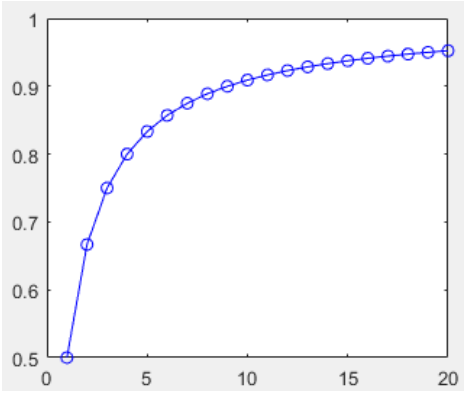
MATLAB Programming NCTU Fall 2017 Exam#1 2017/11/20

You need to write a m-file for each problem. Name your m-files **P1.m**, **P2.m**, etc.

You can consult the lecture slides, your own class notes, MATLAB documentation, and other printed material you can bring with you, but no other material is allowed. You are not allowed to talk with one another or use your own 3C devices or USB drives. You can use any function that we have covered in the class unless noted otherwise, or any standard MATLAB functions you can find. However, you cannot use toolbox functions that are not covered in the class.

1. [20%] Write a function **P1** that computes $f_n = \sum_{k=1}^{n} \frac{1}{k(k+1)}$.

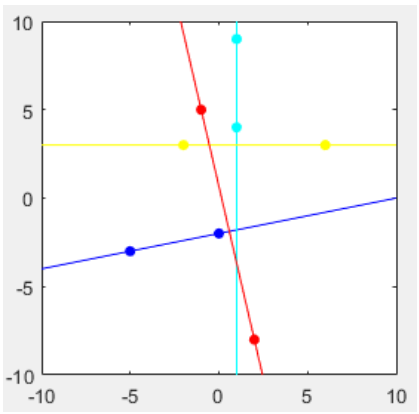
The function takes only one inputs `m`, which is a positive scalar integer. You need to compute f_n for all 1 <= n <= `m` and plot the result. An example output with `m`=20 is shown here. Note: The maximum points you get are reduced by 5 for each layer of loop used.



2. [15%] Write a function **P2** that takes an input `n`, a positive odd integer, and output a `n`x`n` matrix with the type of pattern shown to the right (example for `n`=7); the border elements are always zero and the maximum is at the center. Note: Use no loops.

```
0 0 0 0 0 0 0
0 1 1 1 1 1 0
0 1 2 2 2 1 0
0 1 2 3 2 1 0
0 1 2 2 2 1 0
0 1 1 1 1 1 0
0 0 0 0 0 0 0
```

3. [25%] Write a function **P3** that takes one input matrix `W`. Each row of `W` is a 4-element vector `[x1 y1 x2 y2]` that represents a pair 2-D points. For each point pair, plot a line that goes through the point pair, as well as the points themselves. (Set the range to `-10~10` for both x and y axes; all the coordinates given will be in this range.) Furthermore, draw each line and its pair of points in a different color. Decide for yourself how you choose the colors; a simple approach is to use a color map function. You can use a layer of loop.



Example plot for `P3([-5 -3 0 -2; 1 4 1 9; -2 3 6 3; -1 5 2 -8])`:

4. [20%] Write a function **P4** that takes a structure array as input. This structure array has two fields: `K` and `N`, which all contain scalar integers. The function prints out one line for each unique `K`, and the sum of `N` from the elements in the input structure array with that `K`. You can use a loop here. Format your output as in the example below:

```
P5( struct('K', [1 3 5 2 2 3 1 3], 'N', [16 10 9 8 4 5 6 12]) );

K=1   sum(N)=22
K=2   sum(N)=12
K=3   sum(N)=27
K=5   sum(N)=9
```

5. [20%] Write a function **`P5`** that takes two inputs: The first is an image `im`, and the second is a 4-element vector `[x1 y1 x2 y2]` representing a rectangle. Your function returns `im` with the specified rectangular region replaced by its complement. Note: `im` may be a gray-scale or RGB image, and is of type double. Do nothing if `x2<x1` or `y2<y1`, or if the rectangular region is completely outside of the image. Your function should be able to handle cases when any of these coordinates are outside of the image.

An example is given here: (image size: 360x240; the second input is `[100 40 180 260]`)