```

Scilab code Exa 9.4 solved

```

1 clc
2 //initialisation of variables
3 C0= 0.357
4 H1= 30.5 //ft
5 H2= 5 //ft
6 w= 62.4 // lb/ft^3
7 D= 20
8 g= 112 // lb/ft^3
9 //calculations
10 G= g-w
11 FS= D*G/(C0*w*(H1-H2))
12 //results
13 printf ('safety factor = % 2f ',FS)

```

Chapter 10

stress in a soil mass

Scilab code Exa 10.1 solved

```
1  clc
2  //initialisation of variables
3  sx= 2000 //lb/ft^3
4  sy= 2500 //lb/ft^3
5  T= 800 //lb/ft^3
6  t= 0.348//radians
7  //calculations
8  s1= (sx+sy)/2+sqrt(((sy-sx)/2)^2+T^2)
9  s2= (sx+sy)/2-sqrt(((sy-sx)/2)^2+T^2)
10 sn= (sx+sy)/2+(sy-sx)*cos(2*t)/2-T*sin(2*t)
11 Tn= (sy-sx)*sin(2*t)/2+T*cos(2*t)
12 //results
13 printf ('principle stress s1 = % 2f lb/ft^3 ',s1)
14 printf ('principle stress s2 = % 2f lb/ft^3 ',s2)
15 printf ('normal stress = % 2f lb/ft^3 ',sn)
16 printf ('shear stress = % 2f lb/ft^3 ',Tn)
```

Scilab code Exa 10.3 solved

```
1 clc
2 //initialisation of variables
3 x= 3 //m
4 y= 4 //m
5 P= 5 //kN
6 z= 2 //m
7 //calculations
8 r= sqrt(x^2+y^2)
9 k= r/z
10 I= 3/(2*%pi*((r/z)^2+1)^2.5)
11 s= P*I/z^2
12 //results
13 printf ('verticle stress increase at 2m = %4f kN/m
    ^3 ',s)
```

Chapter 11

Compressibility of Soil

Scilab code Exa 11.1 solved

```
1  clc
2  Tz=150
3  b=1
4  l=2
5  z=5*b
6  Es= (10000*2 + 8000*1 +12000*2)/5
7  a=4
8  H=z
9  m=1/b
10 n=2*H/b
11 F1=0.641 //from tables 11.1 and 11.2
12 F2=0.031
13 u=0.3
14 Is= F1 + ((2-u)/(1-u))*F2
15 If=0.71 //from table 11.3
16 Sef= Tz *a*b/l *(1-u^2)*Is*If/Es
17 Ser=0.93*Sef
18 printf('The elastic settlement at the centre of
        foundation = %f m',Ser)
```

Scilab code Exa 11.2 solved

```
1 clc
2 // one value of e is done
3 Gs=2.75
4 A=30.68
5 Ms=128
6 p=1
7 Hs=Ms/(A*Gs*p)
8 H=2.540
9 Hv=H-Hs
10 e=Hv/Hs
11 printf('the value of e for give values = %f',e)
```

Scilab code Exa 11.3 solved

```
1 clc
2 e11=0.9
3 e21=0.8
4 T2=4
5 T1=2
6 Cc= (e11-e21)/log10(T2/T1) // from loading branch
7 e1=0.67
8 e2=0.655
9 Cs=(e1-e2)/log10(T2/T1)
10 k=Cs/Cc
11 T3=12
12 e3=e11-Cc*log10(T3/T1)
13 printf('Compression index Cc= %f\n',Cc)
14 printf(' Cs/Cc = %f\n',k)
15 printf(' e3 = %f',e3)
```

Scilab code Exa 11.4 solved

```
1  clc
2  Gd=14
3  Gss=18
4  Gsc=19
5  Gw=9.81
6  To= 2*Gd+4*(Gss-Gw)+2*(Gsc-Gw)
7  LL=40
8  Cc=0.009*(LL-10)
9  H=4
10 T=100
11 e=0.8
12 Sc= Cc*H*log10((To+T)/To)/(1+e)
13 printf('a)Primary Consolidation Sc = %f m\n',Sc)
14
15 Tc=190
16 Cs=Cc/6
17 Sc= Cs*H*log10((To+T)/To)/(1+e)
18 printf(' b)Primary Consolidation Sc = %f m\n',Sc)
19
20 Tc=170
21 Sc= Cc*H*log10((To+T)/Tc)/(1+e)+ Cs*H*log10(Tc/To)
    /(1+e)
22 printf(' c)Primary Consolidation Sc = %f m\n',Sc)
```

Scilab code Exa 11.5 solved

```
1  clc
2  Gs=18
3  Gw=9.81
4  H=10
```

```

5 eo=1.1
6 To=5*(Gs-Gw)
7 T1=48
8 T=To+T1
9 e1=1.045 // void ratio corresponding to T
10 e=eo-e1
11 Sc=H*e/(1+eo)
12 printf('The settlement in the field Sc = %f m',Sc)

```

Scilab code Exa 11.6 solved

```

1 clc
2 T=8.5
3 eo=0.8
4 Cc=0.28
5 To=2650
6 T1=970
7 C1=0.02
8 t2=5
9 t1=1.5
10 H=8.5*12
11 epr=Cc*log10((To+T1)/To)
12 ep=eo-epr
13 C2=C1/(1+ep)
14 Sc=epr*H/(1+eo)
15 Ss=C2*H*log10(t2/t1)
16 TS=Sc+Ss
17 printf('Total consolidation settlement of the clay =
%f in ',TS)

```

Scilab code Exa 11.7 solved

```

1 clc

```

```

2 //T50 = Cvlab /H2 lab = Cvfield?H2 fiels
3 t1=140
4 Hf=3
5 Hd=0.025/2
6 tf=t1*Hf2/Hd2
7 k=tf/(3600*24)
8 printf('t field = %f days',k)

```

Scilab code Exa 11.8 solved

```

1 clc
2 //Tv is directly proportional to U2
3 t1=93.333
4 U2=30
5 U1=50
6 t2=t1*U22/U12
7 printf('t2 = %f days',t2)

```

Scilab code Exa 11.9 solved

```

1 clc
2 t90=75*24*60*60 // time in sec
3 T90=0.848
4 Hd=1.5*100 //in cm
5 Cv=T90*Hd2/t90
6 printf('Cv = %f cm2/sec',Cv)

```

Scilab code Exa 11.10 solved

```

1 clc

```

```

2 To=3000 // lb/ft^2
3 eo=1.1
4 e1=0.9
5 e=eo-e1
6 ea=(eo+e1)/2
7 T1=3000 // lb/ft^2
8 T=1 // in
9 t = 2 // min
10 m=(e/T1)/(1+ea)
11 U=50
12 Tv=0.197
13 Gw=62.4 //lb/ft^3
14 Cv=Tv*(T/(2*12)^2)/t
15 k=Cv*m*Gw *10^7
16 printf('a)k = %f x10^-7 ft/min\n',k)
17
18
19 U=60
20 Tv=0.286
21 H=6
22 t60=Tv*H^2/(Cv*60*24)
23 printf(' b)t60 =%f days ',t60)

```

Scilab code Exa 11.11 solved

```

1 clc
2 t50=19
3 Hd=2.24/2
4 Cv=0.197*Hd^2/t50
5 printf('Cv = %f cm^2/min ',Cv)

```

Scilab code Exa 11.12 solved

```

1  clc
2  LL=40
3  Cc=0.009*(LL-10)
4  H=10*12
5  eo=1.0
6  Gss=120
7  Gsc=110
8  Gd=100
9  To=10*Gd +10*(Gss-62.4)+10*(Gsc-62.4)/2
10
11 Tt=0.408
12 Tm=0.232
13 Tb=0.019
14 Tav= (Tt+4*Tm+Tb)/6
15 Sc=Cc*H*log10((To+Tav*1000)/To)/(1+eo)
16 printf('Sc = %f in ',Sc)

```

Scilab code Exa 11.13 solved

```

1  clc
2  H = 6
3  Cc = 0.28
4  eo = 0.9
5  Cv = 0.36
6  To=210
7  Tp=115
8  Sc= Cc*H*log10((To+Tp)/To)/(1+eo)
9  t2=9
10 Hd=3
11 Tv=Cv*t2/Hd^2
12 U=0.67
13 Tf=0.677*Tp
14 printf('Tf = %f kN/m^2 ',Tf)

```

Chapter 12

Shear Strength of Soil

Scilab code Exa 12.1 solved

```
1  clc
2  D=50 // in mm
3  A= %pi/4 *(D/1000)^2
4  // solving for test 1
5  N=150
6  Sp=157.5
7  Sr=44.2
8  Tf=Sp/A
9  Tr=Sr/A
10 // from graph
11 k=tand(27)
12 k1=tand(14.6)
13
14 printf('Peak strength Tf = 40+ t*%f\n',k )
15 printf(' Residual strength Tr = t*%f',k1)
```

Scilab code Exa 12.2 solved

```

1 clc
2 T3=16 // lb/in^2
3 Tf=25 // lb/in^2
4 T1=T3+Tf
5 a= asind((T1-T3)/(T1+T3)) // Mohr's circle
6 printf('a)Angle of friction ,a = %f\n',a)
7 b= 45+ a/2
8 printf(' b)Angle b that the failure plane makes with
    the major principal plane = %f',b)

```

Scilab code Exa 12.3 solved

```

1 clc
2 T1=41
3 T3=16
4 a=58
5 T=(T1+T3)/2 + (T1-T3)*cosd(2*a)/2
6 tf=(T1-T3)*sind(2*a)/2
7 printf('a)the normal stress T = %f lb/in^2',T)
8 printf(' and the shear stress tf = %f lb/in^2',tf)

```

Scilab code Exa 12.4 solved

```

1 clc
2 //For normally consolidated clay , c' = 0.
3 a=30
4 T3=10
5 T1=T3*(tand(45+a/2))^2
6 Tf=T1-T3
7 printf('The deviator stress at failure = %f lb/in^2',
    ,Tf)

```

Scilab code Exa 12.5 solved

```
1  clc
2  T13=70
3  T1f=130
4  T11=T13+T1f
5
6  T23=160
7  T2f=223.5
8  T21=T23+T2f
9
10 a= 2*(atand(((T11-T21)/(T13-T23))^0.5)-45)
11 c= (T11-T13*(tand(45+a/2))^2)/(2*tand(45+a/2))
12 printf('the shear strength parameter c = %f kN/m^2 ',
        c)
```

Scilab code Exa 12.6 solved

```
1  clc
2  T3=12
3  Tf=9.1
4  T1=T3+Tf
5  u=6.8
6  a= asind((T1-T3)/(T1+T3))
7
8  a1= asind((T1-T3)/(T1+T3-2*u))
9
10 printf('a) Consolidated-undrained angle of shearing
        resistance = %f degrees\n',a)
11 printf(' b) Drained friction angle = %f degrees ',a1)
```

Scilab code Exa 12.7 solved

```
1 clc
2 T3=12
3 a=27.8
4 T1=T3*(tand(45+a/2))^2
5 Tf=T1-T3
6 printf('the deviator stress at failure = %f lb/in^2',
    Tf)
```

Scilab code Exa 12.8 solved

```
1 clc
2 PI=28
3 OCR=3.2
4 To=160
5 Kn=0.11+0.0037*PI
6 Ko=OCR^0.8 * Kn
7 Cu=Ko*To
8 printf('the average undrained shear strength of the
    clay = %f kN/m^2',Cu)
```

Chapter 13

Lateral Earth Pressure

Scilab code Exa 13.1 solved

```
1  clc
2  OCR=2
3  a=30
4  Ko=(1-sind(a))*(OCR)^sind(a)
5  //at z=0
6  To1=0
7  Th1=0
8  u1=0
9  //at z=10
10 To2=10*100
11 Th2=Ko*To2
12 u2=0
13 //at z=15
14 To3= 10*100+5*(122.4-62.4)
15 Th3=Ko*To3
16 u3=5*62.4
17 //Lateral force Po =Area 1 + Area 2+ Area3+ Area 4
18 Po =(1/2)*10*707+5*707+(1/2)*5*212.1+(1/2)*5*312
19 z=((3535)*(5+10/3)+3535*(5/2)+530.3*(5/3)+780*(5/3))
    /Po
20 printf('z = %f ft ',z)
```

Scilab code Exa 13.2 solved

```
1  clc
2  //c=0
3  a=36
4  G=16
5  Ka=(1-sind(a))/(1+sind(a))
6  //at z=0 Tp=0
7  z=6
8  To=G*z
9  Ta=Ka*To
10 Pa=z*Ta/2
11
12 printf('a)Rankine active force per unit length of
    the wall = %f kN/m',Pa)
13 printf(' and the location of the resultant is z = 2m
    \n')
14
15
16 p=36
17 G=16
18 Kp=(1+sind(a))/(1-sind(a))
19 //at z=0 Tp=0
20 z=6
21 To=G*z
22 Tp=Kp*To
23 Pp=z*Tp/2
24
25 printf(' b)Rankine passive force per unit length of
    the wall = %f kN/m',Pp)
26 printf(' and the location of the resultant is z = 2m
    ')
```

Scilab code Exa 13.3 solved

```
1  clc
2  H=12
3  a=20
4  b=20
5  G=115
6  c=30
7  Oa= asind(sind(a)/sind(c))-a+2*b
8  Ka= (cosd(a-b)*sqrt(1+(sind(c))^2-2*sind(c)*cosd(Oa)
      ))/((cosd(b))^2*(cosd(a)+sqrt((sind(c))^2-(sind(a)
      ))^2)))
9  Pa=G*H^2*Ka/2
10 B= atand((sind(c)*sind(Oa))/(1-(sind(c)*cosd(Oa))))
11 printf('The active force Pa per unit length of the
      wall = %f lb/ft\n',Pa)
12 printf(' The resultant will act a distance of 12/3
      = 4 ft above the bottom of the wall with B = %f
      degree ',B)
```

Scilab code Exa 13.4 solved

```
1  clc
2  a=30
3  Ka1=(1-sind(a))/(1+sind(a))
4  a=35
5  Ka2=(1-sind(a))/(1+sind(a))
6  //at z=0 so T0=0
7  //atz=3
8  To=3*16
9  Ta1=Ka1*To
10 Ta2=Ka2*To
```

```

11
12 // At z=6
13 To=3*16+3*(18-9.81)
14 Ta2=Ka2*To
15
16 Pa =(1/2)*3*16+3*13.0+ (1/2)*3*36.1
17 z= (24 *(3+3/3)+39.0*(3/2)+54.15*(3/3))/Pa
18 printf('The force per unit length of the wall = %f
        kN/m\n',Pa)
19 printf (' The location of the resultant = %f m ',z)

```

Scilab code Exa 13.5 solved

```

1  clc
2  Ka= (tand(1))^2
3  G=16.5
4  cu=10
5  H=6
6  //at z=0
7  z=0
8  Ta=G*z-2*cu
9  //zt z=6
10 z=6
11 Ta=G*z-2*cu
12
13 zo=2*cu/G
14 // Before the tensile crack occurs
15 Pa= G*H^2/2 - 2*cu*H
16 printf('Pa before the tensile crack occurs = %f kN/m
        \n',Pa)
17 //After the tensile crack occurs
18 Pa=(H-zo)*Ta/2
19 printf(' Pa after the tensile crack occurs = %f kN/m
        ',Pa)

```

Scilab code Exa 13.6 solved

```
1  clc
2  H=15
3  a=10
4  G=118
5  b=20
6  C=250
7  Zo=2*C*sqrt((1+sind(b))/(1-sind(b)))/G
8  //at z=0 Ta=0
9  //at z=15
10 z=15
11 K=0.3
12 Ta=G*z*K*cosd(a)
13 Pa=(H -Zo)*Ta/2
14 printf('The Rankine active force Pa on the retaining
        wall after the tensile crack occurs = %f lb/ft ',
        Pa)
```

Scilab code Exa 13.7 solved

```
1  clc
2  c=30
3  b=15
4  a=10
5  Ka=0.3872 // from table 13.8
6  H=4
7  G=15
8  Pa=G*H^2*Ka/2
9  printf('The active force per unit length Pa = %f kN/
        m\n', Pa)
```

```
10 printf(' The resultant will act at a vertical
    distance equal to  $H/3 = 4/3 = 1.33$  m above \n the
    bottom of the wall and will be inclined at an
    angle of 15to the back face of the wall.')
```

Scilab code Exa 13.9 solved

```
1  clc
2  kh=0.2
3  kv=0
4  H=4
5  a=0
6  b=0
7  c=15
8  d=30
9  G=15.5
10 B= atand(kh/(1-kv))
11 b1=b+B
12 a1=a+B
13 Ka=0.452
14 Pa=G*H^2*Ka/2
15 Pae=Pa*(1-kv)*((cosd(b1))^2/((cosd(b))^2*(cosd(B))^2))
16 Ka=0.3014
17 Pa=G*H^2*Ka/2
18 P1=Pae-Pa
19 z= ((Pa*H/3)+P1*0.6*H)/Pae
20 printf('Pae = %f kN/m\n',Pae)
21 printf(' Z = %f m',z)
```

Scilab code Exa 13.10 solved

```
1  clc
```

```
2 H=28
3 C=210
4 b=10
5 G=118
6 c=20
7 kh=0.1
8 Ka=tand(45-c/2)
9 zo=2*C/(G*(Ka))
10 n=zo/(H-zo)
11 Nac=1.60
12 Nav=0.375
13 L=1.17
14 Pae= G*(H-zo)^2*(L*Nav)-C*(H-zo)*Nac
15 printf('The magnitude of the active force , Pae = %f
        lb/ft ',Pae)
```

Chapter 14

Lateral Earth Pressure Curved Failure Surface

Scilab code Exa 14.1 solved

```
1  clc
2  G=15.7
3  a=0
4  b=15
5  c=30
6  H=3
7  Kp=4.977 // from table 13.9
8  Pp=Kp*G*H^2/2
9  printf('a)the passive force = %f kN/m\n',Pp)
10 // for part b
11 Kp=4.53
12 Pp=Kp*G*H^2/2
13 printf(' b)the passive force = %f kN/m\n',Pp)
14 // for part c
15 Kp=4.13
16 Pp=Kp*G*H^2/2
17 printf(' c)the passive force = %f kN/m\n',Pp)
18 //for part d
19 Kp=4.56
```

```
20 Pp=Kp*G*H^2/2
21 printf(' d)the passive force = %f kN/m\n',Pp)
```

Scilab code Exa 14.2 solved

```
1 clc
2 G=16
3 H=7
4 c=30
5 Ta=0.65*G*H*(tand(45-c/2))^2
6 A=Ta*3*3/4
7 B1=Ta*3-54.61
8 C=Ta*4*4/4
9 B2=Ta*4-97.08
10 s=2
11 As=A*s
12 Bs=(B1+B2)*s
13 Cs=C*s
14 printf('The strut loads at level A = %f kN\n',As)
15 printf(' The strut loads at level B = %f kN\n',Bs)
16 printf(' The strut loads at level C = %f kN',Cs)
```

Chapter 15

Slope Stability

Scilab code Exa 15.1 solved

```
1  clc
2  Gs=17.8
3  Gw=9.81
4  C=10
5  c=20
6  b=15
7  H=6
8  G=Gs-Gw
9  Fs= C/(Gs*H*cosd(b)*cosd(b)*tand(b))+G*tand(c)/(Gs*
    tand(b))
10 printf('a)The factor of safety = %f \n',Fs)
11 Fs=2
12 H=2.247/(Fs-0.61)
13 printf(' b)H= %f m',H)
```

Scilab code Exa 15.2 solved

```
1  clc
```

```

2 G=105
3 c=15
4 C=600
5 b=45
6 Fs=3
7 Cd=C/Fs
8 c1= atand(tand(c)/Fs)
9
10 H= 4*Cd*(sind(b)*cosd(c1)/(1-cosd(b-c1)))/G
11 printf('The depth of the cut slope = %f ft',H)

```

Scilab code Exa 15.3 solved

```

1 clc
2 Cu=40
3 G=17.5
4 b=60
5 a=35
6 c=72.5
7 m=0.195
8 Hc=Cu/(G*m)
9 r=Hc/(2*sind(a)*sind(c/2))
10 BC=Hc*((1/tand(a))-(1/tand(b)))
11 printf(' a)The maximum depth Hc = %f m\n',Hc)
12 printf(' b)The radius , r = %f m\n',r)
13 printf(' c)The distance BC.= %f m',BC)

```

Scilab code Exa 15.4 solved

```

1 clc
2 Gs=17.29
3 d=9.15
4 d1=6.1

```

```

5 D=d/d1
6 a=40
7 m=0.175
8 b=40
9 H=6.1
10 Cu=H*Gs*m
11 printf('a)The undrained cohesion of the clay Cu = %f
      kN/m^2\n',Cu)
12 printf(' b)The nature of the critical circle is
      midpointcircle\n')
13 d=1.5
14 b=40
15 n=0.9
16 D1=n*H
17 printf(' c)Distance = %f m',D1)

```

Scilab code Exa 15.5 solved

```

1 clc
2 Fs=1
3 b=56
4 Kh=0.25
5 M=3.66
6 Cu=500
7 G=100
8 Hc=Cu*M/G
9 printf('a)The maximum depth = %f ft\n',Hc)
10 Fs=2
11 H=Cu*M/(G*Fs)
12 printf(' b)H= %f ft',H)

```

Scilab code Exa 15.6 solved

```

1  clc
2  b=45
3  c=20
4  C=24
5  G=18.9
6  m=0.06
7  Hc=C/(G*m)
8  Cd=G*Hc*m
9  Fc=C/Cd
10 printf('a) Critical height of slope = %f \n',Hc)

```

Scilab code Exa 15.7 solved

```

1  clc
2  FSs=1
3  c=20
4  G=18.9
5  C=24
6  Hcr=C/(G*tand(c)*0.17)
7  printf('a) Critical height Hc = %f m\n',Hcr)
8  H=10
9  k=C/(G*H*tand(c))
10 Fs=4*tand(c)
11 printf(' b)Fs = %f',Fs)

```

Scilab code Exa 15.8 solved

```

1  clc
2  W=22.4
3  C=20
4  a=70
5  s=sind(a)
6  c=cosd(a)

```

```
7 l=2.924
8 Wn=W*s
9 Wn1=W*c
10 //doing this to all values
11 F1=30.501
12 F2=776.75
13 F3=1638
14 Fs=(F1*C+F3*tand(C))/F2
15 printf('Fs = %f',Fs)
```

Scilab code Exa 15.9 solved

```
1 clc
2 C=20
3 G=18.5
4 r=0.25
5 H=21.62
6 C=25
7 b= atand(0.5)
8 //from table 15.3
9 m=1.624
10 n=1.338
11 Fs=m-n*r
12 printf(' The value of Fs for D= 1 is %f',Fs)
```

Scilab code Exa 15.11 solved

```
1 clc
2 C=20
3 G=18.5
4 H=21.62
5 c=25
6 r=0.25
```

```
7 Fs=3.1*tand(c)
8 printf('Fs = %f',Fs)
```

Chapter 16

Soil Bearing Capacity for Shallow Foundations

Scilab code Exa 16.1 solved

```
1  clc
2  c=20
3  // from table 16.1
4  Nc=17.69
5  Nq=7.44
6  Ng=3.64
7
8  Df=3
9  G=110
10 q=G*Df
11
12 C=200
13 B=4
14
15 Qu= C*Nc+q*Nq+G*B*Ng/2
16
17 Fs=3
18 Qa11=Qu/Fs
19 printf('Qa = %f lb/ft^2',Qa11)
```

Scilab code Exa 16.2 solved

```
1 clc
2 G=18.15
3 qa=30000*9.81/1000
4
5 Nc=57.75
6 Nq=41.44
7 Ng=45.41
8 C=0
9 q=G*1
10 B=1
11 (1.3*C*Nc+q*Nq+0.4*G*B*Ng)*B^2/3 == qa
12 B= sqrt(294.3/(250.7+109.9))
13 printf(' B = %f m',B)
```

Scilab code Exa 16.3 solved

```
1 clc
2 B=1.2
3 L=1.2
4 c=32
5 C=0
6 Df=1
7 G=16
8 Nq=23.18
9 Ng=22.02
10 Nc=1
11 Lqs=1+0.1*B*(tand(45+c/2))^2/L
12 Lgs=Lqs
13 Lqd=1+0.1*Df*tand(45+c/2)/B
```

```

14 Lgd=Lqd
15 Lcs=1
16 Lcd=1
17 Gs=19.5
18 q=0.5*G+0.5*(Gs-9.81)
19 Qu= C*Lcs*Lcd*Nc+q*Lqs*Lqd*Nq+(Gs-9.81)*Lgs*Lgd*B*Ng
    /2
20 Qa=Qu/3
21 Q=Qa*B^2
22 printf('the gross load = %f kN',Q)

```

Scilab code Exa 16.4 solved

```

1  clc
2  e=0.1
3  B=1
4  X=B-2*e
5  Y=1.5
6  B1=0.8
7  L1=1.5
8  c=30
9  Df=1
10 Nq=18.4
11 Ng=15.668
12 q=1*18
13 G=18
14 Lqs=1+e*(B1/L1)*(tand(45+c/2))^2
15 Lgs=Lqs
16 Lqd=1+e*(Df/B1)*tand(45+c/2)
17 Lgd=Lqd
18 qu=q*Lqs*Lqd*Nq+Lgs*Lgd*G*B1*Ng/2
19 Qu=qu*B1*L1
20 printf('The magnitude of the gross ultimate load =
    %f kN',Qu)

```

Scilab code Exa 16.5 solved

```
1  clc
2  B=1.5
3  Df=0.75
4  e=0.1*B
5  G=17.5
6  c=30
7  C=0
8  q=G*Df
9  Nq=18.4
10 Ng=15.668
11 Lqd=1+0.1*(Df/B)*tand(45+c/2)
12 Lgd=Lqd
13 Quc=q*Nq*Lqd+Lgd*B*Ng/2
14 k=0.8
15 a=1.754
16 Qua=Quc*(1-a*(e/B)^k)
17 printf('The gross ultimate load per unit length = %f
        kN', Qua)
```

Scilab code Exa 16.6 solved

```
1  clc
2  Qup=280
3  Bp=0.7 // in m
4  Bf=1.5
5  Quf=Qup*Bf/Bp
6  printf('The ultimate bearing capacity = %f kN/m^2',
        Quf)
```

Scilab code Exa 16.7 solved

```
1 clc
2 a=2500
3 //doing for the first values only
4 Bf=4
5 Bp=0.305
6 q=a/Bf^2
7 Sep=4
8 Sef=Sep*(2*Bf/(Bf+Bp))^2
9 printf('Sef = %f mm',Sef)
```

Chapter 18

Subsoil Exploration

Scilab code Exa 18.1 solved

```
1 clc
2 //solving for z=5 only
3 To=0.275
4 Cn=To^(-0.5)
5 N60=8
6 N160=Cn*N60
7 printf(' (N1)60 = %f',N160)
```

Scilab code Exa 18.2 solved

```
1 clc
2 z=5
3 To=0.275
4 Cn=2/(1+To)
5 N60=8
6 N160=Cn*N60
7 printf(' (N1)60 = %f',N160)
```

Scilab code Exa 18.3 solved

```
1 clc
2 pa=1 // 14.7 lb/in^2 = 1ton/ft^2
3 To=0.275 // ton/ ft^2
4 N60=8
5 c= atand((N60/(12.2+20.3*(To/pa)))^0.34)
6 printf('The average soil friction angle = %f',c)
```
