MATH 131: Numerical Methods for scientists and engineers
Coding Exam 3–PRACTICE–Spring 2018

You worked with your classmate Galois on a coding project. Unfortunately Galois is not around, and he left you with an unfinished code. It's up to you to finish, find the bugs, and submit. You will have at hand the exercise statement and Galois's code.

1. Download the `coding3_practice.mlx` on Catcourses.
2. Put a comment for each bug and explain. Full credit will be awarded to both finding the bugs and explaining.
3. Good luck!

The Code Project you and Galois were doing is the following.

"Write a function called `Lagrange_interpolant` *that computes the Lagrange interpolant of a function. Document your function and your code. Use that function to interpolate* $cos(x^3)$ *over* `[0:0.01:2]'` *with at least* 10^{-6} *accuracy. Plot on the same graph the function and it's interpolant. Comment on your results.*
We recall that, the Lagrange interpolant of a function f *over* $x_0, \ldots, x_n$ *is given by*

$$P(x) = \sum_{k=0}^{n} f(x_k) L_{n,k}(x), \quad \textit{with} \quad L_{n,k}(x) = \frac{\prod_{j=0, j\neq k}^{n} (x - x_j)}{\prod_{j=0, j\neq k}^{n} (x_k - x_j).}$$

Notice: there are $n+1$ *points, therefore* n *subintervals."*
On the back there is a screen shot of Galois's code.

Because it's a practice exam you have more bugs to find that expected for the coding exam 3. Here there are no more than 10 small things to be adjusted, or completed.

Code Project Galois and ....

Put your name above

Exercise

```
f = @(x) x.*cos(x);
%  TODO: comment what x, N, datx, daty are.
N = 70;
x = [0:0.1:2]';
dx = 2/N;
datx = [0:dx:2];
daty = f(datx);
L = Lagrange_poly(x,daty,datx)

max(max(abs(f(x)-L)))

plot(x,,x,,'--k')
xlabel('x value')
legend('....', '.....')
```

Function

```
function y = Lagrange_interpolant(x,datx,daty)
    %We need to comment... I'll do that later

    N = length(datx);
    for i = 1:N
        L{i} = ones(length(x),1);
        for j = 1:N
            if i==j
                L{i} = L{i};
            else
                L{i} = L{i}.*(x + datx(j)) / (datx(i) - datx(j));
            end
        end
    end

    %creation of the interpolant
    y = zeros(length(x),1);
    for k = 1: N
        y = y + 2*daty(k)*L{k};
    end

end
```