**UNIVERSITY OF RUHUNA**

**BACHELOR OF INFORMATION AND COMMUNICATION TECHNOLOGY**

**ICT4163 - Digital Image Processing**

**Skill Assessment – Introduction to Digital Image Processing with Python**

* ***Note that: Download the relevant resources for this practical from the resources folder of the LMS.***

1. Read and display the given image (image\_01.jpg).
2. Access and Display the following properties of the given image.
3. Height,
4. Width
5. Number of channels
6. Total number of pixels
7. Data type
8. Convert the given image (image\_01.jpg) to a grayscale image and display.
9. Resize the given image (image\_1.jpg) according to the following width and height of pixles.
   * Width 270 pixels
   * Height 180 pixels
10. Rotate the given image (image\_1.jpg) by 90 degrees and display.
11. Blurring the given Image using “Gaussian Blur” according to the following kernel size and the standard deviation.

* Kernel size 17,
* Standard Deviation : let it to select automatically

1. Draw a rectangle on the given image (image\_01.jpg) according to the following values and write your index number and the name inside that.

The rectangle need to be started from the top-left corner of the given image according to the given coordinates (x=50, y=50).

* Width: 200px
* Height: 150px

1. Modify the pixel values of the given image at the range of [0:9,0:9] to the given color parameters.

Color parameters [193, 224, 44]

1. Extract an ROI of 100x100 pixels from the top-left corner of the given image and write the new image as “ROI.jpg” in to the Google Colab environment.

10.

1. Draw a rectangle image as a 200 x 200 NumPy array.
2. Draw a white filled rectangle on the black image using CV2. *(Use the (25, 25) as the Top-left corner and the (275, 275) as the Bottom-right corner of the rectangle.)*
3. Draw a rectangle image as a 200 x 200 NumPy array.
4. Draw a filled white circle on the black image according to the following properties.
   * The center of the circle : x=100, y=100
   * The radius of the circle: 100 pixels).
   * Color: White

Perform the Bitwise Operations (AND, OR, XOR, NOT) using the drawn images above ‘b’ and ‘d’