
Table of Contents

| | |
|-----------------------|----|
| | 1 |
| Problem 1 | 1 |
| Problem 2 | 8 |
| Problem 3 | 13 |
| Problem 4 | 19 |
| Problem 5 | 21 |
| Problem 6 | 22 |
| Problem 7 | 24 |
| Problem 8 | 25 |
| Problem 9 | 27 |
| Functions for 3 | 28 |
| Function for 5 | 30 |
| Function for 7 | 31 |
| Function for 8 | 32 |
| Function for 9 | 32 |

```
%{
Writer: Akshay S Tharval
1st draft: Oct 30, 2015
Last modified: Oct 30, 2015
```

```
Subject: Assignment 4
%}
```

```
function main()
```

Problem 1

```
%Q1
%Each case has been plotted on three different plots
disp('=====')
disp('Q1')
disp('Q1a')

%Given values and calculated values from given data
p = 0;
q = 1;
r = 1;
a11 = 1;
a12 = 0;
b1 = 0;
b7 = 1;
n = 6;
h = 1/n;
l = -1-(h/2)*p;
d = 2+(h^2*q);
u = -1+(h/2)*p;
A = zeros(n+1);
```

```

% For matrix A
A(1,1) = a11;
A(1,2) = a12;
A(2,1) = 1;
A(n,n+1) = u;
A(n+1,n+1) = 1;
for i = (2:n)
    A(i,i) = d;
    A(i,i+1) = u;
    A(i+1,i) = 1;
end
A(n+1,n) = 0;
disp('Matrix A is')
disp(A)

% For constant vector
b = zeros(n+1,1);
b(1,1) = b1;
b(n+1,1) = b7;
for i = (2:n)
    b(i,1) = -(h^2)*r;
end
disp('Matrix b is')
disp(b)

% For calculating the result
sol = zeros(n+1,1);
sol = inv(A)*b;
disp('Solution using the given condition is')
disp(sol)

%For plotting
for i = (1:n+1)
    x(i,1) = i*h;
end
figure();
plot(x,sol,'b');
hold on
plot(x,sol,'ro');
xlabel('Node (xi)');
ylabel('Value at node (wi)');
title('Problem 1:Case a: u(0) = 0; u(1) = 1');
grid on
legend('w1(xi)')
hold off

disp('-----')
disp('Q1b')

%Given data and value of calculated values from given data
p = 0;
q = 1;
r = 1;
h = 1/n;

```

```

n = 6;
l = -1-(h/2)*p;
d = 2+(h^2*q);
u = -1+(h/2)*p;
alp1 = 1;
alp2 = 0;
alp3 = 0;
bet1 = 1;
bet2 = 1;
bet3 = 1;

% For A matrix
A = zeros(n+1);
A(1,1) = 1;
A(1,2) = 0;
A(2,1) = 1;
A(n,n+1) = u;
A(n+1,n+1) = d - 2* h * u * (bet1/bet2);
for i = (2:n)
    A(i,i) = d;
    A(i,i+1) = u;
    A(i+1,i) = 1;
end
A(n+1,n) = -2;
disp('Matrix A is')
disp(A)

% For constant vector
b = zeros(n+1,1);
b(1,1) = 0;
b(n+1,1) = -h^2*r - 2 * h * u *(bet3/bet2);
for i = (2:n)
    b(i,1) = -(h^2)*r;
end
disp('Matrix b is')
disp(b)

% For finding the solution
sol = zeros(n+1,1);
sol = inv(A)*b;
disp('Solution using the given condition is')
disp(sol)

% For plotting
for i = (1:n+1)
    x(i,1) = i*h;
end
figure();
plot(x,sol,'g');
hold on
plot(x,sol,'ro');
xlabel('Node (xi)');
ylabel('Value at node (wi)');
grid on

```

```

title('Problem 1:Case b:  $u(0) = 0$ ;  $u(1) + u_d(1) = 1$ ');
legend('w2(xi)')
hold off

disp('-----')
disp('Q1c')

% Given data and values calculated from given data
p = 0;
q = 1;
r = 1;
h = 1/n;
n = 6;
l = -1-(h/2)*p;
d = 2+(h^2*q);
u = -1+(h/2)*p;

% For matrix A
A = zeros(n+1);
A(1,1) = 1;
A(1,2) = 0;
A(2,1) = 1;
A(n,n+1) = u;
A(n+1,n+1) = d;
bet = 1
for i = (2:n)
    A(i,i) = d;
    A(i,i+1) = u;
    A(i+1,i) = 1;
end
A(n+1,n) = -2;
disp('Matrix A is')
disp(A)

% For constant vector
b = zeros(n+1,1);
b(1,1) = 0;
b(n+1,1) = -h^2*r - 2 * h * u *bet;
for i = (2:n)
    b(i,1) = -(h^2)*r;
end
disp('Matrix b is')
disp(b)

% For calculating the solution
sol = zeros(n+1,1);
sol = inv(A)*b;
disp('Solution using the given condition is')
disp(sol)

%For plotting
for i = (1:n+1)
    x(i,1) = i*h;
end

```

```

figure();
plot(x,sol,'r');
hold on
plot(x,sol,'bo');
xlabel('Node (xi)');
ylabel('Value at node (wi)');
grid on
title('Problem 1: Case c: u(0) = 0; ud(1) = 1');
legend('w3(xi)')
hold off

```

```
=====
```

Q1

Q1a

Matrix A is

| | | | | | | |
|---------|---------|---------|---------|---------|---------|---------|
| 1.0000 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1.0000 | 2.0278 | -1.0000 | 0 | 0 | 0 | 0 |
| 0 | -1.0000 | 2.0278 | -1.0000 | 0 | 0 | 0 |
| 0 | 0 | -1.0000 | 2.0278 | -1.0000 | 0 | 0 |
| 0 | 0 | 0 | -1.0000 | 2.0278 | -1.0000 | 0 |
| 0 | 0 | 0 | 0 | -1.0000 | 2.0278 | -1.0000 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1.0000 |

Matrix b is

```

0
-0.0278
-0.0278
-0.0278
-0.0278
-0.0278
1.0000

```

Solution using the given condition is

```

0
0.0792
0.1884
0.3306
0.5097
0.7308
1.0000

```

```
-----
```

Q1b

Matrix A is

| | | | | | | |
|---------|---------|---------|---------|---------|---------|---------|
| 1.0000 | 0 | 0 | 0 | 0 | 0 | 0 |
| -1.0000 | 2.0278 | -1.0000 | 0 | 0 | 0 | 0 |
| 0 | -1.0000 | 2.0278 | -1.0000 | 0 | 0 | 0 |
| 0 | 0 | -1.0000 | 2.0278 | -1.0000 | 0 | 0 |
| 0 | 0 | 0 | -1.0000 | 2.0278 | -1.0000 | 0 |
| 0 | 0 | 0 | 0 | -1.0000 | 2.0278 | -1.0000 |
| 0 | 0 | 0 | 0 | 0 | -2.0000 | 2.3611 |

Matrix b is

```

0

```

```

-0.0278
-0.0278
-0.0278
-0.0278
-0.0278
0.3056

```

Solution using the given condition is

```

0
-0.0303
-0.0337
-0.0103
0.0407
0.1205
0.2315

```

Q1c

bet =

```

1

```

Matrix A is

```

1.0000      0      0      0      0      0      0
-1.0000  2.0278 -1.0000      0      0      0      0
      0 -1.0000  2.0278 -1.0000      0      0      0
      0      0 -1.0000  2.0278 -1.0000      0      0
      0      0      0 -1.0000  2.0278 -1.0000      0
      0      0      0      0 -1.0000  2.0278 -1.0000
      0      0      0      0      0 -2.0000  2.0278

```

Matrix b is

```

0
-0.0278
-0.0278
-0.0278
-0.0278
-0.0278
0.3056

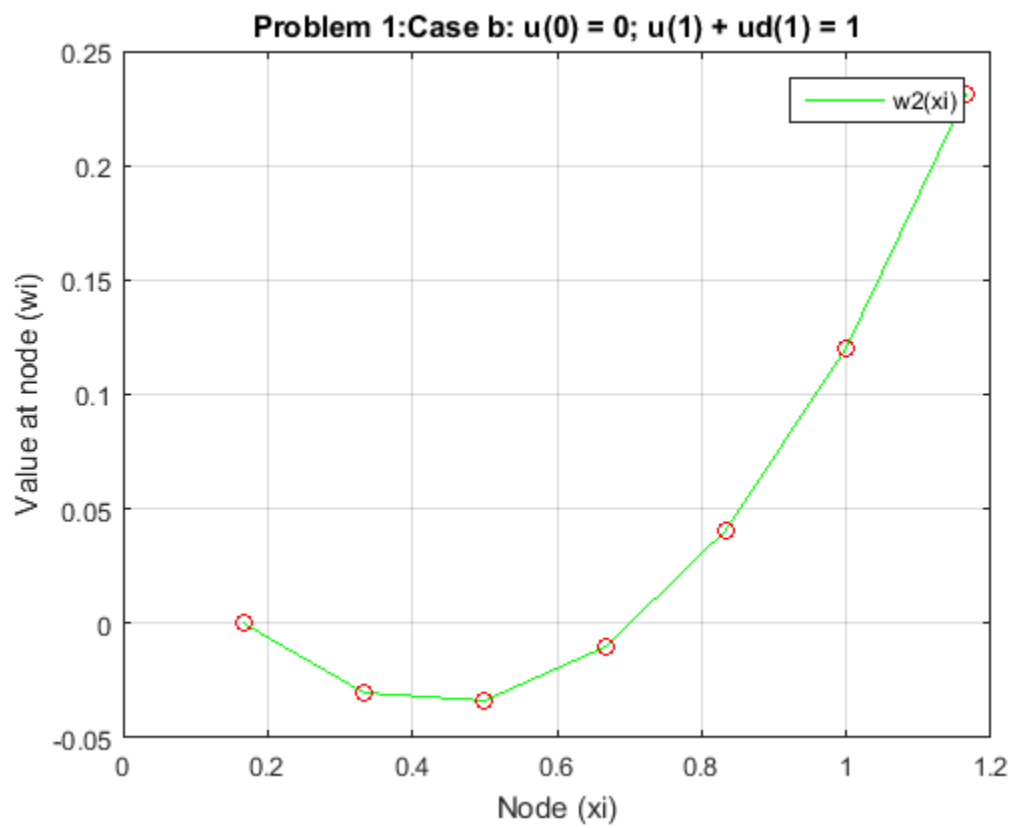
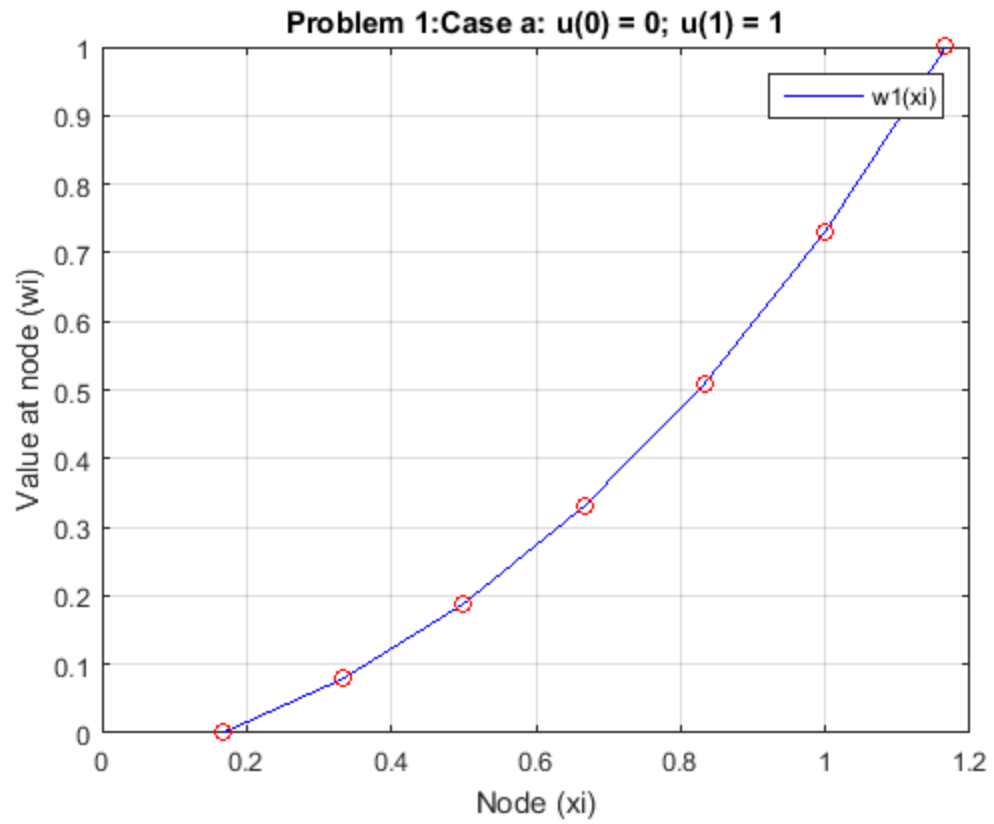
```

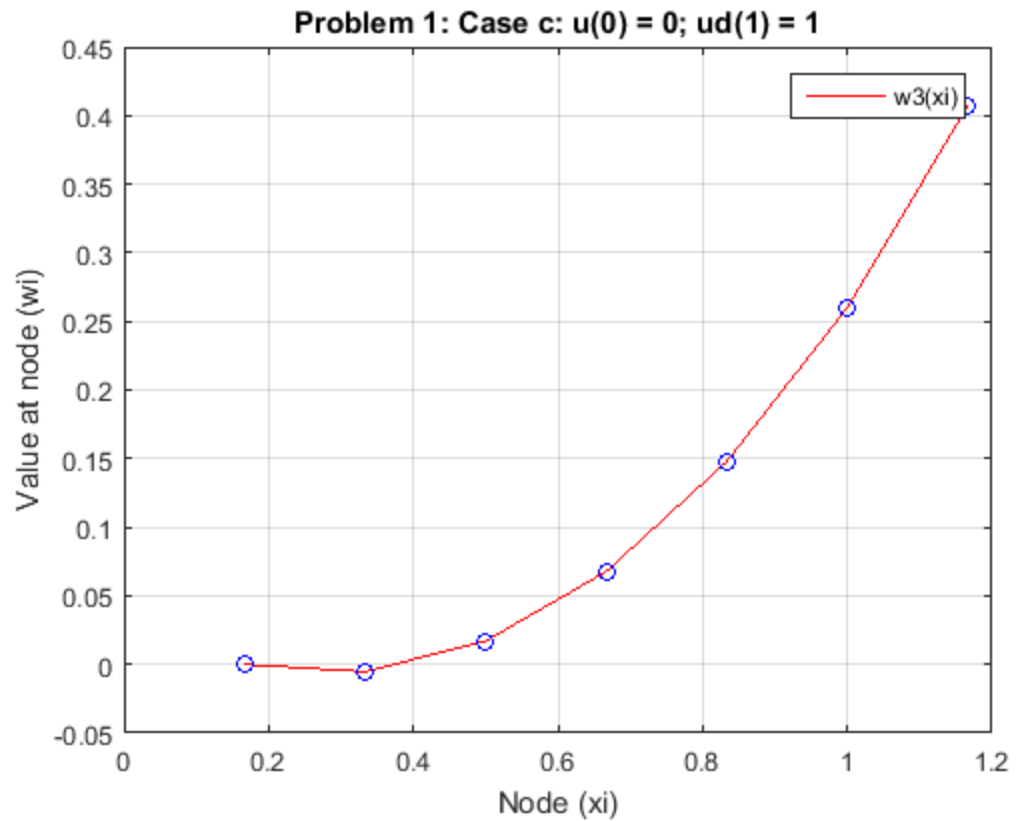
Solution using the given condition is

```

0
-0.0053
0.0170
0.0676
0.1479
0.2600
0.4071

```





Problem 2

```
%Q2
disp('=====')
disp('Q2')
disp('Case: a')

%Given data and values calculated from it
bet = 7.5;
zk = 0.8;
z = 1;
n = 6;
h = (z-zk)/n;
x = (zk:h:z)';
p = -1./x;
l = -1-(h/2)*p;
u = -1+(h/2)*p;
q = -bet^2;
r = -1;
d = 2+(h^2)*q;

% For matrix A
A = zeros(n+1);
A(1,1) = d;
A(1,2) = -2;
```

```

A(2,1) = l(1,1);
A(n,n+1) = u(n,1);
for i = (2:n)
    A(i,i) = d;
    A(i,i+1) = u(i,1);
    A(i+1,i) = l(i,1);
end
A(n+1,n+1) = 1;
A(n+1,n) = 0;
disp('Matrix A is')
disp(A)

% For constant vector
b = zeros(n+1,1);
b(1,1) = (-h^2)*r; %since alpha is 0
b(n+1,1) = 0; %Since beta is 0
for i = (2:n)
    b(i,1) = -r*(h^2);
end
disp('Matrix b is')
disp(b)

% For solving
soll = zeros(n+1,1);
soll = inv(A)*b;
disp('Solution using the given condition is')
disp(soll)

disp('-----')
disp('Case: b')

% Given data and values calculated from it
bet = 8;
zk = 0.8;
z = 1;
n = 6;
h = (z-zk)/n;
x = (zk:h:z)';
p = -1./x;
l = -1-(h/2)*p;
u = -1+(h/2)*p;
q = -bet^2;
r = -1;
d = 2+(h^2)*q;

%For matrix A
A = zeros(n+1);
A(1,1) = d;
A(1,2) = -2;
A(2,1) = l(1,1);
A(n,n+1) = u(n,1);
A(n+1,n+1) = 1;
for i = (2:n)
    A(i,i) = d;

```

```

        A(i,i+1) = u(i,1);
        A(i+1,i) = l(i,1);
    end
    A(n+1,n) = 0;
    disp('Matrix A is')
    disp(A)

    % For constant vector
    b = zeros(n+1,1);
    b(1,1) = (-h^2)*r; %since alpha is 0
    b(n+1,1) = 0; %Since beta is 0
    for i = (2:n)
        b(i,1) = -r*(h^2);
    end
    disp('Matrix b is')
    disp(b)

    % For solving the system
    sol2 = zeros(n+1,1);
    sol2 = inv(A)*b;
    disp('Solution using the given condition is')
    disp(sol2)

    disp('-----')
    disp('Case: c')
    % Given data and calculated values using those
    bet = 8.5;
    zk = 0.8;
    z = 1;
    n = 6;
    h = (z-zk)/n;
    x = (zk:h:z)';
    p = -1./x;
    l = -1-(h/2)*p;
    u = -1+(h/2)*p;
    q = -bet^2;
    r = -1;
    d = 2+(h^2)*q;

    % For the A matrix
    A = zeros(n+1);
    A(1,1) = d;
    A(1,2) = -2;
    A(2,1) = l(1,1);
    A(n,n+1) = u(n,1);
    A(n+1,n+1) = 1;
    for i = (2:n)
        A(i,i) = d;
        A(i,i+1) = u(i,1);
        A(i+1,i) = l(i,1);
    end
    A(n+1,n) = 0;
    disp('Matrix A is')
    disp(A)

```

```

% For a matrix of constants
b = zeros(n+1,1);
b(1,1) = (-h^2)*r; %since alpha is 0
b(n+1,1) = 0; %Since beta is 0
for i = (2:n)
    b(i,1) = -r*(h^2);
end
disp('Matrix b is')
disp(b)

%For calculating the solution
sol3 = zeros(n+1,1);
sol3 = inv(A)*b;
disp('Solution using the given condition is')
disp(sol3)

% For plotting
for i = (1:n+1)
    x(i,1) = i*h;
end
figure();
plot(x,sol1,'r',x,sol2,'y',x,sol3,'b');
xlabel('Node (xi)');
ylabel('Value at node (wi)');
grid on
legend('Case: a (beta = 7.5)', 'Case: a (beta = 8)', 'Case: a (beta = 8.5)');
title('Q2: Plot of different values of beta, for the given system');

```

```

=====
Q2
Case: a
Matrix A is
    1.9375   -2.0000         0         0         0         0         0
   -0.9792    1.9375   -1.0200         0         0         0         0
         0   -0.9800    1.9375   -1.0192         0         0         0
         0         0   -0.9808    1.9375   -1.0185         0         0
         0         0         0   -0.9815    1.9375   -1.0179         0
         0         0         0         0   -0.9821    1.9375   -1.0172
         0         0         0         0         0         0         1.0000

Matrix b is
    0.0011
    0.0011
    0.0011
    0.0011
    0.0011
    0.0011
    0

Solution using the given condition is
    0.1137
    0.1096

```

0.0980
0.0797
0.0562
0.0291
0

Case: b

Matrix A is

| | | | | | | |
|---------|---------|---------|---------|---------|---------|---------|
| 1.9289 | -2.0000 | 0 | 0 | 0 | 0 | 0 |
| -0.9792 | 1.9289 | -1.0200 | 0 | 0 | 0 | 0 |
| 0 | -0.9800 | 1.9289 | -1.0192 | 0 | 0 | 0 |
| 0 | 0 | -0.9808 | 1.9289 | -1.0185 | 0 | 0 |
| 0 | 0 | 0 | -0.9815 | 1.9289 | -1.0179 | 0 |
| 0 | 0 | 0 | 0 | -0.9821 | 1.9289 | -1.0172 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1.0000 |

Matrix b is

0.0011
0.0011
0.0011
0.0011
0.0011
0.0011
0.0011
0

Solution using the given condition is

0.3639
0.3504
0.3122
0.2529
0.1771
0.0908
0

Case: c

Matrix A is

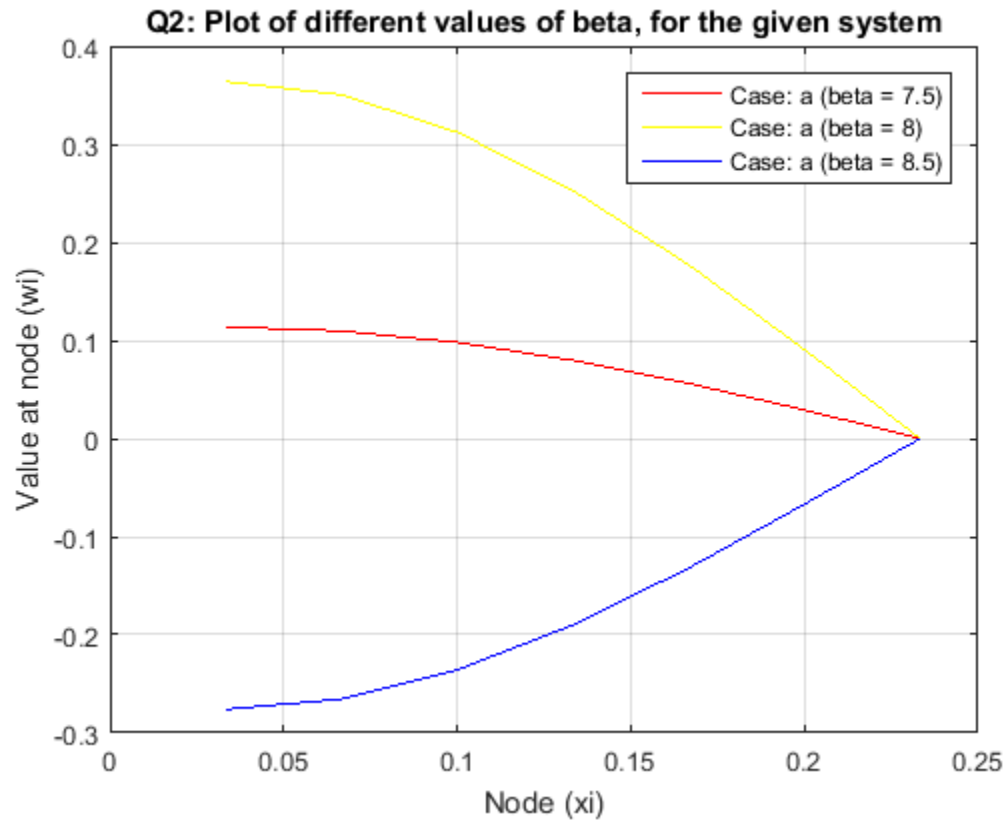
| | | | | | | |
|---------|---------|---------|---------|---------|---------|---------|
| 1.9197 | -2.0000 | 0 | 0 | 0 | 0 | 0 |
| -0.9792 | 1.9197 | -1.0200 | 0 | 0 | 0 | 0 |
| 0 | -0.9800 | 1.9197 | -1.0192 | 0 | 0 | 0 |
| 0 | 0 | -0.9808 | 1.9197 | -1.0185 | 0 | 0 |
| 0 | 0 | 0 | -0.9815 | 1.9197 | -1.0179 | 0 |
| 0 | 0 | 0 | 0 | -0.9821 | 1.9197 | -1.0172 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1.0000 |

Matrix b is

0.0011
0.0011
0.0011
0.0011
0.0011
0.0011
0.0011
0

Solution using the given condition is

-0.2761
-0.2655
-0.2358
-0.1900
-0.1321
-0.0670
0



Problem 3

```
%Q3
%{
Case a: psi^2 = 0.0
Case b: psi^2 = 10
Part A of the problem requires us to solve the system using shooting
method
Part B of the problem requires us to solve the system using finite
differences
Part C is about determining the gradient of concentration at the outer
edge
of the pellet
%}
disp('=====')
```

```

disp('Q3')
disp('Q3a')
disp('Shooting method when  $\psi^2 = 0.01$ ')
psispan = linspace(0.01, 1, 20);
c0 = [0;0];
[~,Y1] = ode45(@eval3a,psispan,c0);
c1 = [1;0];
[~,Y2] = ode45(@eval3a,psispan,c1);
% For the new value of c2
c2 = c0 + ((c1 - c0)*(1 - Y1(end,1))/(Y2(end,1)-Y1(end,1)));
[~,Y] = ode45(@eval3a,psispan,c2);

%Plotting for  $\psi^2 = 0.01$ 
figure();
subplot(1,2,1);
plot(psispan,Y(:,1));
xlabel('\xi');
ylabel('\theta(\xi)');
title('Problem 3a  $\psi^2 = 0.01$ ');
grid on
disp('\theta(\xi) for  $\psi^2 = 0.01$ ');
disp(Y(:,1));
cpart1 = Y(end,2);

% When  $\psi^2 = 10$ 
disp('Shooting method when  $\psi^2 = 10$ ')
c0 = [0;0];
[~,Y3] = ode45(@eval3b,psispan,c0);
c1 = [1;0];
[~,Y4] = ode45(@eval3b,psispan,c1);
% For the new value of c2
c2 = c0 + ((c1 - c0)*(1 - Y1(end,1))/(Y2(end,1)-Y1(end,1)));
[~,Y] = ode45(@eval3b,psispan,c2);

% Plotting for  $\psi^2 = 10$ 
subplot(1,2,2);
plot(psispan,Y(:,1));
xlabel('\xi');
ylabel('\theta(\xi)');
title('Problem 3a  $\psi^2 = 10$ ');
grid on
disp('\theta(\xi) for  $\psi^2 = 0.01$ ');
disp(Y(:,1));
cpart2 = Y(end,1);

disp('-----')
disp('Q3b')
disp('Finite differences method,  $\psi^2 = 0.01$ ')
% Given values and the values calculated from them
q = 0.01;
r = 0;
N = 6;
a = 0;
b = 1;

```

```

h = (b - a)/ N ;
alpha = 0;
beta = 1;

% For matrix A
A2 = zeros(N+1,N+1);
A2(1,1) = 2 + ((h^2)* q);
A2(1,2) = -2;
A2(N+1, N+1) = 1;
A2(N+1, N) = 0;

% For diagonal elements
for i = 2: N
    A2(i,i) = 2 + ((h^2) * q);
end

psi = linspace(0.01, 1, N+1);

% For u
for i = 2: N
    j = i+1;
    A2(i,j) = -1 + (h * (-2 / psi(i) ) / 2);
end

%For l
for i = 2 : N
    j = i - 1;
    A2(i,j) = - 1 - ( h * ( -2 / psi(i) ) / 2);
end
disp('The co-eff matrix:');
disp(A2);

% For the constant matrix
b1 = zeros(N+1,1);
disp('The constant matrix:');
b1 = [-(h^2 * r)+(2 * h * (-1 - (h * (-2 /psi(1) ) / 2)) * alpha);
    -(h^2) * r;
    -(h^2) * r;
    -(h^2) * r;
    -(h^2) * r;
    -(h^2) * r;
    beta];

disp(b1);

% Taking the inverse to solve
w1 = inv(A2)*b1;
disp('Approximate solution with phi^2 = 0.01');
disp(w1);

x1 = linspace(0,1,N+1);

% For plotting
figure();

```

```

subplot(1,2,1);
plot(x1, w1);
xlabel('\xi');
ylabel('\theta(\xi)');
title('Problem 3b phi^2 = 0.01');
grid on

% Given data and values calculated from it
q = 10;
r = 0;
N = 6;
a = 0;
b = 1;
h = (b - a)/ N ;
alpha = 0;
beta = 1;

% Making matrix A
A2 = zeros(N+1,N+1);
A2(1,1) = 2 + ((h^2)* q);
A2(1,2) = -2;
A2(N+1, N+1) = 1;
A2(N+1, N) = 0;
% For diagonal elements
for i = 2: N
    A2(i,i) = 2 + ((h^2) * q);
end

psi = linspace(0.01, 1, N+1);

% For u
for i = 2: N
    j = i+1;
    A2(i,j) = -1 + (h * (-2 / psi(i) ) / 2);
end

% For l
for i = 2 : N
    j = i - 1;
    A2(i,j) = - 1 - ( h * ( -2 / psi(i) ) / 2);
end
disp('The co-eff matrix:');
disp(A2);

% For constant matrix
b2 = zeros(N+1,1);
disp('The constant vector:');
b2 = [-(h^2 * r)+(2 * h * (-1 - (h * (-2 /psi(1) ) / 2)) * alpha);
    -(h^2) * r;
    -(h^2) * r;
    -(h^2) * r;
    -(h^2) * r;
    -(h^2) * r;

```

```

        beta];

disp(b2);

% Taking the inverse to solve
w2 = inv(A2)*b2;
disp('Approximate solution with  $\phi^2 = 10$ ');
disp(w2);

x2 = linspace(0,1,N+1);

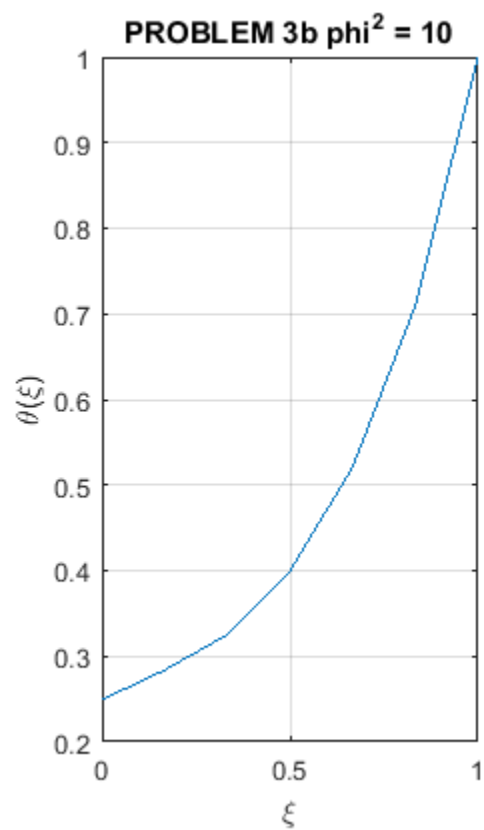
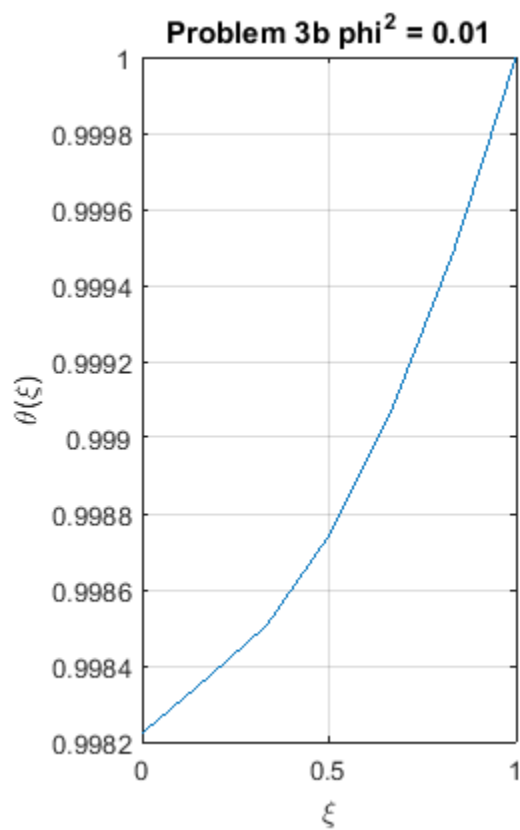
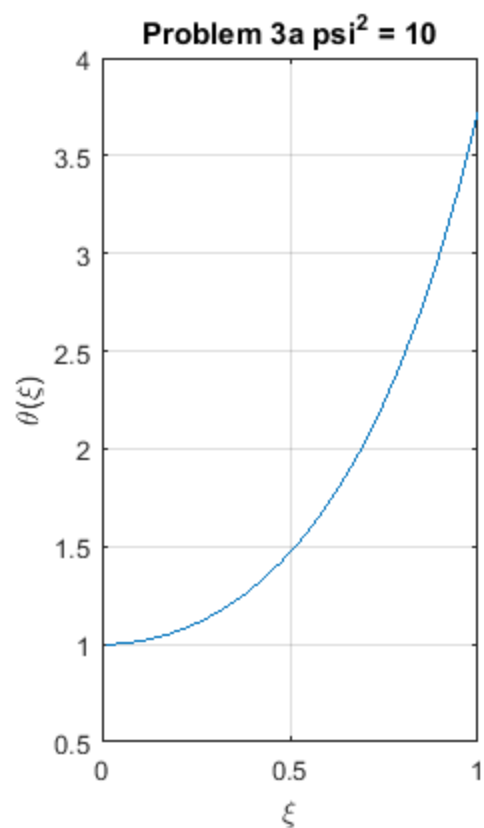
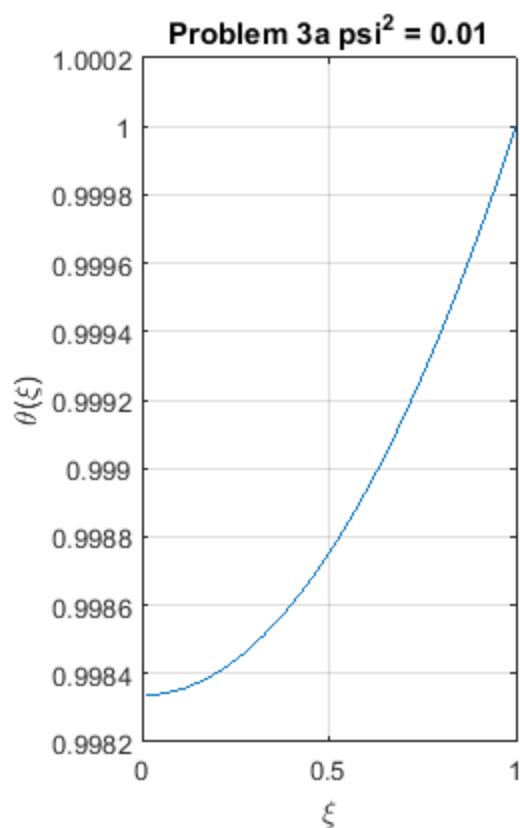
% Plotting the graph
subplot(1,2,2);
plot(x2, w2);
xlabel('\xi');
ylabel('\theta(\xi)');
title('PROBLEM 3b  $\phi^2 = 10$ ');
grid on

%Part C
disp('Q3c')
disp(' The gradient of concentration with  $\phi^2 = 0.01$  is :');
disp(cpart1);

disp(' The gradient of concentration with  $\phi^2 = 10$  is :');
disp(cpart2);

=====
Q3
Q3a
Shooting method when  $\psi^2 = 0.01$ 

```



Problem 4

```
%Q4
disp('=====')
disp('Q4')
%The given data and values calculated from it
D = 10 * (10 ^-12); % m^2/sec
L = 10^-3; % m
v = 0.1 * (10^-6); % m/sec
k = 5 * (10^(-3)); % M^-1 sec^-1
n = 20;
h = (1 - 0)/n;
p = -(v*L)/D;
q = (k*L^2)/D;
r = 0;
A = zeros(n+1);
l = -1-(h/2)*p;
u = -1 + (h/2)*p;
d = 2+(h^2)*q;

% For matrix A
A(1,1) = 1;
A(1,2) = 0;
A(2,1) = 1;
A(n,n+1) = u;
A(n+1,n+1) = 1;
for i = (2:n)
    A(i,i) = d;
    A(i,i+1) = u;
    A(i+1,i) = 1;
end
A(n+1,n) = 0;

% For constant vector
b = zeros(n+1,1);
b(1,1) = 1;
b(n+1,1) = 0.1;
for i = (2:n)
    b(i,1) = -(h^2)*r;
end

%For solving the system
sol = zeros(n+1,1);
sol = inv(A)*b;
disp('Solution using the given condition is')
disp(sol)

%For plotting
for i = (1:n+1)
    x(i,1) = i*h;
end
figure();
plot(x,sol,'r');
```

```

hold on
plot(x,sol,'bo');
xlabel('z*');
ylabel('Concentration');
ylim([-0.2 1])
title('Problme 4: One-Dimensional transport of species');
grid on
legend('C(z*)')
hold off

```

```
=====
```

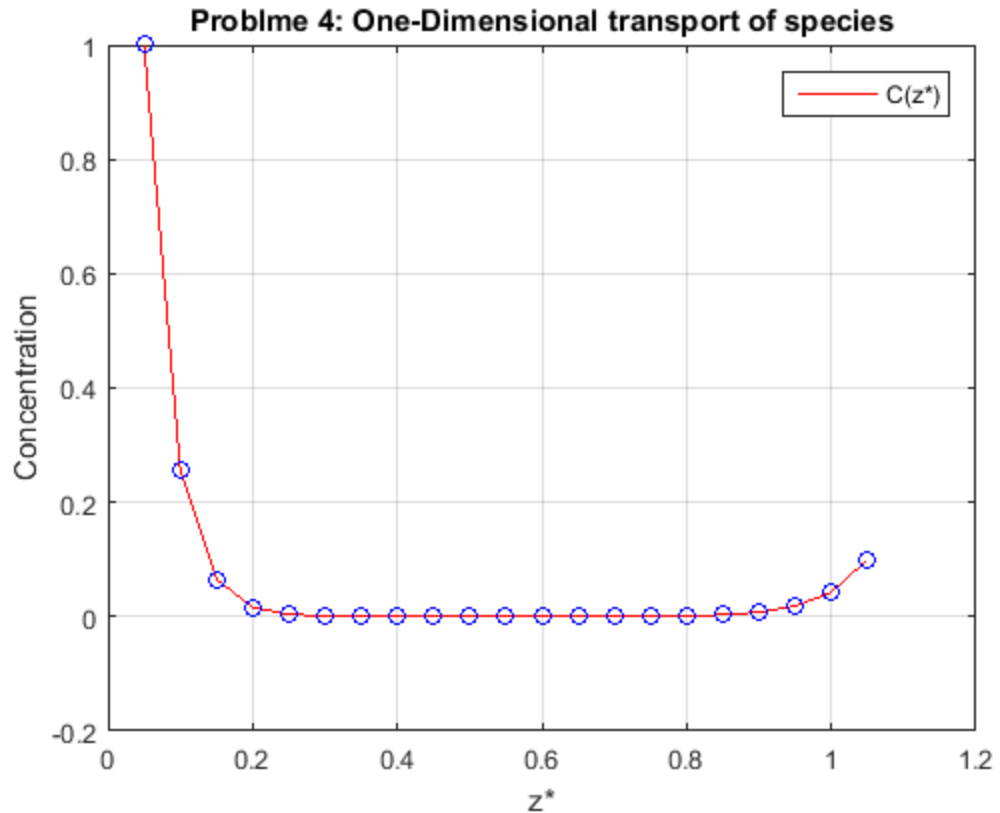
Q4

Solution using the given condition is

```

1.0000
0.2560
0.0655
0.0168
0.0043
0.0011
0.0003
0.0001
0.0000
0.0000
0.0000
0.0000
0.0001
0.0003
0.0006
0.0014
0.0033
0.0078
0.0182
0.0427
0.1000

```



Problem 5

Q5

```

disp('-----')
disp('Q5')
% Here, N = no of points at which the concentration is calculated.
% Solving the non-linear problem using fsolve
N = 7;
h = (1-0)/(N-1);
z = 0:h:1;

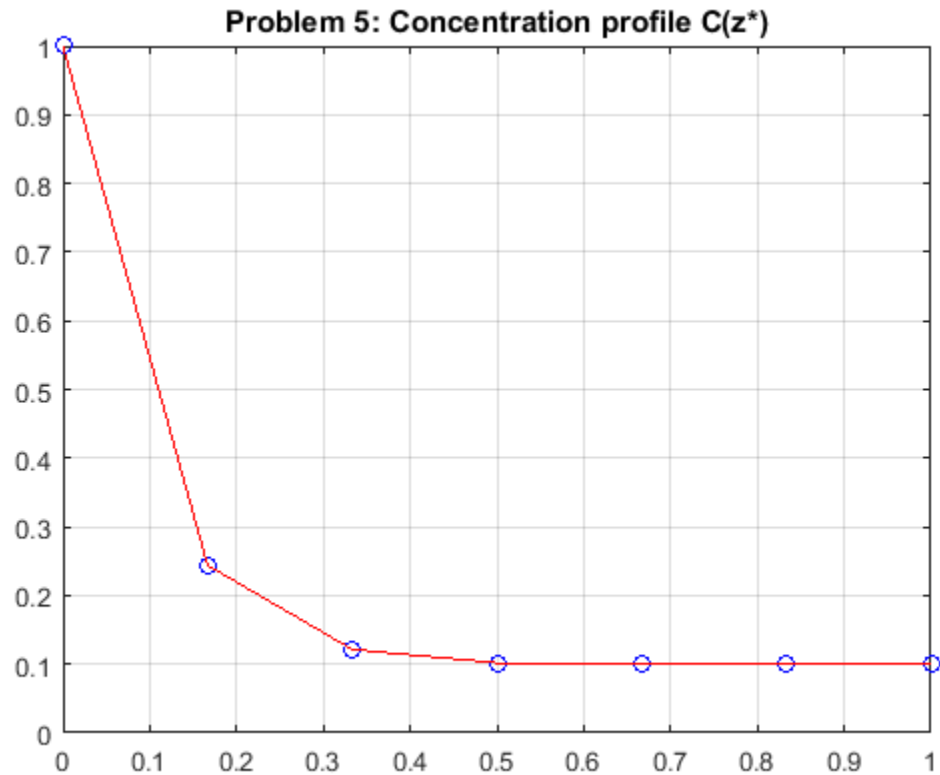
options = optimoptions('fsolve','Display','off');
wo = ones(7,1);
% Calling the fsolve function
w = fsolve(@eval5,wo,options);
disp('The concentrations are')
disp(w)

% For plotting
plot(z,w,'bo')
hold on
plot(z,w,'r')
ylim([0 1])
grid on
title('Problem 5: Concentration profile C(z*)')

```

```
hold off
```

```
=====
Q5
```



Problem 6

```
disp('=====')
disp('Q6')
%Given data and calculated values from it
R = 1.3;
h1 = 0.001;
eta = 0.36;
A = 15;
k = 0.0034;
betasqr = ((R^2)*h1*A)/k/(1-eta)
alpha = 0;
h = 0.0025;
x = (0:h:1)';
x(1,1) = 0.00000001;
p = (-1./x);
q = betasqr^2;
d = 2 + (h^2)*q
r = 0;
n = (1-0)/h;
u = -1+(h/2)*p;
```

```

l = -1-(h/2)*p;

% For matrix A
A(1,1) = d;
A(1,2) = -2;
A(2,1) = l(1,1);
A(n,n+1) = u(n,1);
A(n+1,n+1) = 1;
for i = (2:n)
    A(i,i) = d;
    A(i,i+1) = u(i,1);
    A(i+1,i) = l(i,1);
end
A(n+1,n) = 0;

%For constant vector
b = zeros(n+1,1);
b(1,1) = -h^2*r+2*h*l(1,1)*alpha;
b(n+1,1) = 1;
for i = (2:n)
    b(i,1) = -(h^2)*r;
end

%Solving the system
sol = zeros(n+1,1);
sol = inv(A)*b;

%For plotting
for i = (1:n+1)
    x(i,1) = i*h;
end
figure();
plot(x,sol,'b. ');
xlabel('x');
ylabel('tau(x)');
title('Problme 6: Steady statetemperature profile');
grid on
legend('tau(x)')

=====
Q6

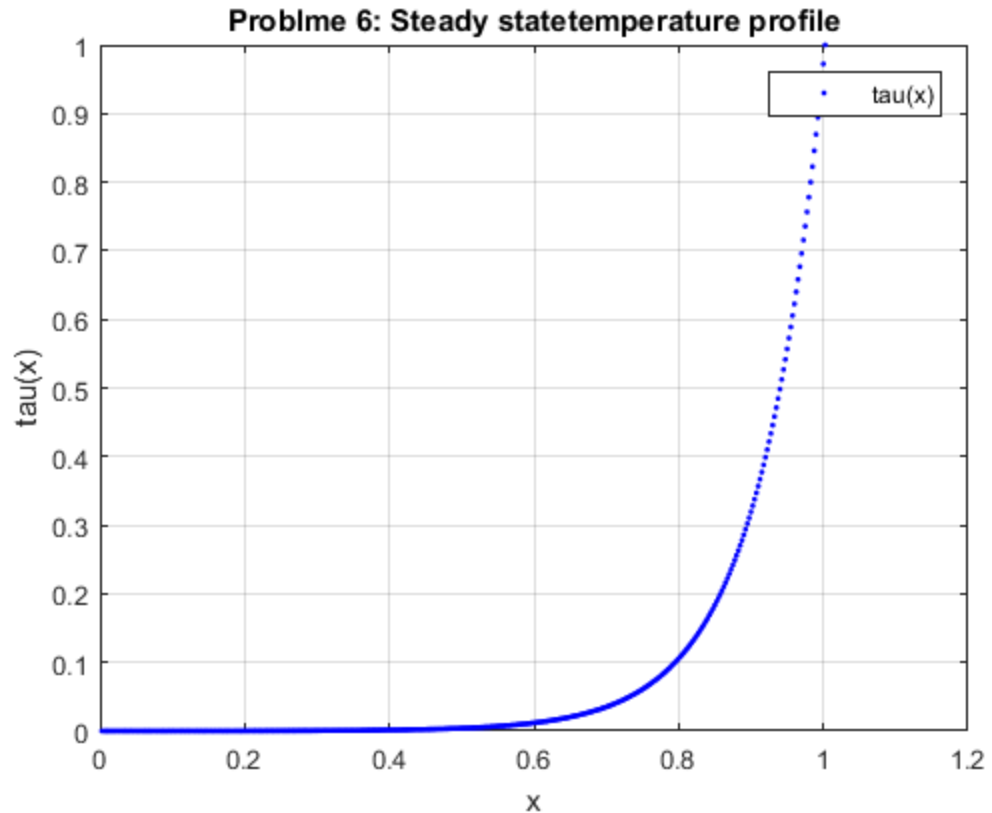
betasqr =

    11.6498

d =

    2.0008

```



Problem 7

```
%Q7
disp('=====')
disp('Q7')

%Given data and values calculated from it
N = 6;
w0 = linspace(1,0.1,N+1);
options = optimoptions('fsolve', 'Display', 'Off');

%Calling fsolve to solve the non-linear system
F = fsolve(@eval7, w0, options);

disp('The Concentrations are as follows:');
disp(F);

%For plotting
zspan = linspace(0, 1, N+1);
figure();
plot(zspan,F, 'ro');
hold on
plot(zspan,F, 'b');
xlabel('Z');
ylabel('Theta(z)');
```

```

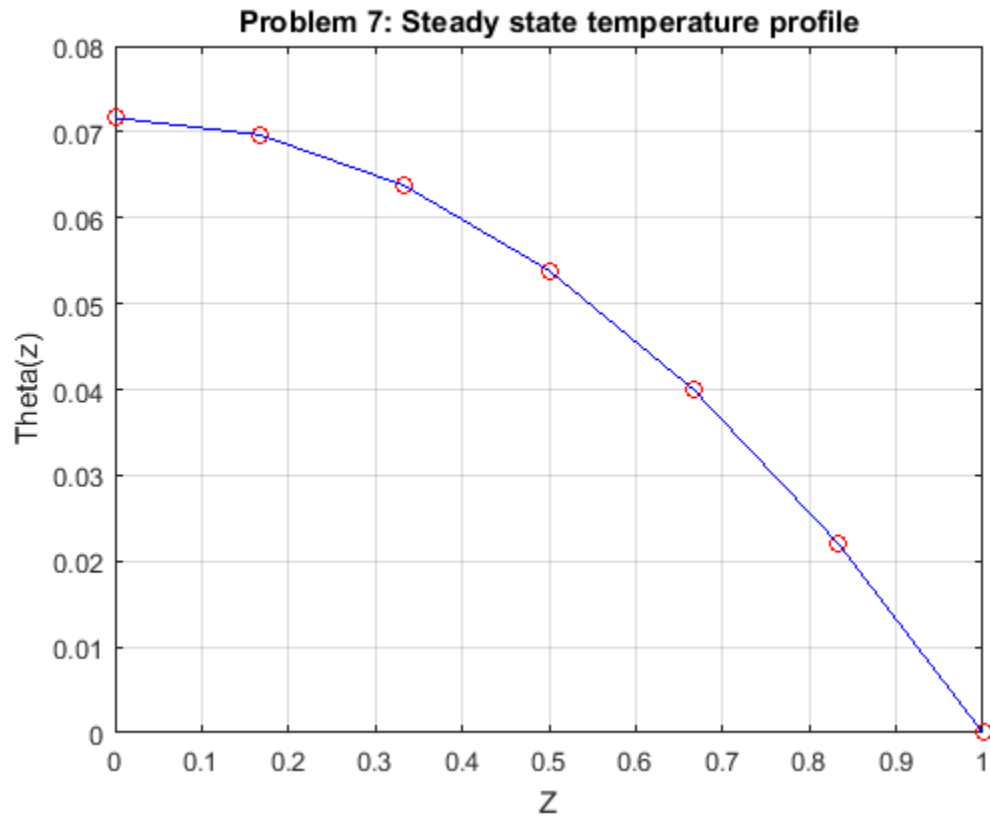
grid on
title('Problem 7: Steady state temperature profile');
hold off

```

```

=====
Q7

```



```

%{
=====
                        OPTIMIZATION PROBLEMS
=====
%}

```

Problem 8

```

%Q8
disp('=====')
disp('Q8')
%Given values
Mw = 627;
C = [5*(10^-5) (10^-4) 4*(10^-4) 5*(10^-4) 1*10^-3 0.002 0.003];
Cbulk = C./Mw;
gamma = [36.42 33.72 30.63 27.45 24.76 22.30 19.71];

%Using fminsearch to find the optimum value by passing appropriate
initial

```

```

%values
init = [4 4];
[f, fval] = fminsearch(@eval81, init);
disp(' The value of T and a are as follows:');
disp(f);

%For plotting
figure();
subplot(2,1,1)
hold all
plot(Cbulk,gamma,'yo')
plot(Cbulk,eval82(f),'r')
xlabel('Cbulk')
ylabel('Value of gamma')
title('Problem 8: Plot of calculated vs observed values of gamma')
legend('Given','Calculated')
grid on
hold off

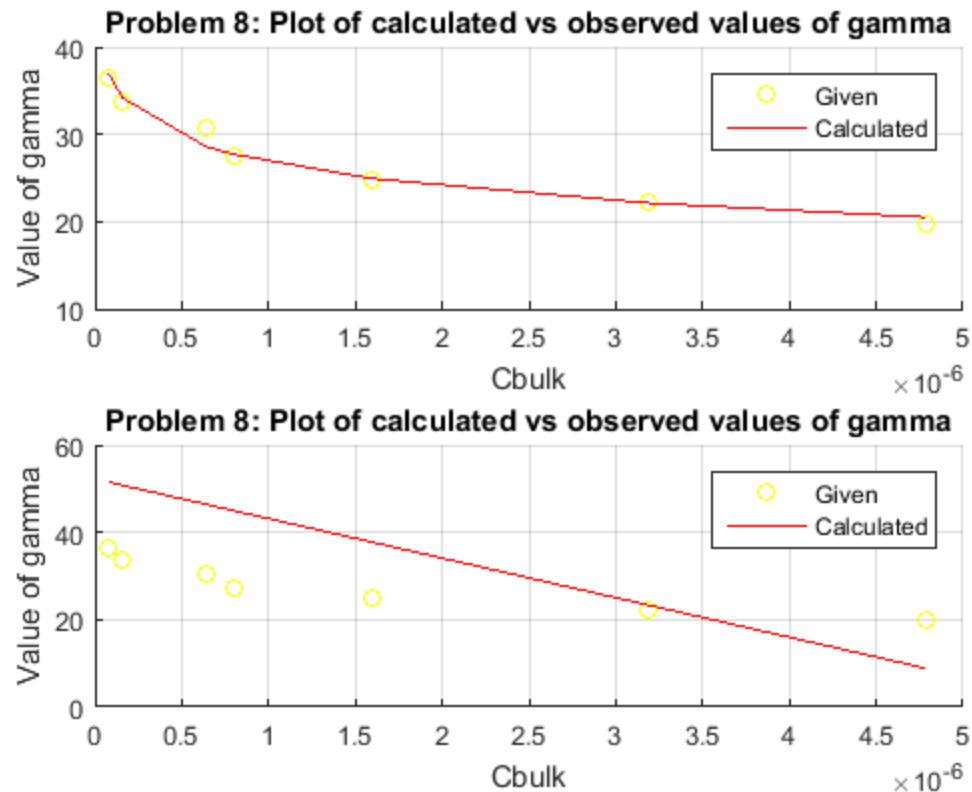
%Using fminsearch to find the optimum value by passing random initial
%values
init1 = [150 150];
[f1, fval] = fminsearch(@eval81, init1);
disp(' The value of T and a are as follows:');
disp(f1);

subplot(2,1,2)
hold all
plot(Cbulk,gamma,'yo')
plot(Cbulk,eval82(f1),'r')
xlabel('Cbulk')
ylabel('Value of gamma')
title('Problem 8: Plot of calculated vs observed values of gamma')
legend('Given','Calculated')
grid on
hold off

%{
As seen in the graph, the plot when we have initial values close to
the
actual value, we get a better plot than the plot we get when the
initial
values are way off the actual values.
From the graph we can be enough confident that the answer we have is
correct.
%}

=====
Q8

```



Problem 9

```
%Q9
disp('=====')
disp('Q9')
%{
The height and width of the room is 10ft each
So both the triangles below the ladder are congruent
Thus the angle is congruent too
Consider the lower part of ladder as x and upper part as y
Thus (approx) a = 10/sin(x) and b = 10/cos(x)
We have our objective function as
Minimize(L): a + b (or (10/sin(x))+10/cos(x)) )
Here we will use fminsearch to find the optimum value of x
Once we have the optimum value of x, we put it in the objective
function to
get the minimum value of length of ladder which does not interfere with
the
room
%}
%Initial value of angle between ladder and ground
theta0 = pi/4;
%Calling the fminunc (unconstraint) to find the optimum value of theta
x = fminunc(@eval9, theta0);
```

```

disp('The value of angle between ladder and wall is')
disp(x)

a = 10/ sin(x)
b = 10 / cos(x)

% Objective function
optiLength = (a + b);
e = [num2str(optiLength,5), ' feet']
disp('The optimum length of the ladder is')
disp(e)

=====
Q9

end

%{
=====
                        FUNCTION DEFINITIONS
=====
%}

```

Functions for 3

```

%Function for problem 3
%Function to describe the system
function f1 = eval3a(psi,t)
dt = t(2);
phisql = 0.01;
d2t =(phisql * t(1))- ((2/psi)* t(2));
f1 = [dt ; d2t];
end

%Function to describe the system
function f2 = eval3b(psi,t)
dt = t(2);
phisql = 10;
d2t =(phisql * t(1))- ((2/psi)* t(2));
f2 = [dt ; d2t];
end

\theta(\xi) for phi^2 = 0.01
0.9983
0.9983
0.9984
0.9984
0.9984
0.9985
0.9985
0.9986
0.9986
0.9987
0.9988
0.9989

```

```

0.9990
0.9991
0.9992
0.9994
0.9995
0.9997
0.9998
1.0000

```

Shooting method when $\psi^2 = 10$
 $\theta(\xi)$ for $\phi^2 = 0.01$

```

0.9983
1.0043
1.0197
1.0445
1.0791
1.1241
1.1802
1.2483
1.3297
1.4256
1.5376
1.6677
1.8182
1.9916
2.1909
2.4196
2.6819
2.9824
3.3265
3.7206

```

Q3b

Finite differences method, $\psi^2 = 0.01$

The co-eff matrix:

| | | | | | | |
|---------|---------|---------|---------|---------|---------|---------|
| 2.0003 | -2.0000 | 0 | 0 | 0 | 0 | 0 |
| -0.0476 | 2.0003 | -1.9524 | 0 | 0 | 0 | 0 |
| 0 | -0.5098 | 2.0003 | -1.4902 | 0 | 0 | 0 |
| 0 | 0 | -0.6700 | 2.0003 | -1.3300 | 0 | 0 |
| 0 | 0 | 0 | -0.7512 | 2.0003 | -1.2488 | 0 |
| 0 | 0 | 0 | 0 | -0.8004 | 2.0003 | -1.1996 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1.0000 |

The constant matrix:

```

0
0
0
0
0
0
0
1

```

Approximate solution with $\phi^2 = 0.01$

```

0.9982
0.9984
0.9985
0.9987
0.9991
0.9995
1.0000

```

The co-eff matrix:

```

2.2778 -2.0000 0 0 0 0 0
-0.0476 2.2778 -1.9524 0 0 0 0
0 -0.5098 2.2778 -1.4902 0 0 0
0 0 -0.6700 2.2778 -1.3300 0 0
0 0 0 -0.7512 2.2778 -1.2488 0
0 0 0 0 -0.8004 2.2778 -1.1996
0 0 0 0 0 0 1.0000

```

The constant vector:

```

0
0
0
0
0
0
0
1

```

Approximate solution with $\phi^2 = 10$

```

0.2493
0.2839
0.3251
0.3998
0.5210
0.7097
1.0000

```

Q3c

The gradient of concentration with $\phi^2 = 0.01$ is :
0.0033

The gradient of concentration with $\phi^2 = 10$ is :
3.7206

Function for 5

Function for problem 5

```

%Function to describe the system
function F = eval5(C)
D = 10; % um2/sec
L = 1*10^(3); % um (micrometer)
v = 0.1; % um/sec
k = 5*10^(-5);

```

```

N = 7;
h= 1/N;

F = [ (C(1) - 1);
      (( C(1)-( 2*C(2) )+ C(3) )/ (h^2))- (k * ( L^2 ) / D *
      ( C(2)^2)) + (v * L / D * ((C(3) - C(1))/(2 * h)));
      (( C(2)-( 2*C(3) )+ C(4) )/ (h^2))- (k * ( L^2 ) / D *
      ( C(3)^2)) + (v * L / D * ((C(4) - C(2))/(2 * h)));
      (( C(3)-( 2*C(4) )+ C(5) )/ (h^2))- (k * ( L^2 ) / D *
      ( C(4)^2)) + (v * L / D * ((C(5) - C(3))/(2 * h)));
      (( C(4)-( 2*C(5) )+ C(6) )/ (h^2))- (k * ( L^2 ) / D *
      ( C(5)^2)) + (v * L / D * ((C(6) - C(4))/(2 * h)));
      (( C(5)-( 2*C(6) )+ C(7) )/ (h^2))- (k * ( L^2 ) / D *
      ( C(6)^2)) + (v * L / D * ((C(7) - C(5))/(2 * h)));
      (C(7) - 0.1)];
end

```

The concentrations are

```

1.0000
0.2439
0.1214
0.1019
0.0992
0.0994
0.1000

```

Function for 7

```

%Function for problem 7
%Function to describe the system
function f = eval7(x)
B = 0.6;
phisquare = 0.25;
gamma = 30;
N = 6;
h = (1 - 0)/N;

% Formation of matrix
f = [((x(2) - ( 2*x(1) ) + x(2) )/(h^2)) + (B * phisquare *(1 -
( x(1)/B )) * exp((gamma * x(1) )/(gamma + x(1) )));
      ((x(1) - ( 2*x(2) ) + x(3) )/(h^2)) + (B * phisquare *(1 -
( x(2)/B )) * exp((gamma * x(2) )/(gamma + x(2) )));
      ((x(2) - ( 2*x(3) ) + x(4) )/(h^2)) + (B * phisquare *(1 -
( x(3)/B )) * exp((gamma * x(3) )/(gamma + x(3) )));
      ((x(3) - ( 2*x(4) ) + x(5) )/(h^2)) + (B * phisquare *(1 -
( x(4)/B )) * exp((gamma * x(4) )/(gamma + x(4) )));
      ((x(4) - ( 2*x(5) ) + x(6) )/(h^2)) + (B * phisquare *(1 -
( x(5)/B )) * exp((gamma * x(5) )/(gamma + x(5) )));
      ((x(5) - ( 2*x(6) ) + x(7) )/(h^2)) + (B * phisquare *(1 -
( x(6)/B )) * exp((gamma * x(6) )/(gamma + x(6) )));
      x(7) - 0];
end

```

The Concentrations are as follows:

| | | | | | | |
|--------|--------|--------|--------|--------|--------|---|
| 0.0716 | 0.0697 | 0.0638 | 0.0539 | 0.0400 | 0.0220 | 0 |
|--------|--------|--------|--------|--------|--------|---|

Function for 8

```
%Function for question8
%Function to describe the system
function f = eval81(g)
gamma0 = 52.2; % mN/m
R = 8314; % cm^3 kPa K^?1 mol^?1
T = 300; % K
Mw = 627; % g/mol

Cbulk = [5*(10^-5) (10^-4) 4*(10^-4) 5*(10^-4) 1*10^-3 0.002
0.003];
Cbulk1 = Cbulk./Mw;
gamma = [36.42 33.72 30.63 27.45 24.76 22.30 19.71];

f = 0.5 * (sum( ((gamma0 + (R .* T .* g(1) .* log (1 - ( Cbulk1 ./
( Cbulk1 + g(2) ))))) - gamma).^2));
end
%Function to describe the system
function v = eval82(g)
gamma0 = 52.2 ;
R = 8314;
T = 300;
Mw = 627;

Cbulk = [5*(10^-5) (10^-4) 4*(10^-4) 5*(10^-4) 1*10^-3 0.002
0.003];
Cbulk1 = Cbulk./Mw;
gamma = [36.42 33.72 30.63 27.45 24.76 22.30 19.71];

v = (gamma0 + (R .* T .* g(1) .* log (1 - ( Cbulk1 ./ ( Cbulk1 +
g(2) )))))));
end
```

The value of T and a are as follows:

*1.0e-05 **

0.1616 0.0002

The value of T and a are as follows:

228.2257 62.8462

Function for 9

```
%Function for question 9
%Function to describe the system
function F = eval9(t)
F = ( 10 / sin(t) ) + ( 10 / cos(t) );
```

end

*Warning: Gradient must be provided for trust-region algorithm;
using quasi-newton algorithm instead.*

Initial point is a local minimum.

*Optimization completed because the size of the gradient at the initial
point
is less than the default value of the function tolerance.*

*The value of angle between ladder and wall is
0.7854*

*a =
14.1421*

*b =
14.1421*

*e =
28.284 feet*

*The optimum length of the ladder is
28.284 feet*

Published with MATLAB® R2015a