Project 0 Questions

Instructions

- Compile and read through the included MATLAB tutorial.
- 3 questions.
- Include code.
- Feel free to include images or equations.
- Please make this document anonymous.
- **Please use only the space provided and keep the page breaks.** Please do not make new pages, nor remove pages. The document is a template to help grading.
- If you really need extra space, please use new pages at the end of the document and refer us to it in your answers.

Questions

Q1: Please find and read the course collaboration policy on the course website, and write a paraphrased version.

A1: Your answer here.

Q2: We wish to set all pixels that have a brightness of 10 or less to 0, to remove sensor noise. However, our code is slow when run on a database with 1000 grayscale images.

Image: grizzlypeakg.png

```
A = imread('grizzlypeakg.png');
[m1,n1] = size( A );
for i=1:m1
    for j=1:n1
        if A(i,j) <= 10
            A(i,j) = 0;
        end
    end
end
```

Q2.1: How could we speed it up?

A2.1: Your answer here.

Q2.2: What factor speedup would we receive over 1000 images? Please measure it.

Ignore file loading; assume all images are equal resolution; don't assume that the time taken for one image $\times 1000$ will equal 1000 image computations, as single short tasks on multitasking computers often take variable time.

A2.2: Your answer here.

Q2.3: How might a speeded-up version change for color images? Please measure it.

Image: grizzlypeak.jpg

A2.3: Your answer here.

Q3: We wish to reduce the brightness of an image but, when trying to visualize the result, all we sees is white with some weird "corruption" of color patches.

Image: gigi.jpg

```
I = double( imread('gigi.jpg') );
I = I - 20;
imshow( I );
```

Q3.1: What is incorrect with this approach? How can it be fixed while maintaining the same amount of brightness reduction?

A3.1: Your answer here.

Q3.2: Where did the original corruption come from? Which specific values in the original image did it represent?

A3.2: Your answer here.