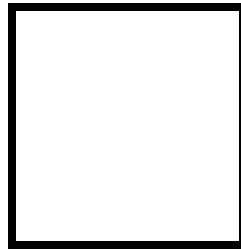




Republic of the Philippines  
**CAVITE STATE UNIVERSITY**  
Don Severino delas Alas Campus  
Indang, Cavite

## ELECTIVE 3

Midterm Exam  
**Image Processing in Octave**



Score

*Submitted by:*  
**Delima, Gerald C.**  
**CPEN111 / 4-2**

*Date Submitted*  
**24-11-2022**

*Submitted to:*  
**Engr. Maria Rizette H. Sayo**

## Methodology

### I. Importing, Displaying, and Converting Images

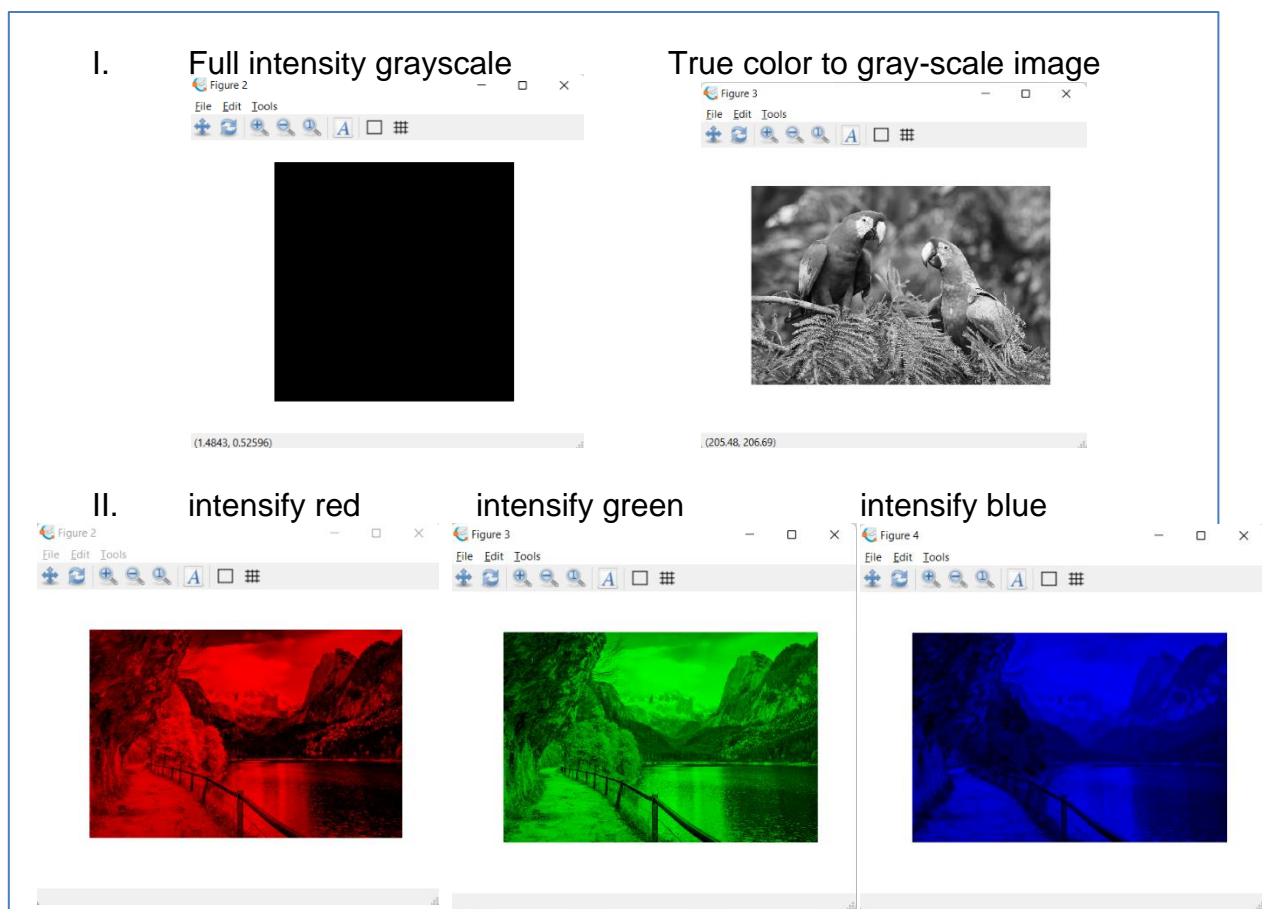
- Using the Editor of Octave, create a program that will load and display parrots.jpg
- Examine the size of the parrots by typing whos to find out the size of the image that you have read in
- Convert the class uint8 color image parrots to a gray scale image, and display the full intensity range gray-scale image using the imshow command
- Convert the true color image to a gray-scale image
- Save the program to this format parrots.m

### II. Display of Color Images

- Open the image file nature.jpg from the source folder
- Read in the file nature.jpg and display it on the screen as a reference image  
How large an image is created when we use the RGB representation compared to a gray-scale image conversion of it?
- Assign an image color that intensifies red, green, and blue and display each image in one window
- Convert each image file extension to png
- Save the program as nature.jpg

*Note: Take a screen shot of the output display of the converted images and do not forget to write your Octave code to this manuscript*

### Screen shot of Output Image



## Source Codes:

```
I
1 %Delima, Gerald C.
2 %BSCpE 4-2
3 %CPEN111
4 %Midterm Examination
5 %I. Importing, Displaying, and Converting Images
6 clc;
7 clf;
8 clear all;
9 close all;
10 parrot = imread('C:\Users\Gerald\Downloads\parrots.jpg');
11 figure(1),imshow(parrot);
12
13 whos parrot
14
15 uint8_parrot = imagesc(parrot, [0,255]);colormap(gray);
16 figure(2),imshow(uint8_parrot);
17 whos uint8_parrot
18 gray_parrot = rgb2gray(parrot);
19 figure(3),imshow(gray_parrot);

II
1 %Delima, Gerald C.
2 %BSCpE 4-2
3 %CPEN111
4 %Midterm Examination
5 %II. Display of Color Images
6 clc;
7 clf;
8 clear all;
9 close all;
10 nature = imread('C:\Users\Gerald\Downloads\nature.jpg');
11 figure(1),imshow(nature);
12
13 whos nature
14
15 gray_nature = rgb2gray(nature);
16
17 whos gray_nature
18
19 img_red = nature;
20 img_red(:,:,2)=0;
21 img_red(:,:,3)=0;
22 figure(2),imshow(img_red);
23
24 img_green = nature;
25 img_green(:,:,1) = 0;
26 img_green(:,:,3) = 0;
27 figure(3),imshow(img_green);
28
29 img_blue = nature;
30 img_blue(:,:,1) = 0;
31 img_blue(:,:,2) = 0;
32 figure(4),imshow(img_blue);
33
34 imwrite(img_red, 'C:\Users\Gerald\Downloads\red_nature.png');
35 imwrite(img_green, 'C:\Users\Gerald\Downloads\green_nature.png');
36 imwrite(img_blue, 'C:\Users\Gerald\Downloads\blue_nature.png');
```

Conclusion

The size of the image that is converted to gray scale is lower than the original image. We can adjust the intensity of grayscale to full intensity.

Github link.  
<https://github.com/Unreadabl/DelimaGerald-CPEN111.git>

Rubrics in Grading the Midterm Exam

	A – Excellent	B – Good	C – Fair	D – Needs Improvement
Specifications	The Program works and meets all of the specifications	The program works and produces the correct results and displays them correctly. It also meets the most of the other specifications	The program produces correct results but does not display them correctly	The program is producing incorrect results
Readability	The code is exceptionally well organized and very easy to follow	The code is fairly easy to read	The code is readable only by someone who knows what it is supposed to be doing	The code is poorly organized and very difficult to read.
Reusability	The code could be reused as a whole or each routine could be reused	Most of the code could be reused in other programs	Some parts of the code could be reused in other programs	The code is not organized for reusability
Documentation	The documentation is well written and clearly explains what the code is accomplishing and how	The documentation consists of embedded comment and some simple header documentation that is somewhat useful	The documentation is simply comments embedded in the code with some simple header comments separating routines	The documentation is simply comments embedded in the code and does not help the reader understand the codes
Efficiency	The code is extremely efficient without sacrificing readability and understanding.	The code is fairly efficient without sacrificing readability and understanding	The code is brute force and unnecessarily long	The code is huge and appears to be patched together
TOTAL				