Chapter 3
Solved Problems

Problem 1

Script file:

```
clear, clc
x=-3:3;
y=x.^2-exp(0.5*x)+x
```

Command Window:

```
y =
    5.7769    1.6321   -0.6065   -1.0000    0.3513    3.2817    7.5183
```

Problem 2

Script file:

```
clear, clc
x=1:6;
y=(x+5).^3./x.^2
```

Command Window:

```
y =
  216.0000   85.7500   56.8889   45.5625   40.0000   36.9722
```

Problem 3

Script file:

```
clear, clc
x=[1.5:5.5 6.6];
y=(x+7).^4./((x+1).*sqrt(x))
```

Command Window:

```
y =

   1.0e+03 *

    1.7049    1.4718    1.4438    1.4991    1.6016    1.7521
```

Problem 4

Script file:

```
clear, clc
x=20:10:70;
y=(2*sind(x)+cosd(x).^2)./sind(x).^2
```

Command Window:

```
y =
   13.3962    7.0000    4.5317    3.3149    2.6427    2.2608
```

Problem 5

Script file:

```
clear, clc
s=50:50:300;
r=sqrt(s/pi)/2;
V=4*pi*r.^3/3;
table=[s' V']
```

Command Window:

```
table =
   50.0000   33.2452
  100.0000   94.0316
  150.0000  172.7471
  200.0000  265.9615
  250.0000  371.6925
  300.0000  488.6025
```

Problem 6

Script file:

```
clear, clc
e0=8.85e-12; lambda=1.7e-7; R=6;
disp('Part (a)')
z=0:2:10;
E=lambda*R*z./(2*e0*(z.^2+R^2).^(3/2))
disp('Part (b)')
z=2:.01:6;
E=lambda*R*z./(2*e0*(z.^2+R^2).^(3/2));
[m indx]=max(E);
maxE=m
at_z=z(indx)
```

Command Window:

```
Part (a)
E =
         0  455.5824  614.7264  565.9518  461.0169  363.3445
Part (b)
maxE =
  616.1301
at_z =
    4.2400
```

Problem 7

Script file:

```
clear, clc
V0=24; R=3800; C=4000*10^-6;
T0=R*C;
t=0:2:20;
Vc=V0*(1-exp(-t/T0));
i=V0/R*exp(-t/T0);
%table display introduced in Ch. 4
table=[t' Vc' i']
```

Command Window:

```
table =
         0         0    0.0063
    2.0000    2.9590    0.0055
    4.0000    5.5531    0.0049
    6.0000    7.8274    0.0043
    8.0000    9.8213    0.0037
   10.0000   11.5694    0.0033
   12.0000   13.1020    0.0029
   14.0000   14.4456    0.0025
   16.0000   15.6236    0.0022
   18.0000   16.6563    0.0019
   20.0000   17.5617    0.0017
```

Problem 8

Script file:

```
clear, clc
u=[23.5 -17 6];
disp('Part (a)')
length_u=sqrt(u(1)^2+u(2)^2+u(3)^2)
disp('Part (b)')
length_u=sqrt(sum(u.*u))
```

Command Window:

```
Part (a)
length_u =
   29.6184
Part (b)
length_u =
   29.6184
```

Problem 9

Script file:

```
clear, clc
u=[7,-4,-11];
vector=18*u/sqrt(sum(u.*u))
```

Command Window:

```
vector =
    9.2388   -5.2793  -14.5181
```

Problem 10

Script file:

```
clear, clc
v=[15,8,-6]; u=[3,-2,6];
disp('Part (a)')
v./u
disp('Part (b)')
u'*v
disp('Part (c)')
u*v'
```

Command Window:

```
Part (a)
ans =
     5    -4    -1
Part (b)
ans =
    45    24   -18
   -30   -16    12
    90    48   -36
Part (c)
ans =
    -7
```

Problem 11

Script file:

```
clear, clc
u=[5,-6,9]; v=[11,7,-4];
disp('Part (a)')
dotuv=sum(u.*v)
disp('Part (b)')
dotuv=u*v'
disp('Part (c)')
dotuv=dot(u,v)
```

Command Window:

```
Part (a)
dotuv =
   -23
Part (b)
dotuv =
   -23
Part (c)
dotuv =
   -23
```

Problem 12

Script file:

```
clear, clc
v=2:2:6;
disp('Part (a)')
a=2*v
disp('Part (b)')
b=v.^3
disp('Part (c)')
c=v.^2
disp('Part (d)')
d=v/2
```

Command Window:

```
Part (a)
a =
     4     8    12
Part (b)
b =
     8    64   216
Part (c)
c =
     4    16    36
```

```
Part (d)
d =
     1     2     3
```

Problem 13

Script file:

```
clear, clc
v=8:-2:2;
disp('Part (a)')
a=v./v
disp('Part (b)')
b=1./v.^2
disp('Part (c)')
c=1./sqrt(v)
disp('Part (d)')
d=v-5
```

Command Window:

```
Part (a)
a =
     1     1     1     1
Part (b)
b =
    0.0156    0.0278    0.0625    0.2500
Part (c)
c =
    0.3536    0.4082    0.5000    0.7071
Part (d)
d =
     3     1    -1    -3
```

Problem 14

Script file:

```
clear, clc
disp('Problem 14')
x=1:5; y=2*x;
disp('Part (a)')
z=(x+y).^2./(x-y)
disp('Part (b)')
w=x.*log(x.^2+y.^2) + sqrt(y.^3./(y-x).^2)
```

Command Window:

```
Part (a)
z =
    -9   -18   -27   -36   -45
```

```
Part (b)
w =
    4.4379    9.9915   16.3190   23.1850   30.4661
```

Problem 15

Script file:

```
clear, clc
r=1.6e3; s=14.2;
t=1:5; x=2*(t-1); y=3*t;
disp('Part (a)')
G=x.*t+r/s^2*(y.^2-x).*t
disp('Part (b)')
R=r*(-x.*t+y.*t.^2)/15-s^2*(y-0.5*x.^2).*t
```

Command Window:

```
Part (a)
G =
   1.0e+03 *
    0.0714    0.5436    1.8450    4.4041    8.6494
Part (b)
R =
   1.0e+04 *
   -0.0285    0.0520    0.6755    2.2759    5.2873
```

Problem 16

Script file:

```
clear, clc
rOA=[8,5,-4]; rOB=[-7,9,6]; rOC=[-5,-2,11];
rAB = rOB-rOA; rAC=rOC-rOA;
Area = sqrt(sum(cross(rAB,rAC).^2))/2
```

Command Window:

```
Area =
  112.4433
```

Problem 17

Script file:

```
clear, clc
rOA=[2,5,1]; rOB=[1,3,6]; rOC=[-6,8,2];
rAC=rOC-rOA;
%note, if order of rOC and rAC reversed will get negative volume
Volume=dot(rOB,cross(rOC,rAC))
```

Command Window:

```
Volume =
   248
```

Problem 18

Script file:

```
clear, clc
u=[5,-2,4]; v=[-2,7,3]; w=[8,1,-3];
%compare LHS and RHS
LHS=dot(u+v,cross(v+w,w+u))
RHS=2*dot(u,cross(v,w))
```

Command Window:

```
LHS =

  -776

RHS =

  -776
```

Problem 19

Script file:

```
clear, clc
r1=[6,-3,2]; r2=[2,9,10];
theta=acosd(dot(r1,r2)/(sqrt(dot(r1,r1))*sqrt(dot(r2,r2))))
```

Command Window:

```
theta =
   86.9897
```

Problem 20

Script file:

```
clear, clc
R=14; xA=8.4; yA=sqrt(R^2-xA^2);
B=[-R,0]; A=[xA,yA]; C=[R,0];
rAB=B-A; rAC=C-A;
disp('Part (a)')
alpha=acosd(dot(rAB,rAC)/(sqrt(dot(rAB,rAB))*sqrt(dot(rAC,rAC))))
disp('Part (b)')
%cross function requires 3rd dimension or could just use
%sqrt(abs(rAB(1)*rAC(2)-rAB(2)*rAC(1))) to explicitly calc cross product
alpha=asind(sqrt(sum(cross([rAB 0],[rAC 0]).^2))/ ...
    (sqrt(dot(rAB,rAB))*sqrt(dot(rAC,rAC))))
```

Command Window:

```
Part (a)
alpha =
    90
Part (b)
alpha =
   90.0000
```

Problem 21

Script file:

```
clear, clc
g=9.81; v0=162; alpha=70;
t=1:5:31;
x=v0*cosd(alpha)*t;
y=v0*sind(alpha)*t - g*t.^2/2;
r = sqrt(x.^2+y.^2)
theta = atand(y./x)
```

Command Window:

```
r =
   1.0e+03 *
    0.1574    0.8083    1.2410    1.4759    1.5564    1.5773    1.7176
theta =
   69.3893   65.7152   60.5858   53.0831   41.6187   24.0270    0.1812
```

Problem 22

Script file:

```
clear, clc
format long
e_squared=exp(2)
disp('Part (a)')
n=0:5;
sum_5=sum(2.^n./factorial(n))
disp('Part (b)')
n=0:10;
sum_10=sum(2.^n./factorial(n))
disp('Part (c)')
n=0:50;
sum_50=sum(2.^n./factorial(n))
```

Command Window:

```
e_squared =
   7.389056098930650
Part (a)
sum_5 =
   7.266666666666667
Part (b)
sum_10 =
   7.388994708994708
Part (c)
sum_50 =
   7.389056098930649
```

Problem 23

Script file:

```
clear, clc
format long
nat_log_10=log(10)
disp('Part (a)')
n=1:10;
sum_10=sum((9/10).^n./n)
disp('Part (b)')
n=1:50;
sum_50=sum((9/10).^n./n)
disp('Part (c)')
n=1:100;
sum_100=sum((9/10).^n./n)
```

Command Window:

```
nat_log_10 =
   2.302585092994046
Part (a)
sum_10 =
   2.118747594831429
Part (b)
sum_50 =
   2.301796252501072
Part (c)
sum_100 =
   2.302582905639062
```

Problem 24

Script file:

```
clear, clc
format long
disp('Part (a)')
n=1:5;
sum_5=sum(1./2.^n)
disp('Part (b)')
n=1:10;
sum_10=sum(1./2.^n)
disp('Part (c)')
n=1:40;
sum_40=sum(1./2.^n)
```

Command Window:

```
Part (a)

sum_5 =

   0.968750000000000

Part (b)

sum_10 =

   0.999023437500000

Part (c)

sum_40 =

   0.999999999999091
```

Problem 25

Script file:

```
clear, clc
format long
x=[1 .5 .1 .01 .001 .0001]
each_result=(cos(2*x)-1)./(cos(x)-1)
disp(' ')
disp('Problem 26')
x=[2, 1.5, 1.1, 1.01, 1.001, 1.00001, 1.0000001]
each_result=(x.^(1/3)-1)./(x.^(1/4)-1)
```

Command Window:

```
x =
  Columns 1 through 3
   1.000000000000000   0.500000000000000   0.100000000000000
  Columns 4 through 6
   0.010000000000000   0.001000000000000   0.000100000000000
each_result =
  Columns 1 through 3
   3.080604611736280   3.755165123780746   3.990008330556008
  Columns 4 through 6
   3.999900000832619   3.999999000133061   4.000000000000000
```

Problem 26

Script file:

```
clear, clc
format long
x=[2, 1.5, 1.1, 1.01, 1.001, 1.00001, 1.0000001]
each_result=(x.^(1/3)-1)./(x.^(1/4)-1)
```

Command Window:

```
x =
  Columns 1 through 3
   2.000000000000000   1.500000000000000   1.100000000000000
  Columns 4 through 6
   1.010000000000000   1.001000000000000   1.000010000000000
  Column 7
   1.000000100000000
each_result =
  Columns 1 through 3
   1.373738243887579   1.356502047955700   1.338663501189040
  Columns 4 through 6
   1.333886511598036   1.333388864983563   1.333333888920624
  Column 7
   1.333333336293928
```

Problem 27

Script file:

```
clear, clc
P=10:10:200;
Q=1020*sqrt(P).*(1-.01*sqrt(P))
```

Command Window:

```
Q =
   1.0e+04 *
  Columns 1 through 7
    0.3124    0.4358    0.5281    0.6043    0.6702    0.7289    0.7820
  Columns 8 through 14
    0.8307    0.8759    0.9180    0.9576    0.9950    1.0304    1.0641
  Columns 15 through 20
    1.0962    1.1270    1.1565    1.1849    1.2122    1.2385
```

Problem 28

Script file:

```
clear, clc
R=0.08206; T=300; n=1; a=1.39; b=0.0391;
V=0.1:.02:1;
P_ideal=n*R*T./V;
P_vW=n*R*T./(V-n*b)-n^2*a./V.^2;
error=100*(P_ideal-P_vW)./P_vW;
[m indx]=max(error);
max_error=m
at_volume=V(indx)
```

Command Window:

```
max_error =
    4.2359
at_volume =
    0.2400
```

Problem 29

Script file:

```
clear, clc
A=[1 -3 5; 2 2 4; -2 0 6]; B=[0 -2 1; 5 1 -6; 2 7 -1];
C=[-3 4 -1; 0 8 2; -3 5 3];
disp('Part (a)')
AplusB=A+B
BplusA=B+A
disp('Part (b)')
AplusBandC=A+(B+C)
AandBplusC=(A+B)+C
disp('Part (c)')
together=3*(A+C)
apart=3*A+3*C
disp('Part (d)')
%element by element
e_by_e_together=A.*(B+C)
e_by_e_apart=A.*B+A.*C
%matrix multiplication
mm_together=A*(B+C)
mm_apart=A*B+A*C
```

Command Window:

```
Part (a)
AplusB =
     1    -5     6
     7     3    -2
     0     7     5
BplusA =
     1    -5     6
     7     3    -2
     0     7     5
Part (b)
AplusBandC =
    -2    -1     5
     7    11     0
    -3    12     8
AandBplusC =
    -2    -1     5
     7    11     0
    -3    12     8
```

```
Part (c)
together =
    -6     3    12
     6    30    18
   -15    15    27
apart =
    -6     3    12
     6    30    18
   -15    15    27
Part (d)
e_by_e_together =
    -3    -6     0
    10    18   -16
     2     0    12
e_by_e_apart =
    -3    -6     0
    10    18   -16
     2     0    12
mm_together =
   -23    35    22
     0    70     0
     0    68    12
mm_apart =
   -23    35    22
     0    70     0
     0    68    12
```

Problem 30

Script file:

```
clear, clc
disp('Part (a)')
p1=A*B
p2=B*A
disp('no')
disp('Part (b)')
v1=A*(B*C)
v2=(A*B)*C
disp('yes')
disp('Part (c)')
t1=(A*B)'
t2=A'*B'
disp('no')
disp('Part (d)')
s1=(A+B)'
s2=A'+B'
disp('yes')
```

Command Window:

```
Part (a)
p1 =
    -5    30    14
    18    26   -14
    12    46    -8
p2 =
    -6    -4    -2
    19   -13    -7
    18     8    32
no
Part (b)
v1 =
   -27   290   107
   -12   210    -8
   -12   376    56
v2 =
   -27   290   107
   -12   210    -8
   -12   376    56
yes
Part (c)
t1 =
    -5    18    12
    30    26    46
    14   -14    -8
t2 =
    -6    19    18
    -4   -13     8
    -2    -7    32
no
Part (d)
s1 =
     1     7     0
    -5     3     7
     6    -2     5
s2 =
     1     7     0
    -5     3     7
     6    -2     5
Yes
```

Problem 31

Script file:

```
clear, clc
A=10*rand(4,4)
disp('Part (a)')
disp('linear algebra multiplication')
R=A*A
disp('Part (b)')
disp('element-by-element multiplication')
R=A.*A
disp('Part (c)')
disp('linear algebra, left division (left multiply by inverse)')
R=A\A
disp('Part (d)')
disp('element-by element, right division')
R=A./A
disp('Part (e)')
disp('determinant')
R=det(A)
disp('Part (f)')
disp('inverse')
R=inv(A)
```

Command Window:

```
A =

    8.1472    6.3236    9.5751    9.5717

    9.0579    0.9754    9.6489    4.8538

    1.2699    2.7850    1.5761    8.0028

    9.1338    5.4688    9.7059    1.4189

Part (a)

linear algebra multiplication

R =

  223.2405  136.6999  247.0195  198.8841

  139.2180  111.6463  158.4599  175.5387

  110.6692   58.9020  119.1899   49.6407

  149.2358   97.8828  169.2935  193.6574

Part (b)

element-by-element multiplication
```

```
R =

   66.3775   39.9878   91.6819   91.6169

   82.0459    0.9514   93.1010   23.5590

    1.6126    7.7561    2.4842   64.0449

   83.4255   29.9079   94.2050    2.0132
```

Part (c)

linear algebra, left division (left multiply by inverse)

```
R =

    1.0000         0    0.0000   -0.0000

    0.0000    1.0000   -0.0000   -0.0000

   -0.0000         0    1.0000    0.0000

    0.0000         0         0    1.0000
```

Part (d)

element-by element, right division

```
R =

     1     1     1     1

     1     1     1     1

     1     1     1     1

     1     1     1     1
```

Part (e)

determinant

```
R =

 -261.4072
```

```
Part (f)

inverse

R =

   -1.5300    0.3076    1.4723    0.9645

   -0.0209   -0.1844    0.1037    0.1871

    1.4569   -0.1934   -1.4650   -0.9041

   -0.0369    0.0535    0.1438   -0.0401
```

Problem 32

Script file:

```
clear, clc
M=magic(6);
disp('check rows')
sum_rows=sum(M')
disp('check columns')
sum_cols=sum(M)
disp('check one diagonal')
dum_d1=sum(diag(M))
disp('check other diagonal')
dum_d1=sum(diag(fliplr(M)))
```

Command Window:

```
check rows
sum_rows =
   111   111   111   111   111   111
check columns
sum_cols =
   111   111   111   111   111   111
check one diagonal
dum_d1 =
   111
check other diagonal
dum_d1 =
   111
```

Problem 33

Script file:

```
clear, clc
A=[-4 3 1; 5 6 -2; 2 -5 4.5]; y=[-18.2 -48.8 92.5]';
result=A\y
```

Command Window:

```
result =

    2.8000

   -6.4000

   12.2000
```

Problem 34

Script file:

```
clear, clc
B=[2.5 -1 3 1.5 -2; 3 4 -2 2.5 -1; -4 3 1 -6 2; 2 3 1 -2.5 4; 1 2 5 -3 4];
y=[57.1 27.6 -81.2 -22.2 -12.2]';
result=B\y
disp('check')
B*result
```

Command Window:

```
result =
    8.2000
   -2.0000
    4.8000
    6.0000
   -5.6000
```

Problem 35

Script file:

```
clear, clc
R=[3 1 1 2 1; 1 2 1 3 1; 1 1 0 3 3; 2 0 3 1 2; 1 2 3 0 2];
p=16*[128 118 112 112 104]';
result=R\p
```

Command Window:

```
result =
  320.0000
  224.0000
  192.0000
  256.0000
  160.0000
```

Problem 36

Script file:

```
clear, clc
V1=18; V2=18; V3=12; V4=28;
R1=4; R2=4; R3=6; R4=4; R5=2; R6=3; R7=2.5;
A=[-(R1+R2+R4) R2 R4 0; R2 -(R2+R3+R5) 0 R5; R4 0 -(R4+R6) R6; ...
    0 R5 R6 -(R5+R6+R7)];
V=[18 -18 12 -28]';
I=A\V
```

Command Window:

```
I =
   -1.1310
    1.7795
   -0.6725
    3.9389
```

Problem 37

Script file:

```
clear, clc
V1=40; V2=30; V3=36;
R1=16; R2=20; R3=10; R4=14; R5=8; R6=16; R7=10; R8=15; R9=6; R10=4;
A=[-(R1+R2+R3) R2 R3 0 0; R2 -(R2+R4+R5+R6) R5 R6 R4; ...
    R3 R5 -(R3+R5+R7) R7 0; 0 R6 R7 -(R6+R7+R8+R9) R8; ...
    0 R4 0 R8 -(R4+R8+R10)];
V=[-V1 0 -V2 V3 V1]';
I=A\V
```

Command Window:

```
I =
    0.7406
   -0.6047
    0.6161
   -1.5316
   -2.1649
```