MATH 131 Final Exam

Name: Steven Choi

Date: May 9, 2022

PROBLEM 1:

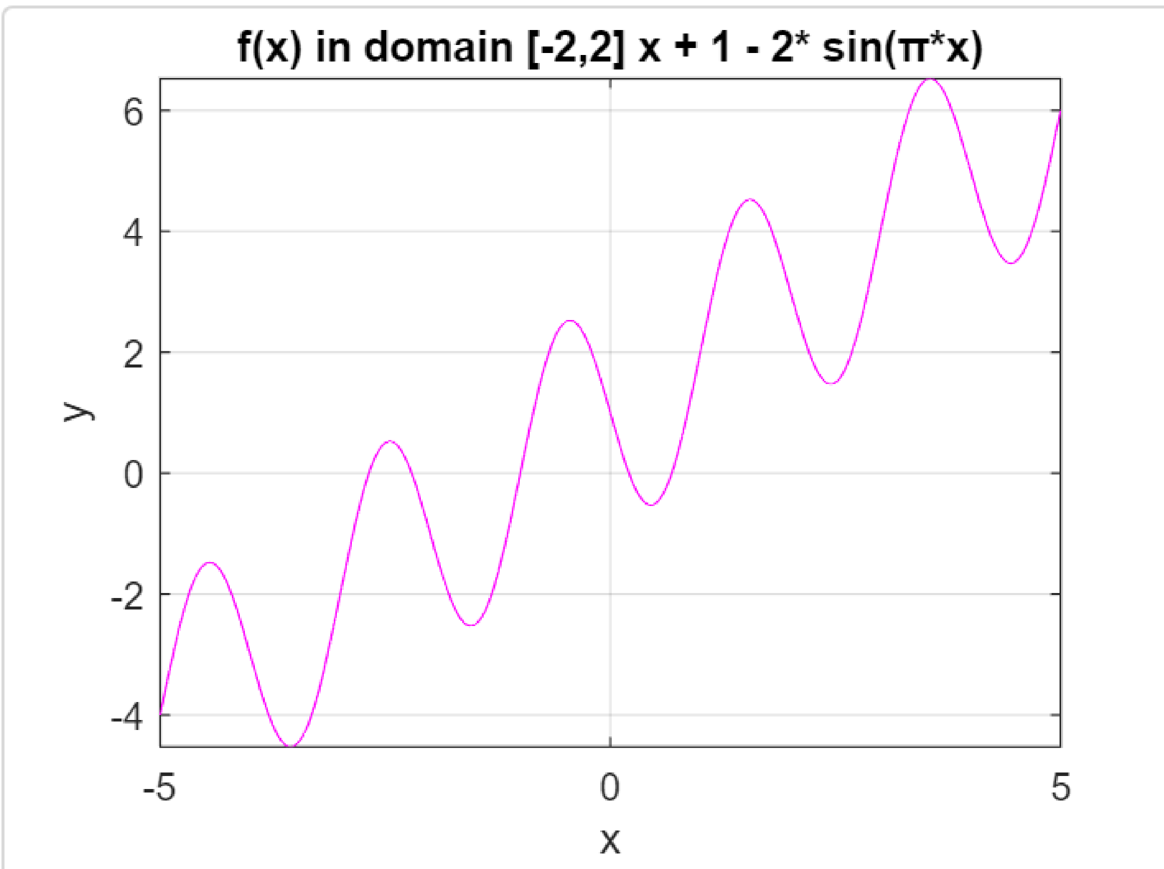
1. *25 points*. Consider the following root-finding problem:

$$f(x) = x + 1 - 2\sin(\pi x) = 0. \tag{1}$$

Please use what you have learned and answer the following questions:

a. Plot f(x) in domain [−2, 2].

i. Plot:



ii.

CODE:

```
% MATH 131 Final Q1a
```

```
clear all
clc

fplot(@(x) x + 1 - 2* sin(pi*x),'m' )
hold on
grid on
title('x + 1 - 2* sin(pi*x')
xlabel('x')
ylabel('y')
```

b. Use the bisection method to approximate the first negative solution, i.e the negative root that is closest to the origin. The accuracy must be of the order 10^{-4}.
   i. SOLUTION BELOW:

c. Reformulate equation (1) into a fixed point problem. Then use the Fixed-Point theorem (Theorem 2.4 in the textbook) to determine whether the Fixed-Point iteration converges at the first negative root. Note: you do not need to numerically implement the fixed point iteration, a theoretical analysis of its convergence would suffice.
   i. SOLUTION:

CODE:

```
% MATH131 Final Q1

clear all
clc

%f = @(x)(x+1)-(2*sin(pi*x)); % function
f = @(x)(3*x^2 - 2*exp(x));
a = -1;
b = 1;
tol = 10^-4;
N = 10^5;
[c,n,error,theerror] = bisection_method(f,a,b,tol,N);
c = fzero(f,[-1,1]);

x0 = -1;
x1 = 1;
[c,n,error1,theerror1] = fixed_point_iteration(f,x0,x1,tol,N);

% plot the graph
semilogy(1:length(theerror),theerror,'ro-', 1:length(theerror1),theerror1,
```

```
'b^-')
grid on
title('Root Finding (x + 1 -2sin(pix) = 0')
xlabel('# of Iterations')
ylabel('Root Finding Problems')
legend({'Bisection Method','Fixed-Point 5'},'Location','best')


function [c,n,error,theerror] = bisection_method(f,a,b,tol,N) %function for
bisection method

n = 0;
error = 1;
fa = f(a);
fb = f(b);

% if bisection method works or not display error
if fa * fb > 0 || a > b
    disp('The bisection method does not work!')
    c = [];
    error = inf;
    n = 0;
else % if works, continute with rooting
    while error > tol && n < N
        c = (a + b) / 2;
        fc = f(c);
        error = abs(b - a) / 2;
        % keep continuting until it reaches the closes to error
        if ( error >= tol)
            if fb * fc > 0
                b = c;
                fb = fc;
            elseif fa * fc > 0
                a = c;
                fa = fc;
            end
            n = n + 1;
            theerror(n) = error;
        end
    end
end

end
```

```
function [c,n,error1,theerror1] = fixed_point_iteration(f,x0,x1,tol,N) %
function for secant method
    p1 = f(x1); %  two initial approximations supplied
    p0 = f(x0);
    n = 0;

% Divide with x until it intersects the x axis at third point
    for i = 1:N
        c = x1 - (p1 * (x1 - x0)) / (p1 - p0);
        error1 = abs(c - x1);
        if error1 < tol
            break;
        end
        x0 = x1; % This third point and the second point are the again used
as two initial approximations to find the fourth point.
        p0 = p1;
        x1 = c;
        p1 = f(c);
        n = n + 1;
        theerror1(n) = error1;
    end

end
```

PROBLEM 2:

2. *25 points*. Consider the following improper integral:

$$\int_{1}^{+\infty} \frac{1}{x^2+1}dx \tag{2}$$

a. Please use the following indefinite integral formula to find the exact value of (2):

$$\int \frac{1}{x^2+1}dx = arctan(x) + C \tag{3}$$

i. Solution:

$$\text{Let } I = \int_1^{+\infty} \frac{1}{x^2+1}\,dx$$

$$\text{Given in (3): } \int \frac{1}{x^2+1}\,dx = arctanx + C$$

$$\text{Now, } I = \int_1^{+\infty} \frac{1}{x^2+1}\,dx = [arctanx]_1^{+\infty}$$

$$\Rightarrow [\arctan\infty - \arctan(1)]$$

$$\boxed{\Rightarrow \frac{\pi}{4} = 0.785398}$$

/

CODE:

```
% M131 Final Q1a

clear all
clc

syms a x
f = 1/(x^2+1);
F = int(f, 1, inf)
```

OUTPUT:

```
F =
```

$$\frac{\pi}{4}$$

b. Now we want to use the composite Simpson's rule to numerically evaluate (2). To this end, finish the following tasks: [1] Use the indefinite integral formula (3) and the exact integral value you obtained in problem (a) to determine a constant C_1 such that

$$\left| \int_1^{C_1} \frac{1}{x^2+1}dx - \int_1^{+\infty} \frac{1}{x^2+1}dx \right| \leq 0.01 \tag{4}$$

1.

Note that there are many constants to make inequality (4) hold. You can arbitrarily choose one of them. [2] Use the composite Simpson's rule to numerically calculate the *definite integral*:

$$\int_1^{C_1} \frac{1}{x^2+1}dx \tag{5}$$

where C1 is the constant you determined in step [1], and then compare it with the exact result:

$$\int_1^{C_1} \frac{1}{x^2+1}dx = arctan(C_1) - arctan(1). \tag{6}$$

The accuracy must be of the order 10^{-3}.

ii. Solution:

First find constant C such that:

$$\left|\int_1^{C_1} \frac{1}{x^2+1}dx - \int_1^{+\infty} \frac{1}{x^2+1}dx\right| \leq 0.01$$

That is...

$$\left|\int_1^{+\infty} \frac{1}{x^2+1} - \frac{\pi}{4}\right| \leq 0.01 \Rightarrow \left|arctanC - arctan1 - \frac{\pi}{4}\right| \leq 0.01$$

=> $\left|arctanC - \frac{\pi}{2}\right| \leq 0.01$

Then we take C = 10:

$arctanC = 1.471 \Rightarrow \arctan C - \frac{\pi}{2} = -0.099 \Rightarrow \boxed{\boldsymbol{C = 10}}$

Now using Simpson's rule to find definite integral $\int_1^{10} \frac{dx}{1+x^2}$.

$$\int_1^{10} \frac{dx}{1+x^2} \approx \frac{h}{3} f(x_0) + f(x_{10}) + 4\{f(x_1) + f(x_3)$$

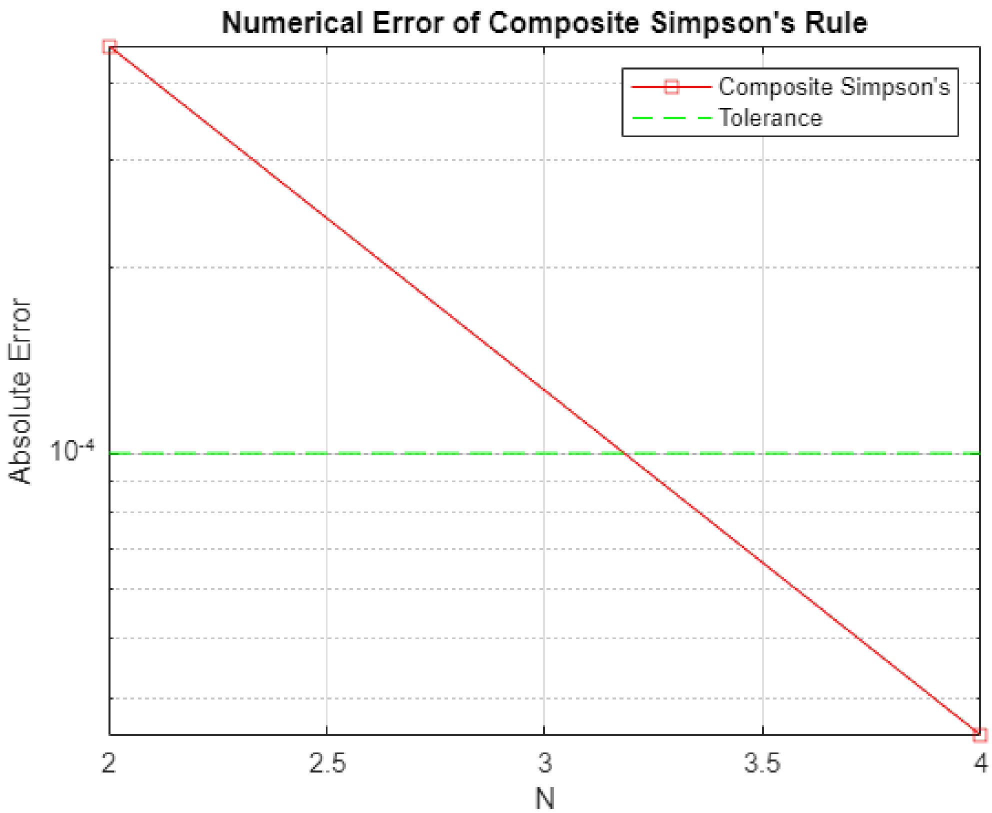
$= \frac{0.9}{3}(0.5 + 0.9 + 0.23 + 0.27 + 0.09 + 0.128 + 0.05 + 0.07 + 0.029 + 0.047 + 0.009) = \boxed{\mathbf{0.6969}}$

CODE:

```
%MATH131 Final Q2b

clc;
close all;
clear variables;

% Initial variables

f1 = @(x) log(2*x); % function
a = 1; % lower limit
b = 2; % upper limit

exact = 2*log(4)- log(2) - 1; % exact integralvalue

TOL = 10^-4; % given tolerance

% Initializing
n1 = 0;
err = inf;
i = 1;

while err > TOL % while loop to go though error > tolerance
    n1 = n1 + 2;
    N(i) = n1; % stores n
    I = composite_simpsons(f1, a, b, n1);
    err (i) = abs(exact - I);
    i = i + 1; % i counts the iterations using while loop
end



t = zeros(1, length(N)) + TOL;  % plot tolerance
semilogy(N, err,'r-s', N, t,'--g')
grid on
title("Numerical Error of Composite Simpson's Rule")
xlabel('N')
ylabel('Absolute Error')
legend({"Composite Simpson's",'Tolerance'})



%% Ploting answer
```

```
function I = composite_simpsons(f,a,b,n)

    h = (b - a) / n;
    s1 = 0; % variable odd
    s2 = 0; % varaible even

    for i=1:2:n-1 % for loop
        s1 = s1 + f(a + i*h);
    end

    for i=2:2:n-2
        s2 = s2 + f(a + i*h);
    end

    I = (h / 3) * (f(a) + (2 * s2) + (4 * s1) + f(b));

end
```

OUTPUT:



Speculation for Plot: 'Numerical Error of Composite Simpson's Rule' looking at the graph we can see that the composite simpson's line is very linear and the tolerance is also very linear on the x-axis.

PROBLEM 3:

3. 25 points. Consider the following ordinary differential equation:

$$\frac{dy}{dt} = 2\frac{y}{t} - t^2y^2 \tag{7}$$

The analytic solution to (7) is given by

$$y = \frac{5t^2}{t^5 + C} \tag{8}$$

where C is a constant to be determined. Please use what you have learned and answer the following questions:

a. Given the initial condition y(1) = 1, determine the constant C and the exact form of the analytic solution (8).

Given: $\frac{dy}{dt} = 2\frac{y}{t} - t^2y^2$

& General solution of (1)

$$y = \frac{5t^2}{t^5 + C}$$

Applying initial condition:

We get $y(1) = 1 = \frac{5}{1+C}$

$= C + 1 = 1\ 5$

C = 4

Exact Solution of (1)

i. Solution: $\boxed{y(x) = \frac{5^2}{t^5+4}}$

b. Use the Adams-Bashforth Two-Step Explicit Method, i.e.

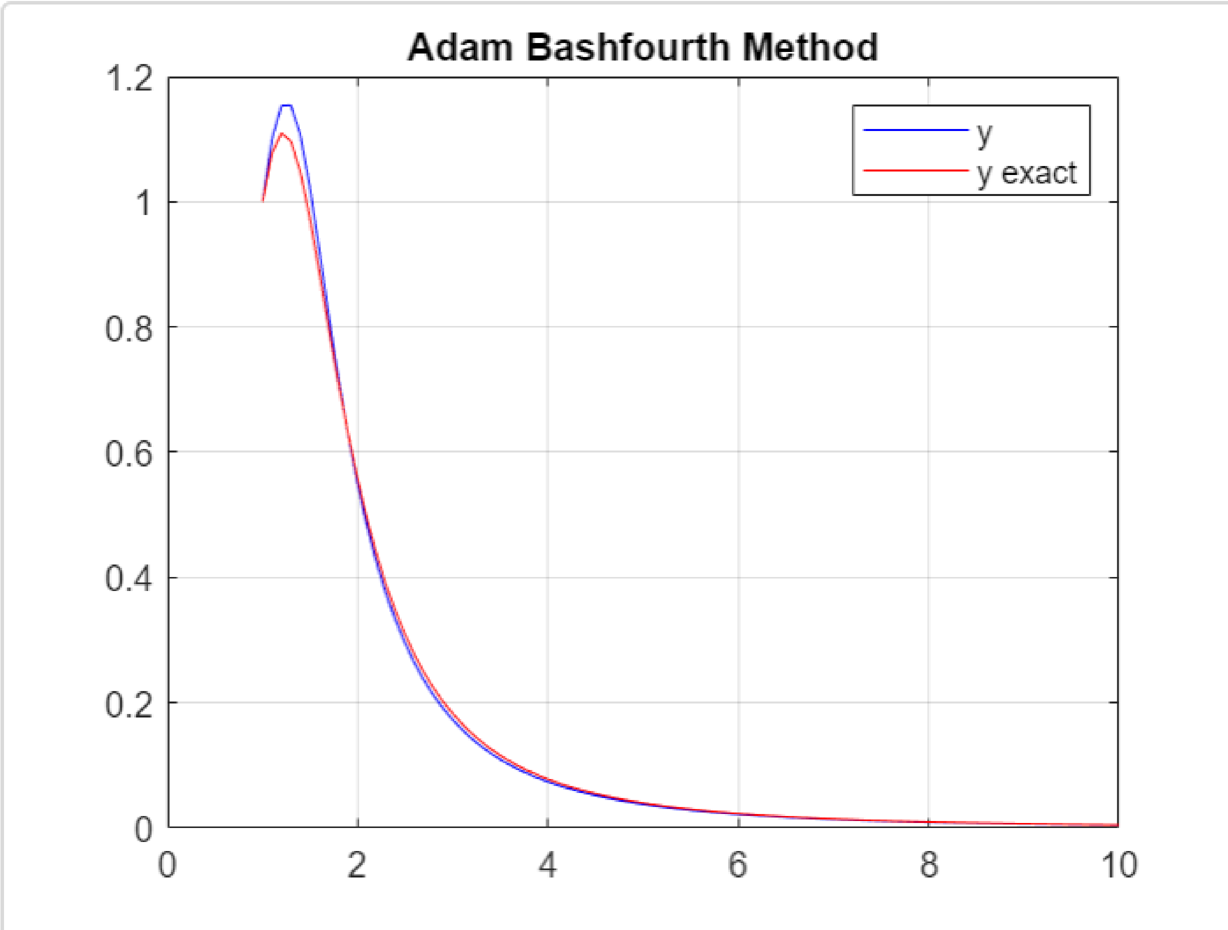
$$\omega_0 = \alpha, \omega_1 = \alpha_1$$

$$\omega_{i+1} = \omega_i + \frac{h}{2}(3f(t_i, \omega_i) - f(t_{i-1}, \omega_{i-1}))$$

to numerically solve the initial value problem with y(1) = 1. Note that here α = y(1), α1 = y(1 + h), h is the step size, $f(t, y) = 2y/t - t^2y^2$ and $\omega i = y(t_i)$. To show your numerical result, please generate plots for the numerical solution and the exact solution (8) and put them in the same figure.

Solution:

i. Plot / OUTPUT:



Ii. **Speculation for Plot**: Looking at this Adam Bashforuth Method Plot, I can see that the y and y exact is very similar in terms of linear trendline, however near the 1-1.2 where the line arcs with a slight indifferent result. I think the reason for this

CODE:

```
% MATH131 Final Exam Q3b

clear all
clc

% Define funtion Right-Hand side of ODE
```

```
f = @(t,y)2 * (y/t) - t^2*y^2;

% Define perameters
h = 0.1;
t(1) = 1;
y(1) = 1;
n = (10 - 1)/h;

% Using Adam Bashfourth Method

for i = 1:n
    % Update t
    t(i + 1) = t(i) + h;

    % Use of Euler method find y1
    y(i + 1) = y(i) + h*f(t(i),y(i));

    % Now use method which is given here i replace by i+1
    y(i + 2) = y(i + 1) + (h/2)*(3*f(t(i + 1), y(i + 1)) - f(t(i), y(i)));
end

% Plotting
t = linspace(1,10,92); % range 1 to 6 between 92 points
y_exact = (5*t.^2)./(t.^5 + 4);
plot (t, y, 'b')
hold on
plot(t,y_exact, 'r');
grid on
title('Adam Bashfourth Method')
legend('y','y-exact')
```

PROBLEM 4:

4. *Bonus problem, 25 points*. Considering the following linear equation Ax = b, where

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 6 & 1 \\ 2 & 0 & 3 \end{bmatrix} \qquad b = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} \tag{9}$$

and finish the following tasks:

a. Solve Ax = b by hand and get the exact solution x_{ex}.

Given: Linear equation in Ax = b

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 6 & 1 \\ 2 & 0 & 3 \end{bmatrix} \ \& \ b = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}$$

Ax = b

$$\begin{bmatrix} 1 & 0 & 0 \\ 2 & 6 & 1 \\ 2 & 0 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}$$

$$\boxed{x_1 = -1}$$

$\& \ 2x_1 + 6x_2 + x_3 = 1$

$=> 2(-1) + 6x_2 + x_3 = 1$

$=> -2 + 6x_2 + x_3 = 1$

$=> 6x_2 + x_3 = 3$

$=> x_3 = 3 - 6x_2$

$\& \ 2x_1 + 3x_3 = 1$

$=> 2(-1) + 3x_3 = 1$

$=> 2 + 3x_3 = 1$

$=> 3x_3 = 3$

$$\boxed{=> x_3 = 1}$$

$x_3 = 3 - 6x_2$

$1 = 3 - 6x_2$

$6x_2 = 3 - 1$

$6x_2 = 2$

$$\boxed{x_2 = \frac{1}{3}}$$

i.

$$\boxed{\text{Exact Solution:} \quad x_{ex} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -1 \\ \frac{1}{3} \\ 1 \end{bmatrix}}$$

b. Solve Ax = b numerically using the Gaussian elimination and backward substitution to get a numerical solution.

Gaussian Elimination Method: Ax = B

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 6 & 1 \\ 2 & 0 & 3 \end{bmatrix} \;\&\; b = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}$$

$$R_2 \leftrightarrow R_2 - 2R_1$$
$$R_2 \leftrightarrow R_3 - 2R_1$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 6 & 1 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -1 \\ 3 \\ 3 \end{bmatrix}$$

=> $3x_3 = 3$

$\Rightarrow$ $x_3 = 1$

& $6x_2 + x_3 = 3$

=> $6x_2 + 1 = 3$

=> $6x_2 = 2$

$x_2 = \frac{1}{3}$ & $x_1 = -1$

i. $\boxed{X = \begin{bmatrix} -1 \\ \frac{1}{3} \\ 1 \end{bmatrix}}$

c. Use Theorem 7.21 in the textbook and determine whether the Jacobi iterative method converges for this Ax = b problem. If so, then calculate the matrix TJ and vector c_J explicitly in the Jacobi iterative scheme:

$$x^{(k+1)} = T_J x^{(k)} + c_J \tag{10}$$

and then use (10) to get a numerical solution x_{nu}. Please make sure that the l2-norm

$$\|x_{ex} - x_{nu}\|_2 < 10^{-3}.$$

I. Solution:

Jacobi Iterative Method:

|1| > |0| |0|

|6| > 2|| |1|

|3| > 2|| |0|

Jacobi Iterative Method Converges

& $x^{(k+1)} = T_J x^{(k)} + c_J$

Where $T_J = -D^{-1}(L+U)$& $c_J = D^{-1}b$

L = $\begin{bmatrix} 0 & 0 & 0 \\ 2 & 0 & 0 \\ 2 & 0 & 0 \end{bmatrix}$ $\& \; U = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$

L + U = $\begin{bmatrix} 0 & 0 & 0 \\ 2 & 0 & 1 \\ 2 & 0 & 0 \end{bmatrix}$ $\& \; D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 3 \end{bmatrix}$

$$D^{-1} = \frac{1}{|D|} Adj(D)$$

|D| = $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 3 \end{bmatrix} = (1)(18-0) = 18$ & Adj (D) = $\begin{bmatrix} 18 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 6 \end{bmatrix}$

$$D^{-1} = \frac{1}{18} \begin{bmatrix} 18 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 6 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/6 & 0 \\ 0 & 0 & 1/3 \end{bmatrix}$$

$$T_J = D^{-1}(L+U)$$

$$T_J = D^{-1}(L+U)$$

$$T_J = -\begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{6} & 0 \\ 0 & 0 & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 2 & 0 & 1 \\ 2 & 0 & 0 \end{bmatrix}$$

$$T_J = -\begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{6} \\ \frac{2}{3} & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ -\frac{1}{3} & 0 & -\frac{1}{6} \\ -\frac{2}{3} & 0 & 0 \end{bmatrix}$$

$$\& \, D^{-1}b = (J = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{6} & 0 \\ 0 & 0 & \frac{1}{3} \end{bmatrix} \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ \frac{1}{6} \\ \frac{1}{3} \end{bmatrix}$$

$$x^{(k+1)} = \begin{bmatrix} 0 & 0 & 0 \\ -\frac{1}{3} & 0 & -\frac{1}{6} \\ -\frac{2}{3} & 0 & 0 \end{bmatrix} x^{(k)} + \begin{bmatrix} -1 \\ \frac{1}{6} \\ \frac{1}{3} \end{bmatrix}$$

Take $(x)^{(6)} = (0\ 0\ 0)$

Put k = 0

$$x^{(1)} = \begin{bmatrix} 0 & 0 & 0 \\ -\frac{1}{3} & 0 & -\frac{1}{6} \\ -\frac{2}{3} & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -1 \\ \frac{1}{6} \\ \frac{1}{3} \end{bmatrix} = \begin{bmatrix} -1 \\ 0.4444 \\ 1 \end{bmatrix}$$

$$\boxed{x = \begin{bmatrix} -1 \\ 0.4444 \\ 1 \end{bmatrix}}$$

CODE:

```
% MATH131 Final Exam Q4
clc
clear all

% Answer for x
x = [-1 0.2 0.4; 2 6 1; 2 0 3];
y = [-1;1;0];

x_exact = mldivide(x,y);

% Answer for y
xy = [x y];
xy(3,:) = xy(2,:)-xy(3,:)
xy(2,:) = 2*xy(1,:)+ xy(2,:)
xy(3,:) = 2*xy(2,:)+1.8*xy(3,:)
Xgauss = linsolve(xy(:,1:3),xy(:,4))
l2=sqrt((x_exact(1))^2+(x_exact(2))^2+(x_exact(3))^2
)-sqrt((Xgauss(1))^2+(Xgauss(2))^2+(Xgauss(3))^2 );

% Answer for c

D = diag(diag(x));
R = x-D; %L+U

% Specify # of iterations to operate
itr = 15;

% Specified, the beginning estimate x_0 equals 0
x0 = zeros(1,length(y));
x0=x0';
n = length(y);

% Tally results and define matrix (x) as
x = [x0,zeros(n,itr)];

% Iterative method runs a specified number of iterations that calculates
the values of x^k.
for k = 1:itr

for i = 1:n
```

```
%"sigma"  used to sum the value calculations for each equation)
sigma = 0;

for j = 1

sigma = sigma + x(:,k);

%x_i is calculated on the equation marked down its iteration
end

end
x(:,k+1) = ((inv(D)*y)+(inv(D)*(-R)*sigma));
end
l2_jac=sqrt((x_exact(1))^2+(x_exact(2))^2+(x_exact(3))^2
)-sqrt((x(1,5))^2+(x(2,5))^2+(x(3,5))^2 );
```

OUTPUT:

```
xy = 3x4

   -1.0000    0.2000    0.4000   -1.0000
    2.0000    6.0000    1.0000    1.0000
         0    6.0000   -2.0000    1.0000


xy = 3x4

   -1.0000    0.2000    0.4000   -1.0000
         0    6.4000    1.8000   -1.0000
         0    6.0000   -2.0000    1.0000


xy = 3x4

   -1.0000    0.2000    0.4000   -1.0000
         0    6.4000    1.8000   -1.0000
         0   23.6000         0   -0.2000


Xgauss = 3x1

    0.7881
   -0.0085
   -0.5254
```