# **Binary Images**

# TASK 1

1. Create a Python program that converts RGB to Binary Images using a threshold.
2. Load trees.png.
3. Convert the image into three binary images using the following thresholds:
   1. Image 1 : 25%
   2. Image 2 : 50%
   3. Image 3 : 75%
4. By storing the images in an array of images and using iteration, display the original image and your 3 outputs using a 2x2 subplot.

# TASK 2:

1. Research and explain Adaptive thresholding.
2. Load ‘creative-proposal.jpg’ as grayscale.
3. Convert the grayscale image to binary using the following methods, displaying the original and 3 results in a 2x2 subplot:
4. 50% thresholding.
5. Adaptive Mean thresholding.
6. Adaptive Gaussian thresholding.

# TASK 3

**Dilation** adds pixels to the boundaries of objects in an **image**, while **erosion** removes pixels on object boundaries. The number of pixels added or removed from the objects in an **image** depends on the size and shape of the ***structuring element*** used to **process** the **image**.

Dilation:

|  |
| --- |
| ORIGINAL DILATED  The value of the output pixel is the maximum value of all pixels in the neighborhood. In a binary image, a pixel is set to 1 if any of the neighboring pixels have the value 1.  Morphological dilation makes objects more visible and fills in small holes in objects. |
| ORIGINAL ERODED  The value of the output pixel is the minimum value of all pixels in the neighborhood. In a binary image, a pixel is set to 0 if any of the neighboring pixels have the value 0.  Morphological erosion removes islands and small objects so that only substantive objects remain. |
|  |

1. Create a Python program to perform dilation and erosion morphological processing as follows:
2. Load j.png.
3. Convert the image into binary using a 50% threshold.
4. Create the following structural element:

**kernel = np.ones((5,5),np.uint8)**

1. Research **cv2.erode** and **cv2.dilate** functions and apply them on your binary image.
2. Explain your results.

# TASK 4 – Opening and Closing

**Opening** is just another name of **erosion followed by dilation,** while **closing** is **dilation followed by erosion**.

1. Load image **opening\_original.png**
2. Research cv2.morphologyEx() to perform opening.
3. Load image **closing\_original.png**
4. Research cv2.morphologyEx() to perform closing.
5. Display all four images in a 2x2 subplot.
6. Comment on the effect and use of opening and closing morphology operations.