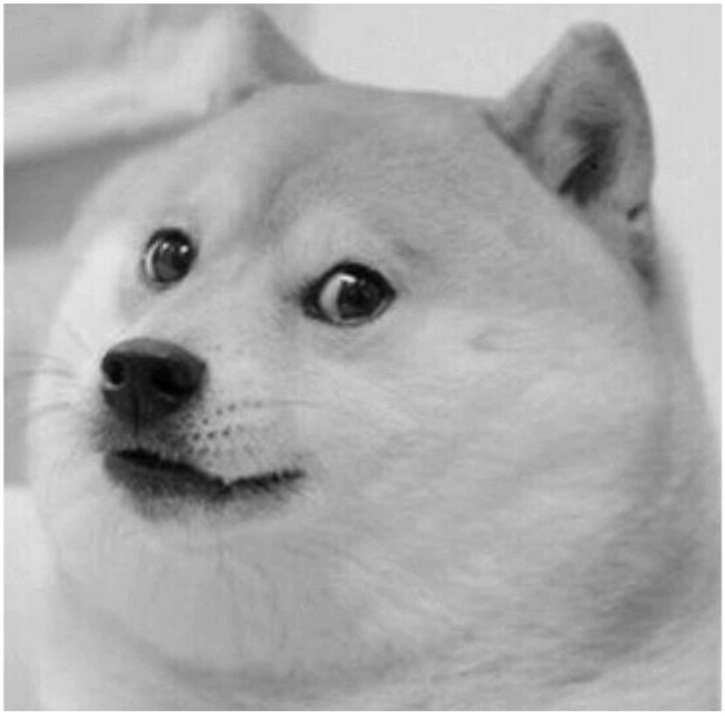
Recitation Assignment #9

Please do the following problems during your recitation session, including any additional problems given to you by your TA. Within 10 days of your recitation session, you must upload the complete solutions to these problems to Sakai, so that your TAs can evaluate them in a timely fashion. **Please submit any .m files to Sakai. Your .m files should be named in the following format: NETIDRecitation9Problem#, where NETID is your NetID, and # is the problem number. For instance if your NetID was aaa111, and you were answering problem 3, your .m file would be named aaa111Recitation9Problem3.m. All of your .m files should be in the form of a function. If you do not meet these naming requirements, and do not save as a function, you will not receive credit for your submission.** Collaborative problems can be worked on in teams of up to 5 people, as long as each team member individually completes the problem uploads the solution as part of their own Sakai submission or shows the solution to their instructor individually, and lists the names of all collaborators in the Sakai submission. Collaboration and discussion of solutions is not permitted for questions labeled as individual problems.

1. [Collaboration] Write a program that will take the negative of a black and white image. That is, it will turn dark places in the image into light places and light places into dark places. For example, it will turn this:



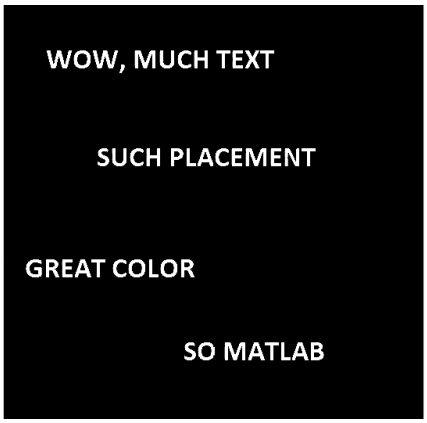
Into this:

2. [Collaboration] Write a program that calls your program from problem 1 to take the negative of a color image. That is, it turns this:

Into this:

3. [Collaboration] Write a program that takes 5 inputs, in the following order
    1. A string that is the file location of an image, as in ‘dogeBlank.jpg’
    2. A string that is the file location of a .csv file, as in ‘hiddenText.csv’
    3. A string that is either ‘red’, ‘green’, or ‘blue’
    4. A Boolean for taking the negative of the image
    5. A Boolean for taking the negative of the hidden message

    Two images and two hidden messages are provided for you on Sakai. The first image and its corresponding hidden message look like this:



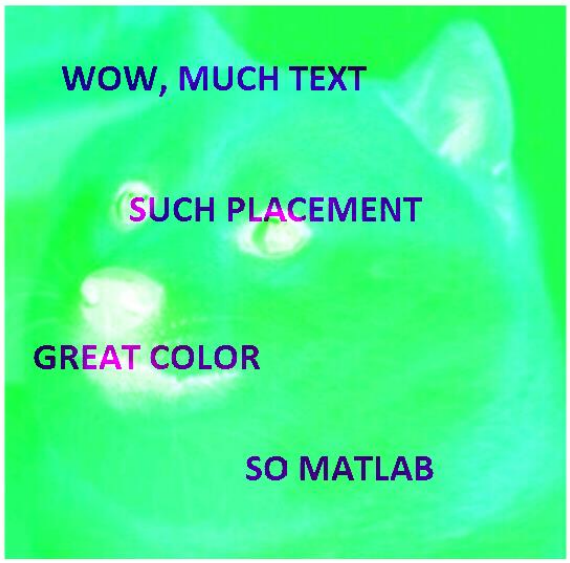
Your program should do the following in a single function:

a. Read in the image file and store the image in a variable.
b. Read in the csv file and store the resulting matrix in a variable.
c. Convert the matrix from the csv file into a uint8.
d. If the Boolean for taking the negative of the image is true, call the function from the second problem to take the negative of it, using this negative in place of the image from now on.
e. If the Boolean for taking the negative of the message is true, call the function from the first problem to take the negative of it, using this negative in place of the hidden message from now on.
f. Replace the given color in the image with the hidden message.
g. Return the resulting combination as an output.

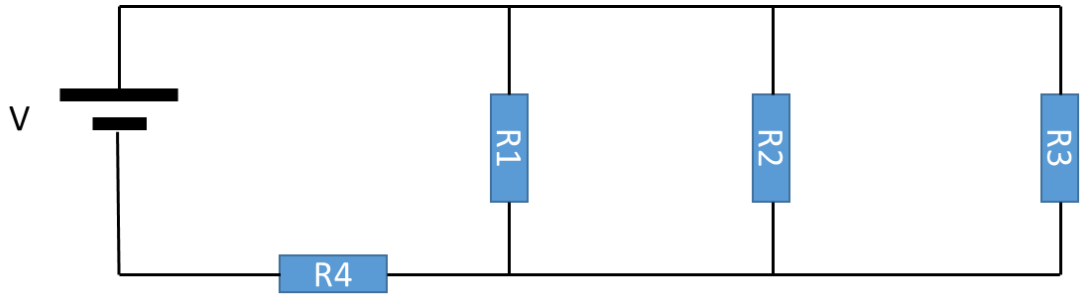
For the first image, the resulting output for the function call:

```
img = rec9prob3('dogeBlank.jpeg', 'hiddenDogeText.csv', 'green', 1, 1);
```

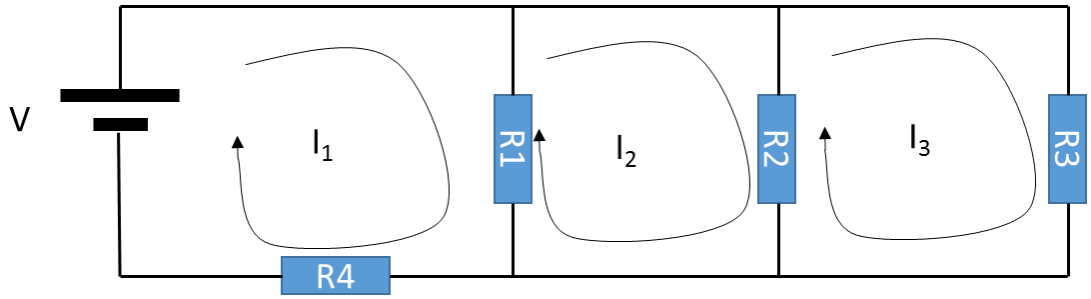
Should look like this:



4. [Collaboration] For the following electric circuit:



Where each blue square is a resistor, and the two parallel lines on the left represent a DC voltage source, we can perform a Kirchhoff's voltage analysis to find the voltage across each resistor using loop currents, like this:



To set up the following system of equations:

$$(R1 + R4)I_1 - R1I_2 = V$$
$$-R1I_1 + (R1 + R2)I_2 - R2I_3 = 0$$
$$(R2 + R3)I_3 - R2I_2 = 0$$

Which can be used to find the voltage drop across each resistor:

$$V_1 = (I_1 - I_2)R1$$
$$V_2 = (I_2 - I_3)R2$$
$$V_3 = I_3R3$$
$$V_4 = I_1R4$$

Write a function that has the following inputs in order:
- The voltage of the voltage source V, in volts.
- The value of R1 on ohms.
- The value of R2 in ohms.
- The value of R3 in ohms.
- The value of R4 in ohms.

This function should:

a. Create a matrix, A, for solving the system of equations above.

b. Create a vertical array, b, for solving the system of equations above.
c. Solve for a vertical matrix containing I_1, I_2, and I_3.
d. Use those values to calculate the voltage drop across each resistor, returning those drops as outputs in numbered order.

For example, the following function call:

```
[v1,v2,v3,v4] = rec9prob1(5,55,33,100,200);
```

Will result in:

v1 = 0.3938V

v2 = 0.3938V

v3 = 0.3938V

v4 = 4.6062V