**Image Manipulation Assignment Report**

**Manohar Kowthavarapu (11554625)**

**Section-1:**

Below is the my\_flip1 method that takes image as an input and returns the mirror image.

def my\_flip1(lena\_image):

  print("Mirror Image")

  cv2\_imshow(lena\_image[:, ::-1, :])

Below is the my\_flip2 method that takes image as an input and returns the rotated image with grey in colour.

def my\_flip2(lena\_image):

    radius = math.radians(270)

    image = np.uint8(np.zeros(lena\_image.shape))

    image\_height = image.shape[0]

    image\_width  = image.shape[1]

    image\_midx, image\_midy = (image\_width//2, image\_height//2)

    for i in range(image\_height):

        for j in range(image\_width):

            x = round((i - image\_midx) \* math.cos(radius) \

            + (j - image\_midy) \* math.sin(radius)) + image\_midx

            y = round(-(i- image\_midx) \* math.sin(radius) \

            + (j - image\_midy) \* math.cos(radius)) + image\_midy

            if ( x >= 0 and y >= 0 and x < image\_height and y < image\_width):

                image[i,j,:] = lena\_image[x,y,:]

    print("Rotated Image with Grey in color")

    cv2\_imshow(image[:, :, 1])

Below is the upper\_quardant method that takes image as an input and return the 1/4th of the right upper quadrant image.

def upper\_quadrant(lena\_image):

  image\_height, image\_width = lena\_image.shape[:2]

  image\_x, image\_y = (image\_width // 2, image\_height // 2)

  upper\_quadrant\_right\_image = lena\_image[0:image\_y, image\_x: image\_width]

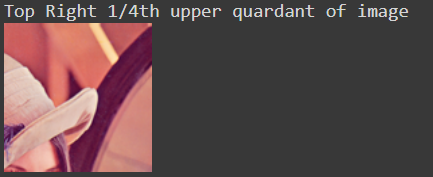
  print("Top Right 1/4th upper quardant of image")

  cv2\_imshow(upper\_quadrant\_right\_image)

**Section-2:**

Output result for the given above methods.

** **

** **

**Section-3:**