MA 302 – Spring 2020
Homework 3: Due on Wednesday, February 19

Instructions:
The code submitted should be your own creation. You may consult MATLAB's documentation or the notes from class. The submission of codes obtained from online sources is a violation of Loyola's honor code.

Exercise 1: Relational expressions and Arrays

1. Let `x=1:10` and `y = [2 5 6 1 8 2 9 4 0 7]`. Execute and interpret the results of the following commands

   (a) `x > y`
   (b) `x & (~y)`
   (c) `y==2`
   (d) `y(logical(~rem(y,2)))`
   (e) `x(y<0)`
   (f) `y((x<2)|(x>=8))`

2. Let `x = [3 16 9 12 -1 0 -12 9 6 1]`. Provide command(s) that will

   (a) set the positive values of `x` to zero.
   (b) set the values that are multiples of 3 to 3.
   (c) set the values of `x` that are less than or equal to the mean to 0.
   (d) set the values of `x` that are above the mean to their difference from the mean.

3. Let `A = ceil(5*randn(6,6))`. Use logical indexing and the `find` command without any loops to write commands that will:

   (a) find the indices and list all the elements of `A` which are smaller than -3.
   (b) find the indices and list all elements of `A` that are smaller than 6 but larger than 1.
   (c) Remove those columns of `A` which contain at least one `0` element.

Exercise 2: Loops and Functions

1. Write a script to find the largest value of n such that the sum:

$$\sqrt{1^3} + \sqrt{2^3} + \ldots + \sqrt{n^3}$$

   is less than 1000.

2. Create an 20 by 10 array of random numbers using `rand`. Loop through the array, element by element, and set any value that is less than 0.5 to 0 and greater than or equal to 0.5 to 1. Check your script by comparing to calling MATLAB's built in functions.

3. Write a function `approx_pi = series_approx_pi(N)` that computes the value of π using the following series:

$$\frac{\pi^2 - 8}{16} = \sum_{n=1}^{\infty} \frac{1}{(2n-1)^2(2n+1)^2}$$

   You may begin by computing the sum using a for loop to get an idea of how the approximation works but your submission should be a vectorized computation of the sum. This is way more efficient. See `for_loop_demo.m` from class. To run your code, create a script `run_series_approx_pi` that will test the approximation of π for $N = 10, 100, 1000, 10000$, When you run your script, it should output the following table:

```
>> run_series_approx_pi
N       Error
10      4.574574e-05
100     5.226273e-08
1000    5.297141e-11
10000   4.352074e-14
```

4. Write a function `m = wmean(x,w)` that computes the weighted mean, given the the vector `x` and a vector of non-negative weights `w`, such that $\sum_{i=1}^{n} w_i > 0$. The mean should be computed as

$$\frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$$

   Use the MATLAB `error` function to display an error message and terminate execution in the case when:

   (a) `x` and `w` are not of the same length
   (b) at least one of the elements of `w` are negative
   (c) the sum of all weights is negative

   Test your code as follows

   (a) Pick a random vector `x` of length 10 and a random vector of weights, `w` on $[0, 1]$
   (b) Test your error routines with a vector of weights with at least one negative value and another whose sum is negative.

5. Write a function `[approx_root, num_its] = bisection(f,a,b,tol)` that implements the bisection method. You function should take as input 4 arguments with the last argument being optional, i.e, if the user does not provide the accuracy `tol` use a default of `1.0e-6` (use `varargin` to attain this). Your function should output the approximate root, `approx_root` and the number of iterations it took to attain the root, `num_its`. However, if the user calls the function with one argument, your function should return a vector consisting of the approximate root and the number of iterations (see how we used `nargout` in class). In addition, your function should display and error message and terminate if the user provides an interval without a root. Test your method to solve the following:

   (a) $x^x = 50$ on $[3, 4]$
   (b) $\ln(x) = \cos(x)$ on $[1, 2]$

   In each case first find an interval of size 1 that contains the root, sketch the graphs to determine this. You should also provide tests showing

   (a) Your error code handling error messages is correct.
   (b) Run the code with the different options for input and output.
   (c) Run your code with `tol = 1.0e-12` and the default, `tol = 1.0e-06`.

Submission of exercises

Place all your files (`m-files, summary.txt`) in a folder named `lastname_hwN` and zip the folder to create a file `lastname_hwN.zip`. Email your zip file `lastname_hwN.zip` to `pchidyagwai@loyola.edu` with subject `MA302_hwN`.