1. $f'(x_j) = a_0 f(x_j) + a_1 f(x_{j+1}) + a_2 f(x_{j+2}) + a_3 f(x_{j+3})$

Taylor Expansion is given by $\sum_{0}^{\infty} \frac{(x-a)^n}{n!} f^{(n)}(a)$
at Point a

$$f(x_{j+1}) = f_j + hf_j' + \frac{h^2}{2} f_j'' + \frac{h^3}{6} f_j''' + \frac{h^4}{24} f_j^4 + \frac{h^5}{120} f_j^5$$

$$f(x_{j+2}) = f_j + 2hf_j' + \frac{4h^2}{2} f_j'' + \frac{8h^3}{6} f_j''' + \frac{16h^4}{24} f_j^4 + \frac{32h^5}{120} f_j^5$$

$$f(x_{j+3}) = f_j + 3hf_j' + \frac{9h^2}{2} f_j'' + \frac{27h^3}{6} f_j''' + \frac{81h^4}{24} f_j^4 + \frac{243h^5}{120} f_j^5$$

Taylor Table

	f_j	f_j'	f_j''	f_j^3	f_j^4	f_j^5
f_j'	0	-1	0	0	0	0
$a_0 f_j$	a_0	0	0	0	0	0
$a_1 f_{j+1}$	a_1	$a_1 h$	$a_1 h^2/2$	$a_1 h^3/6$	$a_1 h^4/24$	$a_1 h^5/120$
$a_2 f_{j+2}$	a_2	$a_2 2h$	$a_2 2h^2$	$a_2 4/3 h^3$	$a_2 2/3 h^4$	$a_2 4/15 h^5$
$a_3 f_{j+3}$	a_3	$a_3 3h$	$a_3 9/2 h^2$	$a_3 9/2 h^3$	$a_3 27/8 h^4$	$a_3 81/40 h^5$

We need a system of 4 by 4 equations which is highlighted in the table and has solutions

$a_0 = 11/6h$

$a_1 = -3/h$

$a_2 = 3/2h$

$a_3 = -1/3h$

Problem 1

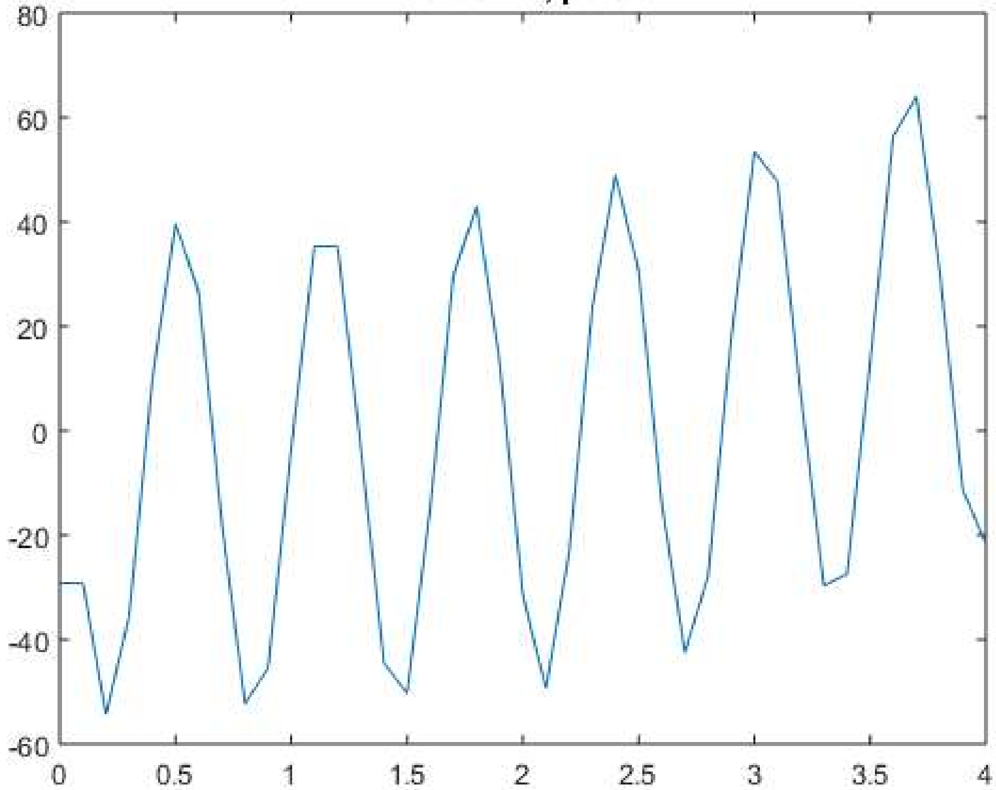
```
Taylor_Table = [
    1, 1, 1, 1
    0 h 2*h 3*h
    0 1/2*h^2 2*h^2 9/2*h^2
    0 1/6*h^3 4/3*h^3 9/2*h^3
    ];

Sol = [0, -1, 0, 0];

linsolve(Taylor_Table, Sol')
```

```
ans =

 11/(6*h)
     -3/h
  3/(2*h)
 -1/(3*h)
```

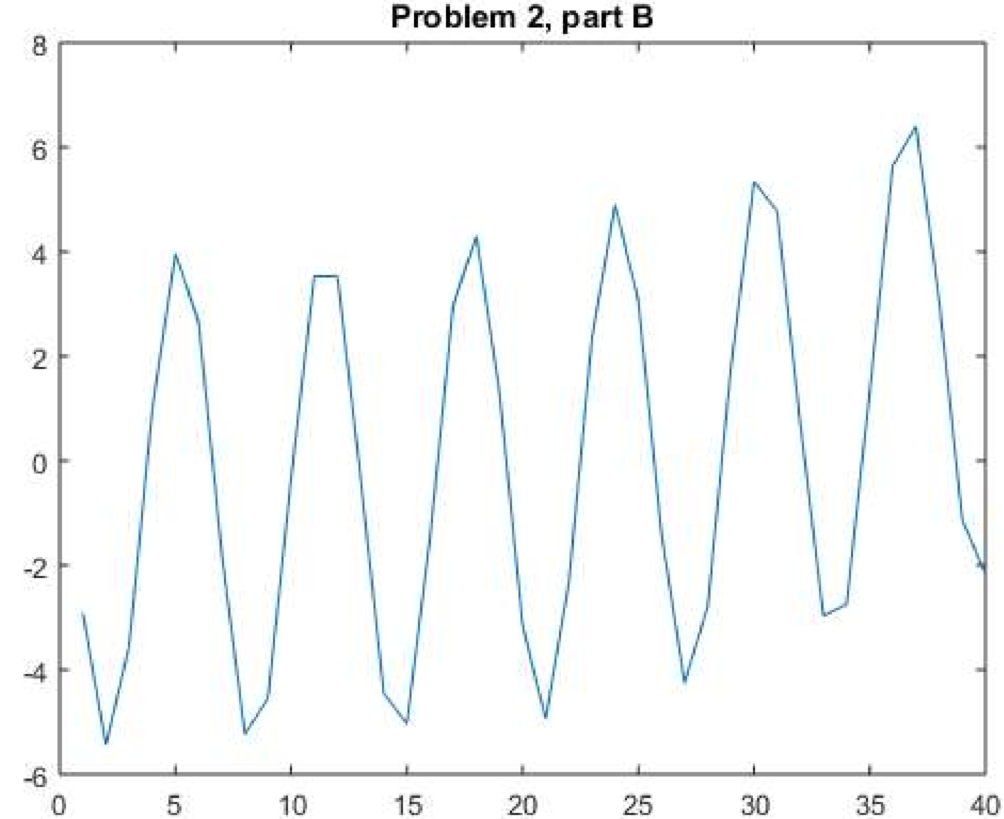
Problem 2

```
X = [];
Y = [];
for j = 0.0:0.1:4.0
    % Fill in with descrete values
    X = [X, j];
    Y = [Y, foo(j)];
end

Yp = [fow_diff(X, 1)];
for j = 2:numel(X)
    Yp = [Yp, back_diff(X, j)];
end

plot(X, Yp)
title('Problem 2, part A')
figure
plot(diff(Y))
title('Problem 2, part B')

function [ y ] = back_diff ( X, j )
    y = ( foo(X(j)) - foo(X(j-1)) ) / 0.1;
end

function [ y ] = fow_diff ( X, j )
    y = ( foo(X(j+1)) - foo(X(j)) ) / 0.1;
end

function [ y ] = foo ( x )
    y = 5*cos(10*x) + x^3 - 2*x^2 - 6*x + 10;
end
```

Problem 2, part A

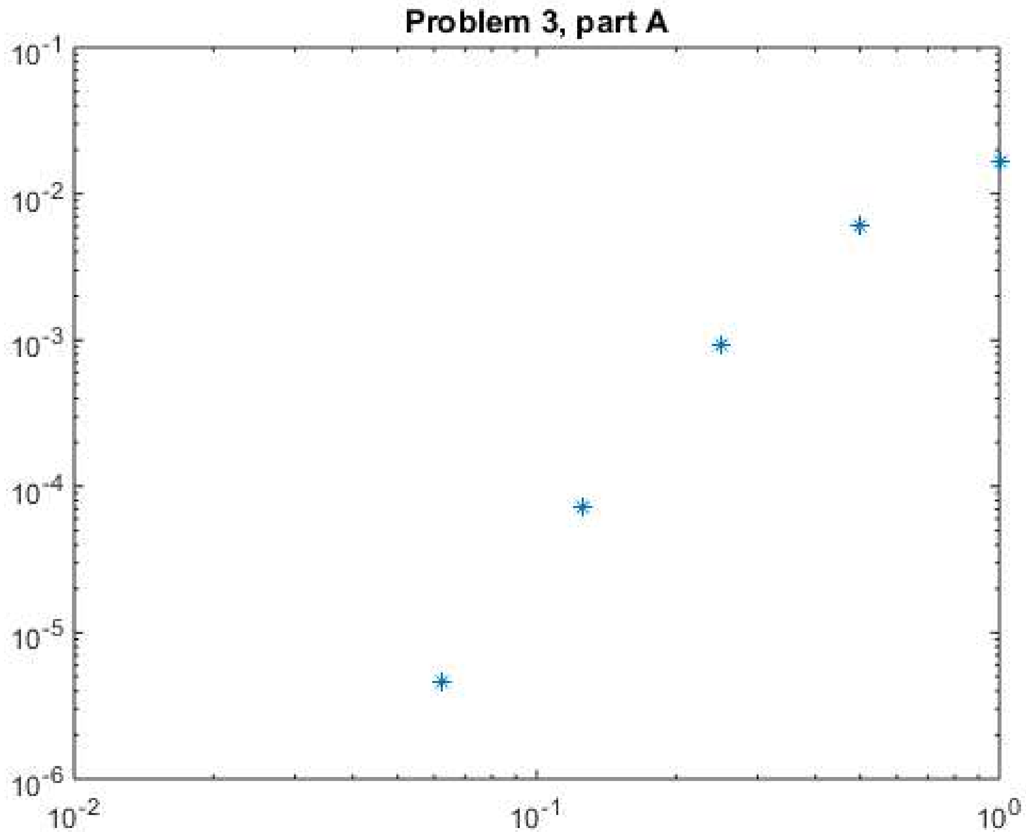


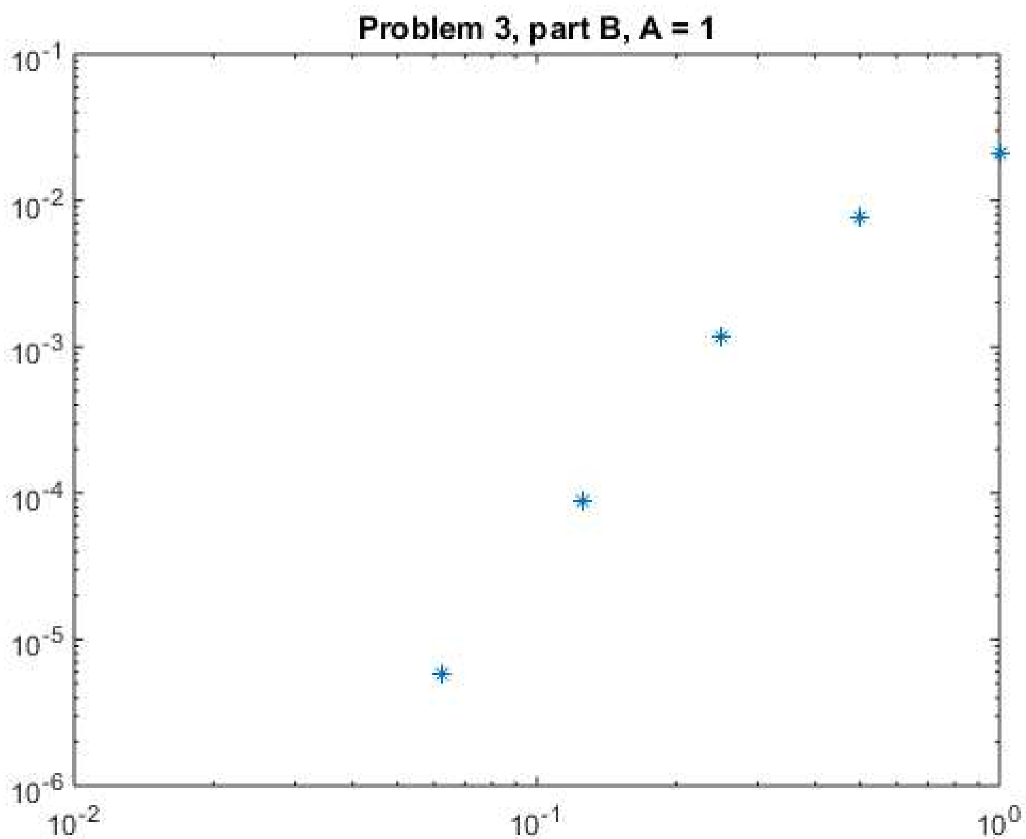
Problem 2, part B

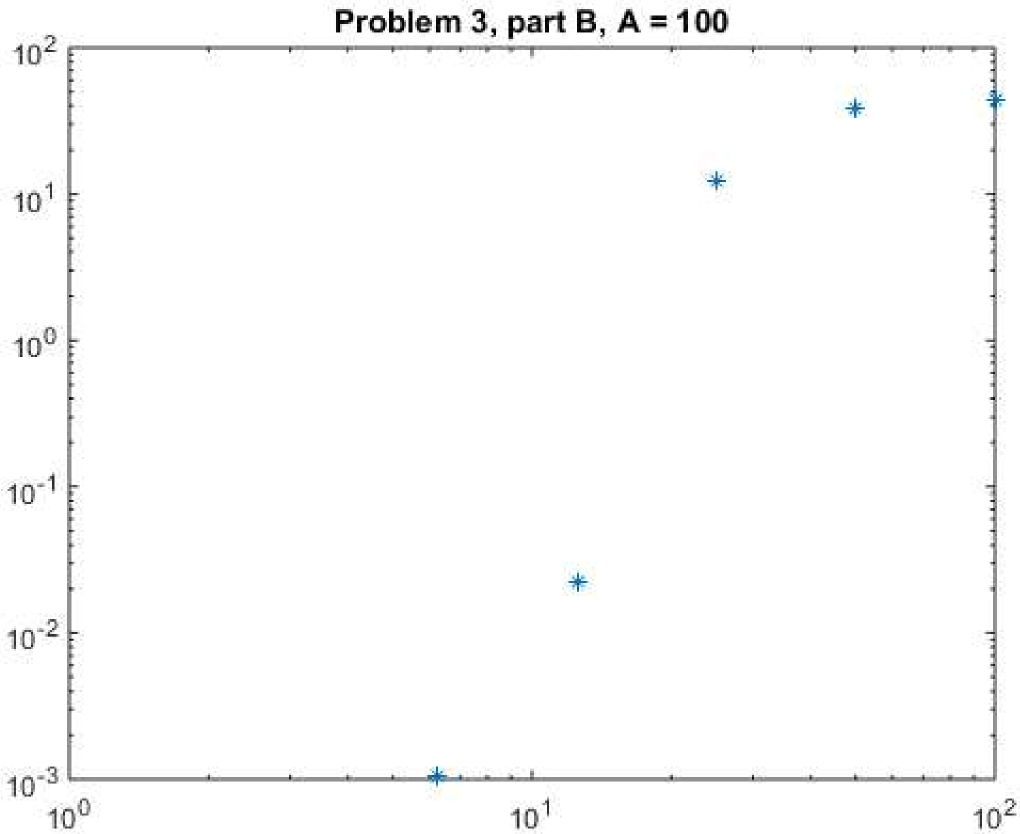


Problem 3

```
% Part a

Int = [1, 0.5, 0.25, 0.125, 0.0625];
Err = [];
x = 0.5;
for i = 1:numel(Int)
    Err = [Err, abs(1/(1 + x^2) - del(x, Int(i)))];
end

loglog(Int, Err, '*')
title('Problem 3, part A')
figure

% Part b

Int = [1, 0.5, 0.25, 0.125, 0.0625];
Err = [];
x = 0.5;
for i = 1:numel(Int)
    exact = 1/(1 + x^2);
    numer = del(x, Int(i));
    Err = [Err, abs( (exact - numer) / exact )];
end

loglog(Int, Err, '*')
title('Problem 3, part B, A = 1')
figure

A = 100;
Int = E([1, 0.5, 0.25, 0.125, 0.0625], A);
Err = [];
x = E(0.5, A);
for i = 1:numel(Int)
    exact = 1/(1 + x^2);
    numer = del(x, Int(i));
    Err = [Err, abs( (exact - numer) / exact )];
end

loglog(Int, Err, '*')
title('Problem 3, part B, A = 100')

function [ y ] = del ( j, h )
    y = ( f(j-2*h) - 8*f(j-h) + 8*f(j+h) - f(j+2*h) ) / (12*h);
end

function [ y ] = E ( x, A )
    y = A*x;
end

function [ y ] = f ( x )
    y = atan(x);
end
```


Problem 3, part A
10^-1
10^-2
10^-3
10^-4
10^-5
10^-6
10^-2
10^-1
10^0


Problem 3, part B, A = 1
10^-1
10^-2
10^-3
10^-4
10^-5
10^-6
10^-2
10^-1
10^0



Problem 3, part B, A = 100
10^2
10^1
10^0
10^-1
10^-2
10^-3
10^0
10^1
10^2

3.b How does the error term change when the x coordinate is scaled with the transformation $\xi(x) = Ax$

We have $f_j' = \dfrac{f_{j-2} - 8f_{j-1} + 8f_{j+1} - f_{j+2}}{12h} + O(h^4)$

meaning that the error ε of this formula behaves quadratical with respect to h : $\varepsilon = kh^4$

If we apply the transformation we get $x_\xi = Ax$
$h_\xi = Ah$

$$\frac{\partial f(Ax_j)}{\partial Ax_j} = \frac{f(Ax_{j-2}) - 8f(Ax_{j-1}) + 8f(Ax_{j+1}) - f(Ax_{j+2})}{12(Ah)} + O[(Ah)^4]$$

in this new formula, the error also behaves quadratically with h, however the proportionality factor (previously unknown) is also scaled. as $A^4k = k'$: $\varepsilon' = k'h^4$

$\therefore$ The accuracy is not dependent on the choice of indep variable. we should spect the same behaviour, and a slope of 4 in the log log scale

4. b. $\frac{\delta(u_j v_j)}{\delta x} = \frac{u_{j+1} v_{j+1} - u_{j-1} v_{j-1}}{2h} =$ using central finite diff

$$= \left[2u_{j+1} v_{j+1} - 2u_{j-1} v_{j-1} \right] / 4h =$$

$$= \left[\begin{array}{l} 2u_{j+1} v_{j+1} + u_{j-1} v_{j+1} - u_{j+1} v_{j-1} - 2u_{j-1} v_{j-1} \\ \quad - u_{j-1} v_{j+1} + u_{j+1} v_{j-1} \end{array} \right] / 4h$$

$$= \left[\begin{array}{l} \overset{a}{\cancel{u_{j+1} v_{j+1}}} + \overset{b}{\cancel{u_{j-1} v_{j+1}}} - \overset{a}{\cancel{u_{j+1} v_{j-1}}} - \overset{b}{\cancel{u_{j-1} v_{j-1}}} \\ \overset{a'}{\cancel{u_{j+1} v_{j+1}}} - \overset{a''}{\cancel{u_{j-1} v_{j+1}}} + \overset{b'}{\cancel{u_{j+1} v_{j-1}}} - \overset{b'}{\cancel{u_{j-1} v_{j-1}}} \end{array} \right] / 4h$$

$$= \left[\begin{array}{l} u_{j+1} (v_{j+1} - v_{j-1})^{(a)} + v_{j+1} (u_{j+1} - u_{j-1})^{(a')} \\ u_{j-1} (u_{j+1} - v_{j-1})^{(b)} + v_{j-1} (u_{j+1} - u_{j-1})^{(b')} \end{array} \right] / 4h$$

$$= \frac{(u_{j+1} + u_{j-1})(v_{j+1} - v_{j-1})}{4h} + \frac{(v_{j+1} + v_{j-1})(u_{j+1} - u_{j-1})}{4h}$$

$$= \frac{(u_{j+1} + u_{j-1})}{2} \frac{(v_{j+1} - v_{j-1})}{2h} + \frac{(v_{j+1} + v_{j-1})}{2} \frac{(u_{j+1} - u_{j-1})}{2h}$$

$$= \bar{u}_j \frac{\delta v_j}{\delta x} + \bar{v}_j \frac{\delta u_j}{x} \quad \square$$

a. We conclude from the above proof that the analogous finite differentiation does <u>not</u> hold in descrete form since

$$u_j \neq \bar{u}_j \wedge v_j \neq \bar{v}_j \text{ in general}$$

5. $x_0 = -1$, $x_1 = 1$, $x_2 = 2$, and the formulas for the Lagrange polynomials for 3 points are

$$L_0 = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} = \frac{(x-1)(x-2)}{(-2)(-3)} = \frac{x^2-3x+2}{6}$$

$$L_1 = \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} = \frac{(x+1)(x-2)}{(2)(-1)} = -\frac{x^2-x-2}{2}$$

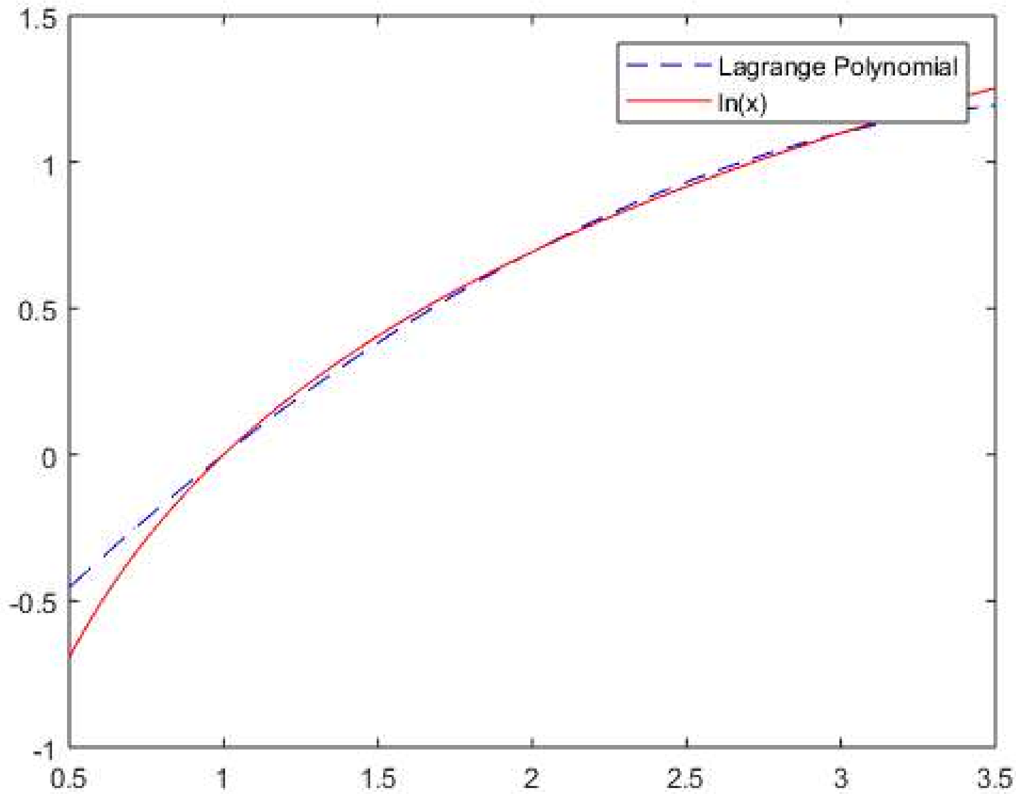
$$L_2 = \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} = \frac{(x+1)(x-1)}{(3)(1)} = \frac{x^2-1}{3}$$

Problem 5

```
% Part a

syms x x0 x1 x2

x0 = -1;
x1 = 1;
x2 = 2;

L = [
    (x - x1)*(x - x2) / ( (x0 - x1)*(x0 - x2) )
    (x - x0)*(x - x2) / ( (x1 - x0)*(x1 - x2) )
    (x - x0)*(x - x1) / ( (x2 - x0)*(x2 - x1) )
    ];

disp(L)

% Part b

x0 = 1;
x1 = 2;
x2 = 3;

L = [
    (x - x1)*(x - x2) / ( (x0 - x1)*(x0 - x2) )
    (x - x0)*(x - x2) / ( (x1 - x0)*(x1 - x2) )
    (x - x0)*(x - x1) / ( (x2 - x0)*(x2 - x1) )
    ];

X = [x0 x1 x2];
Y = log(X);
LagPol = Y*L;

disp( LagPol )

Test = [1.5 2.4];
for i = 1:numel(Test)
    fprintf('Testing %f \n', Test(i));
    fprintf('Estimate: %f Real: %f \n', vpa (subs( LagPol, x, Test(i))), log( Test(i)));
end

X = linspace(1/2, 7/2);
y1 = vpa (subs( LagPol, x, X));
y2 = log(X);

figure
plot(X,y1,'b--',X,y2,'r')
legend('Lagrange Polynomial','ln(x)')
```

```
  ((x - 1)*(x - 2))/6
 -((x + 1)*(x - 2))/2
  ((x - 1)*(x + 1))/3
```

```
(2473854946935173*(x - 1)*(x - 2))/4503599627370496 - (6243314768165359*(x - 1)*(x - 3))/9007
199254740992

Testing 1.500000
Estimate: 0.382534 Real: 0.405465
Testing 2.400000
Estimate: 0.889855 Real: 0.875469
```



Published with MATLAB® R2017a