MATLAB Programming NYCU Fall 2024 LAB#01 2024/9/12

For this lab, the goal is just to get familiar with some basic operations covered in the class. There are a few short practices. Notes:

(1) Use **No Loop** in these exercises.

(2) Use only functions mentioned in the class so far, unless noted otherwise.

(3) Save your code in script files (`*.m`) for easy testing/debugging.

Practices:

1. Given a vector v of length 4 representing two fractal numbers, give the text output of its fractional sum:

Example: v: `3 7 5 2` Your output: `3/7+5/2=41/14`

Note: Use **`fprintf`** for this task.

2. Compute `1/1 + 1/2 + 1/3 + ... + 1/999 + 1/1000`.

3. Compute $1 + a^1/1! + a^2/2! + a^3/3! + ... + a^{100}/100!$. (Note: **`cumprod`** is useful here.) Compare the result with **`exp(a)`** for several values of `a`.

4. Make `n`x`n` diagonal matrices (`n` given in a variable) whose diagonal values are 1 to `n`. (Note: Do not use the **`diag`** function for this exercise. First determine the linear indices of the diagonal elements, and then assign `1:n` to them.) Example for `n`=5:

```
1 0 0 0 0
0 2 0 0 0
0 0 3 0 0
0 0 0 4 0
0 0 0 0 5
```

5. For a vector of numbers, output another vector such that:

The odd-indexed elements of the output vector are the numbers in the input;

Each even-indexed element of the output is the mean of its two neighbors.

Example: For input `[1 3 6 2 5]`, the output is `[1 2 3 4.5 6 4.2 3.5 5]`.

6. Implement the functionality of the function **`meshgrid`** yourself. First check the documentation of **`meshgrid`** and also call it from the command window to see the generated matrices from given input vectors. Note: Use **`repmat`** in your implementation.

(You do not need to write a function here; just write your own statements to generate the output matrices from the input vectors. Use only the form of **`meshgrid`** for 2-D grids, with two input vectors `x` and `y`, and two output matrices `X` and `Y`.)