For this homework assignment, you will design the electromagnetic cloak illustrated below using transformation optics (TO). Your cloak will be in the shape of a square and it will hide an equilateral triangular object centered inside of it as shown below. Your grid should be of physical size 1.0 in the both the x and y directions. Your square cloak should be of length w = 0.9 on each side. The triangle should have a side length of t = 0.4. You will <u>not</u> simulate the device in this homework.
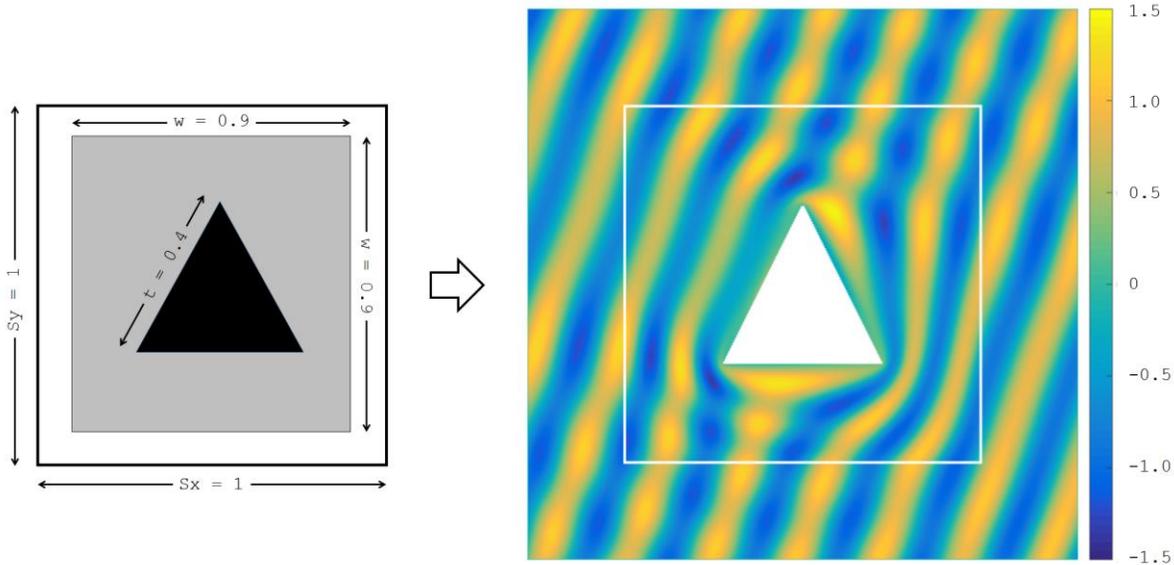
Use the following header for your MATLAB code.

```matlab
% Homework
%
% This MATLAB code designs an invisibility cloak using
% transformation electromagnetics.
%
% EE5322 21st Century Electromagnetics
% Spring 2016
%
% Instructor: Dr. Raymond C. Rumpf

% INITIALIZE MATLAB
close all;
clc;
clear all;

% OPEN FIGURE WINDOW
fig = figure('Color','w');


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% DASHBOARD
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% OBJECTS
w = 0.9;        %dimension of square cloak
t = 0.4;        %length of triangle side

% GRID
Sx = 1;
Sy = Sx;
Nx = 100;
Ny = round(Nx*Sy/Sx);
```
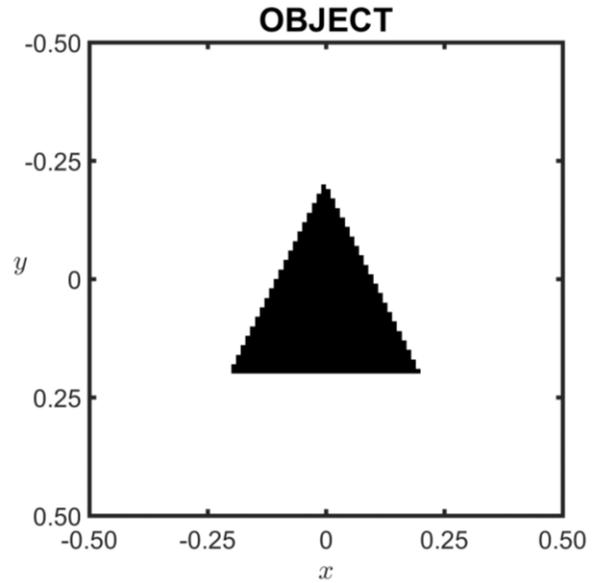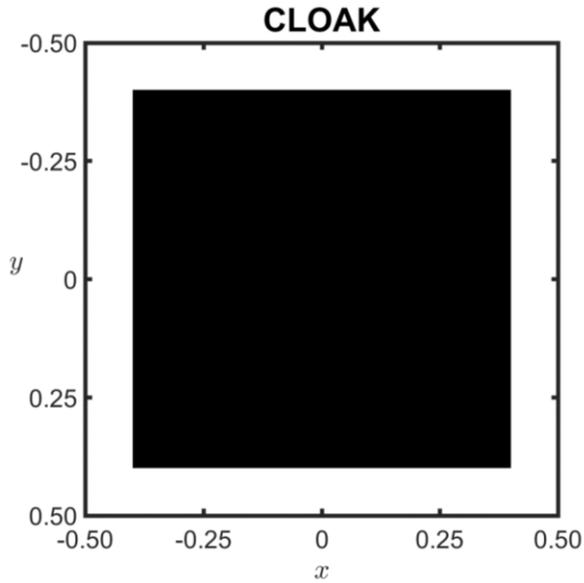
Problem #1: Build Cloak and Object Shapes

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% PROBLEM #1 -- BUILD CLOAK AND OBJECT
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Add a section to your code to build the square cloak `CLK` and triangle object `OBJ` on the grid. Center both objects. Visualize both your cloak and object shapes. They should look something like below.

Problem #2: Detect Edges

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% PROBLEM #2 - DETECT EDGES
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Add a section to your code to identify the edges of both of your cloak and object, ECLK and EOBJ. Your code must be generic and able to detect the edges of any shaped cloak and object without modification. As shown in the figure below, detect the points immediately outside of the cloak and immediately inside of the object. Notice that no edges are detected inside of the cloak region. In this manner, it will be possible to calculate values for μ and ε at each point in the cloak.
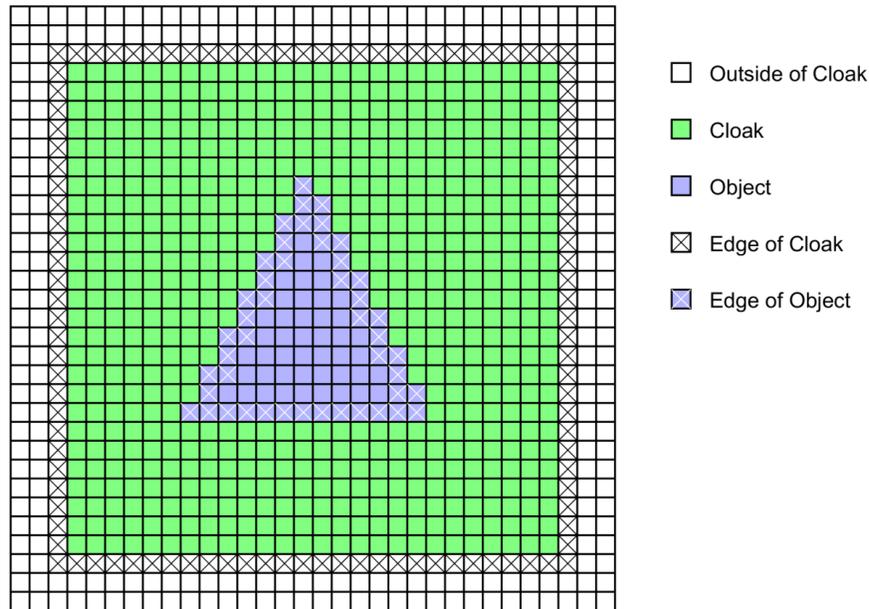

Visualize your data by plotting the cloak (and object) plus two times its edges. Your graphics should look something like below. It is possible to determine the proper placement of your edges by observing the maximum value of your data.
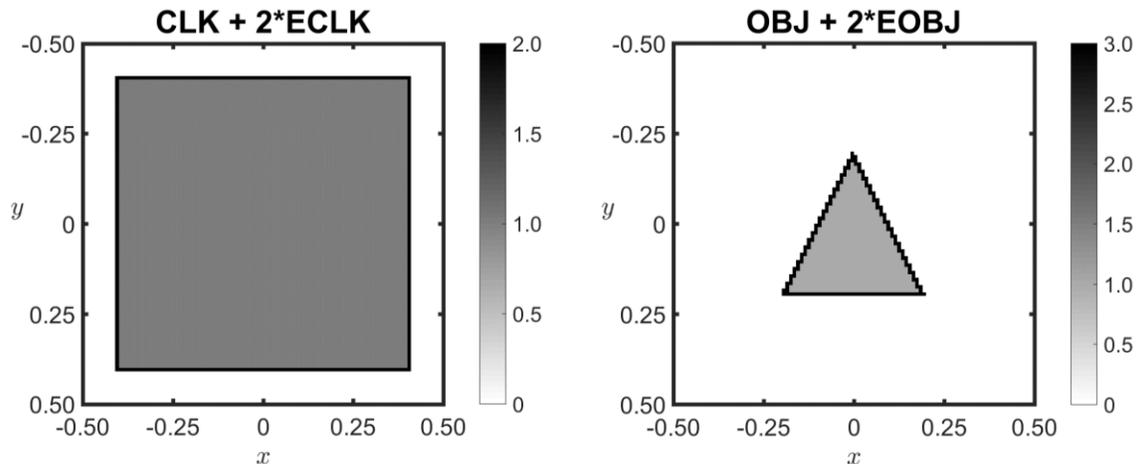
Problem #3: Generate Boundary Conditions for Spatial Transform

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% PROBLEM #3 -- GENERATE BOUNDARY CONDITIONS FOR SPATIAL TRANSFORM
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Add a section to your code to generate the boundary conditions for your spatial transforms. This will include: (1) a force map `F` which identifies the edges of your cloak and object, and (2) the coordinate values `XF` and `YF` that will be forced at theses edges. Visualize: (1) your force map `F`, (2) the original coordinates `XA` and `YA`, and (3) the coordinate values to be forced `XF` and `YF`. Your arrays should look something like below.
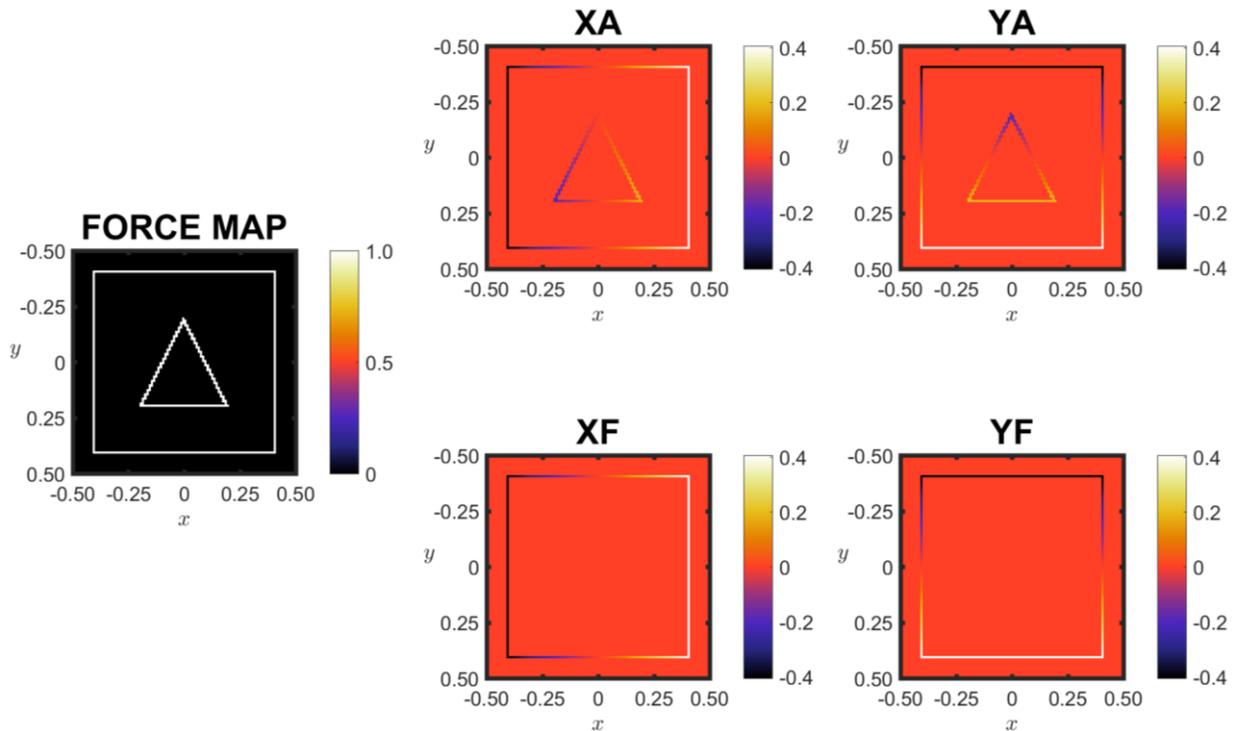
Problem #4: Calculate Coordinates by Solving Laplace's Equation

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% PROBLEM #4 -- CALCULATE COORDINATES BY SOLVING LAPLACE'S EQUATION
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Add a section to your code which solves Laplace's equation twice to calculate the intermediate coordinates XB and YB inside the cloak and outside of the object. Force all coordinate values outside of the cloak and inside of the object to zero.

Visualize XB and YB. Your two solutions should look something like below.
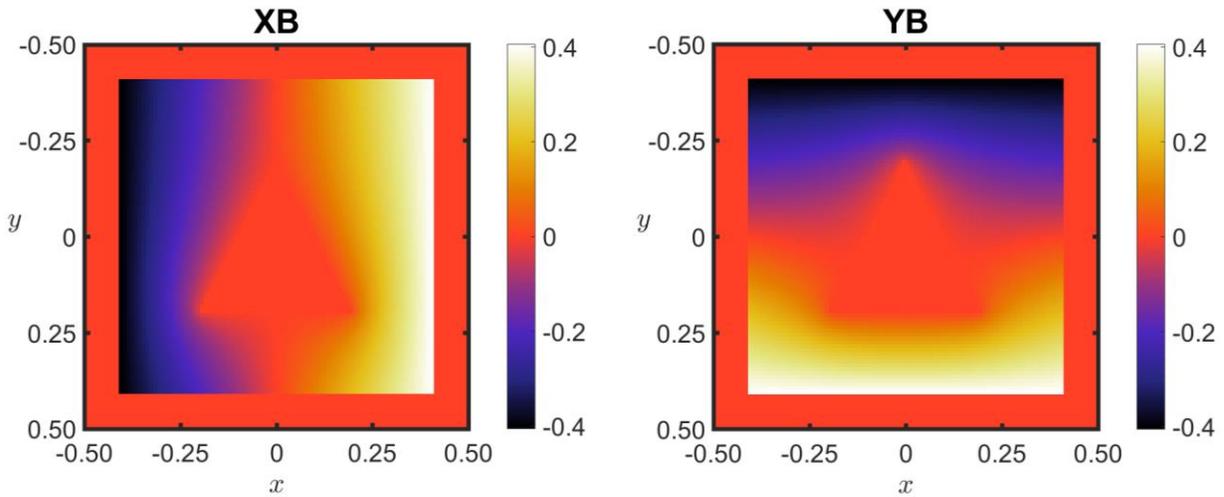
Problem #5: Calculate the Permittivity Tensor

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% PROBLEM #5 -- CALCULATE PERMITTIVITY TENSOR
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Add a section to your code to calculate the permittivity tensor from the spatial transform calculated above. Your data should look something like that shown below. Your peak values of permittivity may exceed what is shown. The color axis in this figure was limited to more clearly convey the smaller values on the grid.