**Assignment report** **Course** DIP

**Assignment number** 1

………………………………………………………………………………

Topics

1. Question1 explanation, output image
2. Question2 explanation, output image
3. Question3 explanation, output image
4. Question4 explanation, output image
5. Question5 explanation, output image

**Submitted by:** sohaib shahid ul haq

**Roll number:** 18i0728

**Submitted to:** Dr. Akhter Jamil

**Question 1:**

I read image using imread from matplotlib. then I defined 2 variables and copied height and weight of image to it. then I define a 2d matrix using this width and height with each element equal to the number of channels in the given image.

arr = [[[0 for col in range(4)]for row in range(width)] for x in range(height)]

d3 = np.array(arr,'f')

then I used 2 while loop to iterate through pixels of the image. and copied each pixel value from original image to it. but when colour value =orange opacity =replaced by 0.

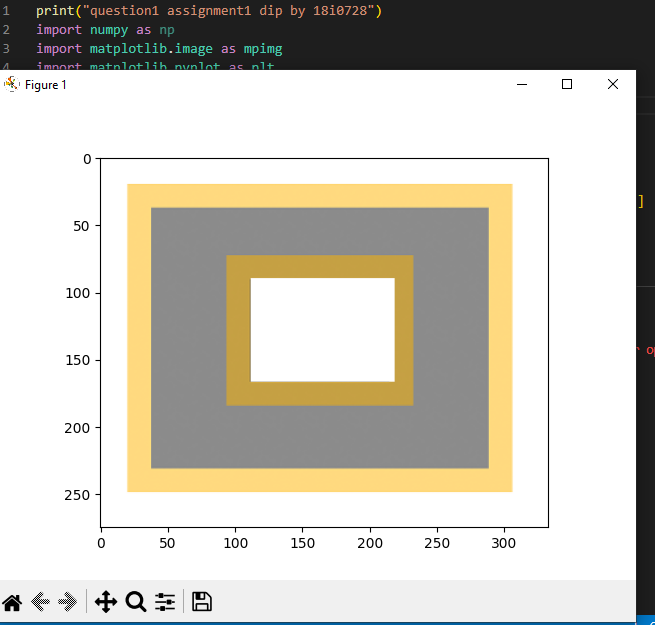
img[i][j][3]=0

making the 4th element value equal to 0. then displayed the image with

imgplot = plt.imshow(img)

plt.show()

**OUTPUT**



**Question 2**

I read image using imread from matplotlib. then I defined 2 variables and copied height and weight of image to it. then I define a 2d matrix using this width and height with each element equal to the number of channels in the given image.

img = mpimg.imread('test.png')

then I defined the function Convertogray

ConvertToGray(img\_para):

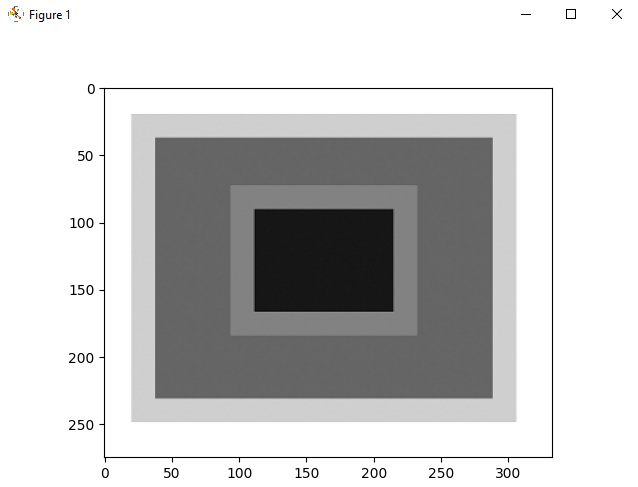
Then I extracted the red blue green color values and using the formula I defined the gray value.

R, G, B = img\_para[:,:,0], img\_para[:,:,1], img\_para[:,:,2]

    imgGray = 0.2989 \* R + 0.5870 \* G + 0.1140 \* B

then displayed the resulted image using plot.imshow().

**Output**

****

**Question 3**

I read image1 using imread from matplotlib. then I defined 2 variables and copied height1 and width1 of image to it.

Then 1 read image2 using imread from matplotlib. then I defined 2 variables and copied height2 and width2 of image to it.

    height\_img1, width\_img1 = img1\_para.shape[0:2]

    height\_img2, width\_img2 = img2\_para.shape[0:2]

Then I defined a matrix of h1\*w1+h2\*w2 size. then I define a 2d matrix using this width and height with each element equal to the number of channels in the given image.

 d3 = [[[0 for col in range(3)]for row in range(width\_img1\*2)] for x in range(height\_img1)]

Then I defined the condition that wither both images are compatible are not.

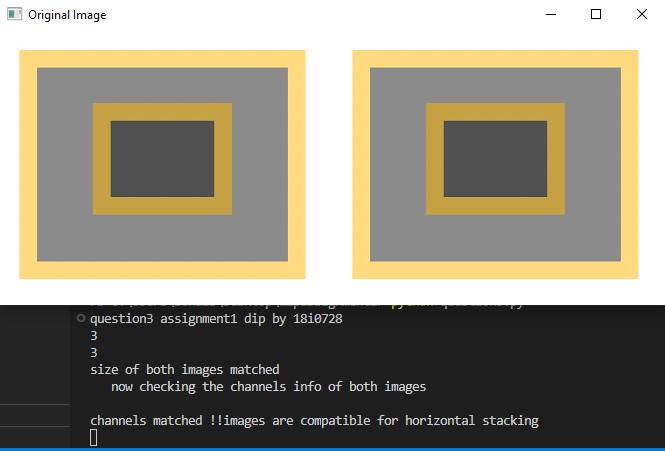
(height\_img1==height\_img2) and (width\_img1==width\_img2)

if(channels\_img1==channels\_img2):

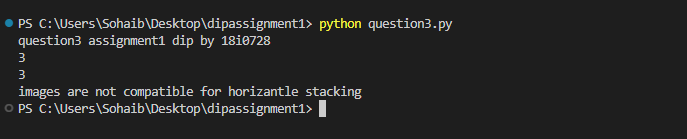
in case of compatibility I simply concatenated both using loop.

Then displayed the resulting matrix using implot.imshow()

**output**

**when compitable**

**when size did not match**



……………………………………………………………………………

**Question 4**

I read image using imread from matplotlib. then I defined 2 variables and copied height and weight of image to it. then I define a 2d matrix using this width and height with each element 4thelement equal to the number of channels in the given image

img = mpimg.imread('test2.jpg')

then I defined a function Flipimage(img\_para,flag=0)nwith 1 parameter and default parameter flag.

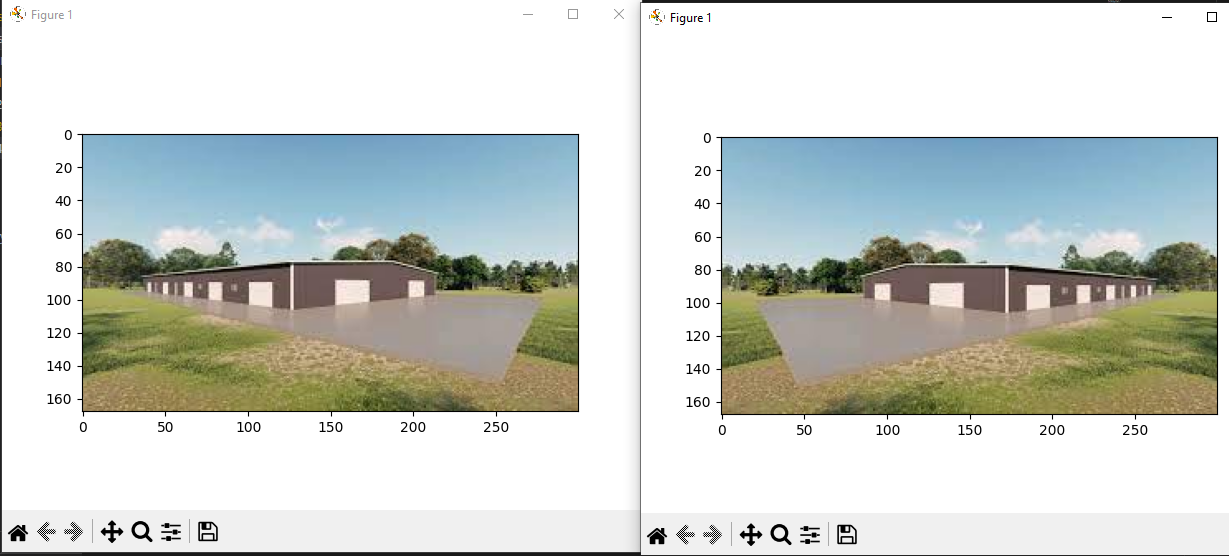
When the user send image to this function without sending the flag value in that case 0 is used so flip along y axis is done by using simple loop I copied element from image1 and placed it in the last end of the row.

 d3[i][iflp][k]=img\_para[i][j][k]

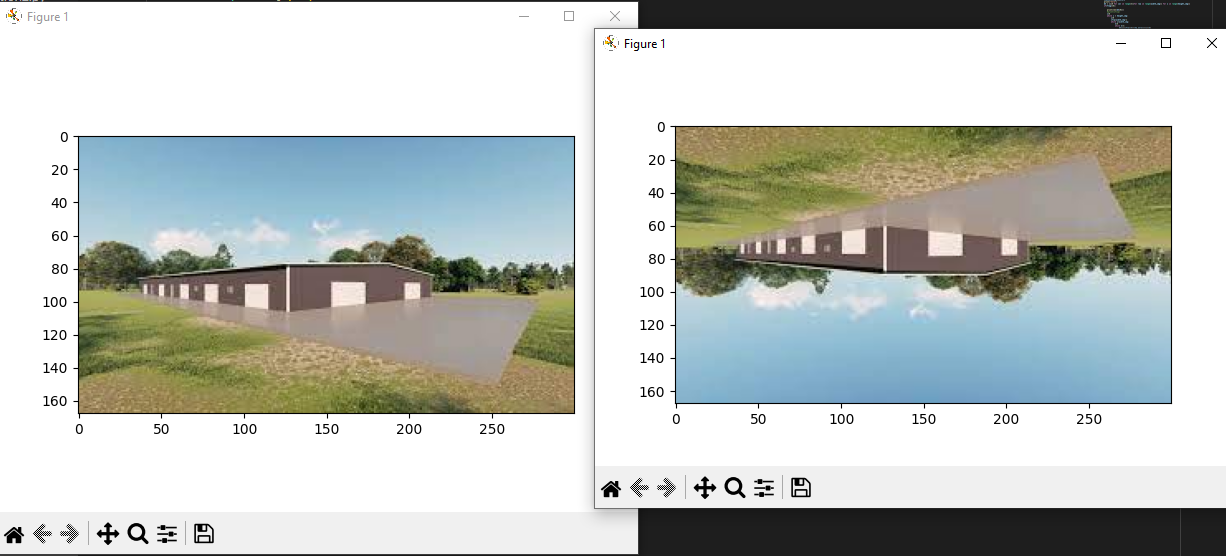
When the user send image to this function and also sends the flag value 1 in that case horizontal flip is used so flip along x axis is done by using simple loop I copied element from image1 and placed it in the last end of the column.

      d3[iflp][j][k]=img\_para[i][j][k]

**output**

**flag==0**

***flag=*01**



**Question 5**

I read image using imread from matplotlib. then I defined 2 variables and copied height1 and width1 of image1 to it. then I defined 2 more variables and copied height2 and width2 of image2 to it then I define a 2d matrix using this width and height with each element 4thelement equal to the number of channels in the given image

import numpy as np

import matplotlib.image as mpimg

import matplotlib.pyplot as plt

img1 = mpimg.imread('rect1.jpg')

img2 = mpimg.imread('rect1.jpg')

def CommonImg(img\_para1,img\_para2):

    height1, width1 = img\_para1.shape[0:2]

    height2, width2 = img\_para2.shape[0:2]

here the logic is that I take size of large size image among the two and define another image with it.

 if(height1\*width1 >= height2\*width2):

       d3 = [[[0 for col in range(3)]for row in range(width1)] for x in range(height1)]

then I used loop and copied only those only whose position and values are same in both images and assigned 0 to all remaining elements.

 if i<=height2 and j<=width2:

                      if (img\_para1[i][j][0]==img\_para2[i][j][0])and(img\_para1[i][j][1]==img\_para2[i][j][1])and(img\_para1[i][j][2]==img\_para2[i][j][2]):

                          d3[i][j][0]=img\_para1[i][j][0]

                          d3[i][j][1]=img\_para1[i][j][1]

                          d3[i][j][2]=img\_para1[i][j][2]

                      else:

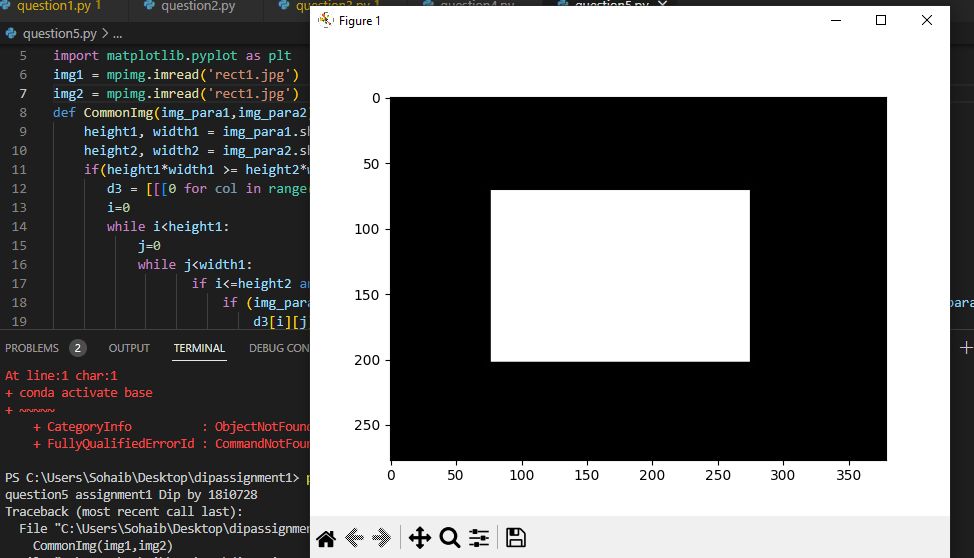
                          d3[i][j][0]=0

                          d3[i][j][1]=0

                          d3[i][j][1]=0

then displayed the resulted image using plot.imshow().

**Output**

**When both input images are rect1.jpg**

**When rect1.jpg and rect2.jpg is used as input images.**

