**Assignment 1**



**Spring 2025**

**CSE-408 Digital Image Processing**

Submitted by: **Ali Asghar**

Registration No.: **21PWCSE2059**

Class Section: **C**

Submitted to:

**Engr. Mehran Ahmad**

Date:

**21st May 2025**

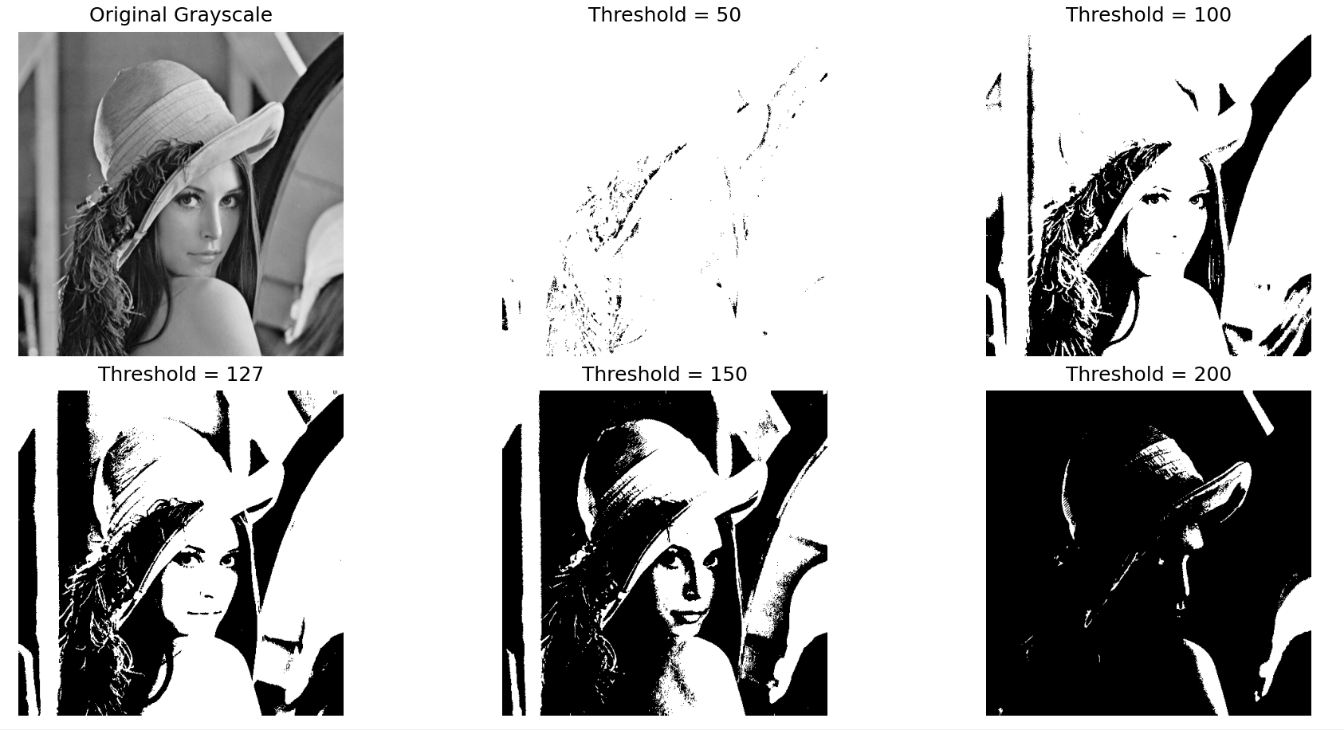
**Department of Computer Systems Engineering**

**University of Engineering and Technology, Peshawar**

**Activity 1:** Implement thresholding using MATLAB or Python. **Show the code and the output result** (original image and thresholded image).

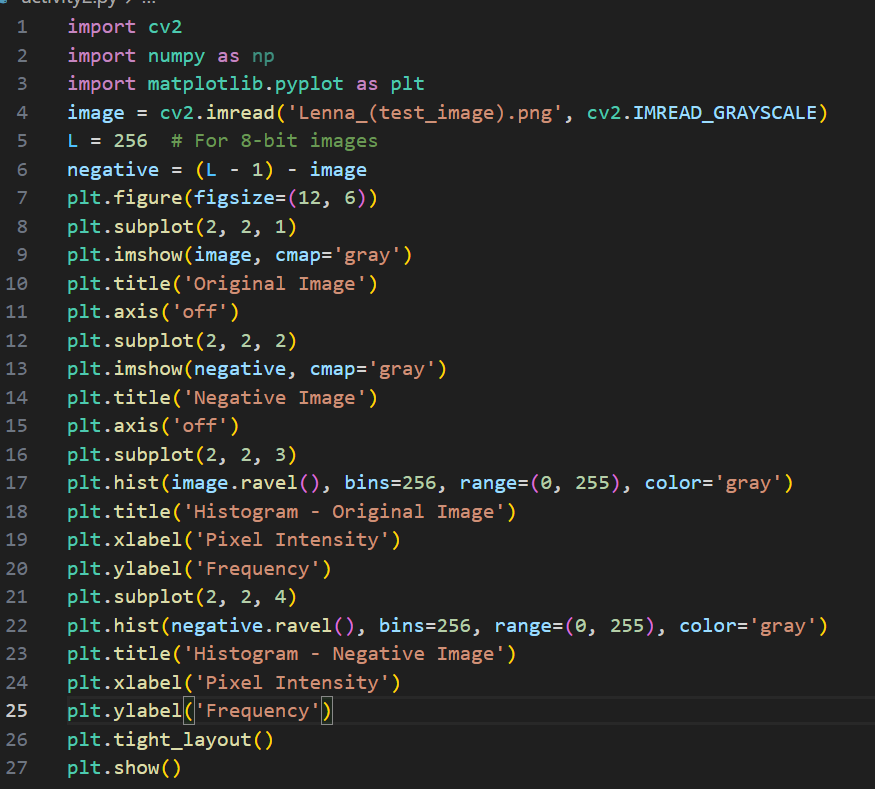
**A screen shot of a computer program

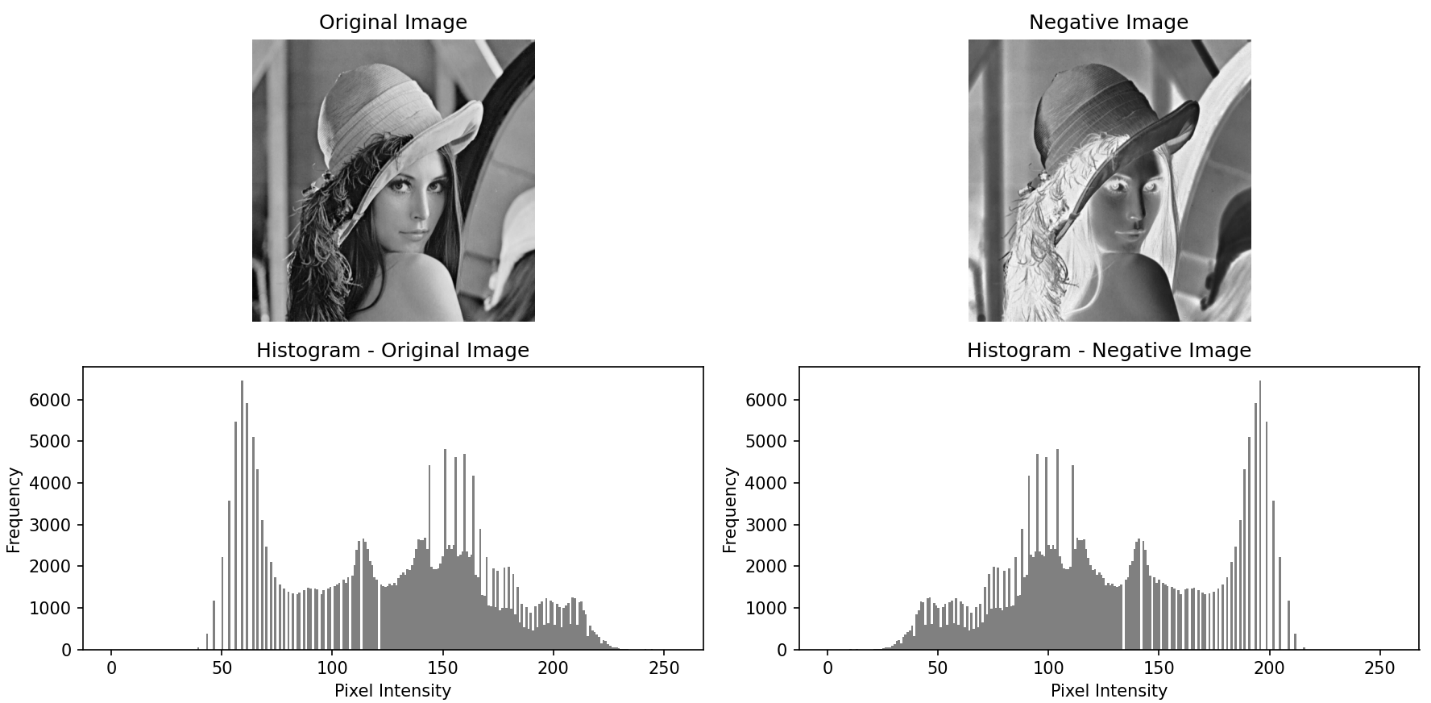
AI-generated content may be incorrect.Code:**

**Output:**

**Analysis:** Lower thresholds (e.g., 50) result in brighter images with poor edge contrast, while higher thresholds (e.g., 200) retain only the brightest regions. An optimal mid-value (e.g., 127) balances detail and contrast effectively for segmentation.

**Activity 2:** Write a MATLAB/Python script to perform negative transformation of an input image. **Show the code and the output result** (original and negative image, along with histograms).

**Code:**

**  
Output:**

**Analysis:** The negative image inverts pixel intensities, making dark areas bright and enhancing hidden details. Its histogram is a mirror of the original, confirming that each intensity III is transformed to 255−I255 - I255−I. This technique is effective for analyzing features in dark regions.

**Activity 3:** Implement a logarithmic transformation in MATLAB/Python. **Show the code and the output result** (original image and log-transformed image).

**Code:**

**Output:**

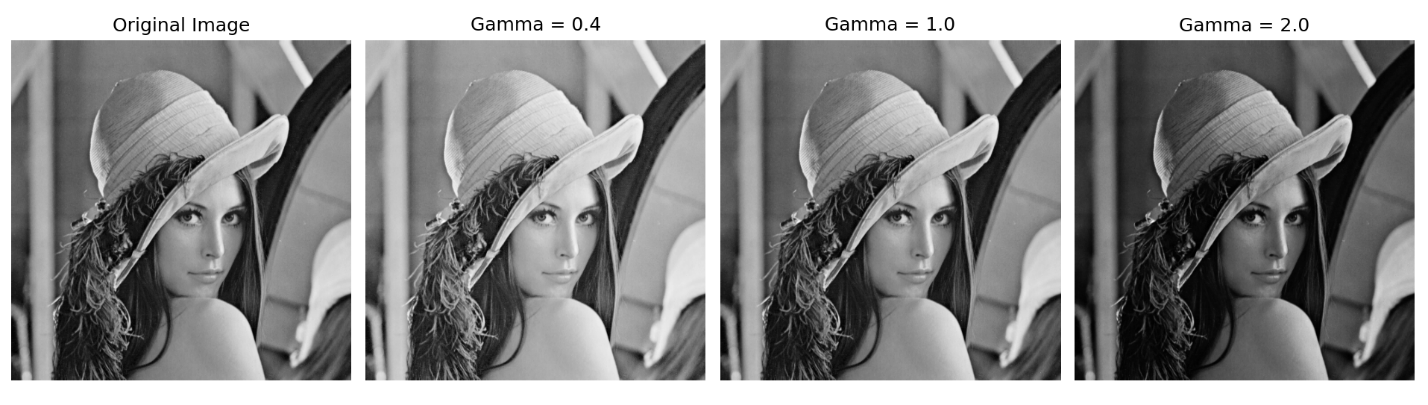
**Analysis:** Logarithmic transformation enhances low-intensity (dark) regions by compressing the dynamic range of pixel values. Varying the constant c controls the degree of enhancement — higher c values make dark areas brighter and more detailed. This is especially useful for images with shadowed or low-contrast features.

**Activity 4:**

Write MATLAB code to apply a power-law transformation on an image using different values of *γ*. **Show the code and the output result** (original image and transformed images for various γ values).

**A screen shot of a computer program

AI-generated content may be incorrect.Code:**

**  
Output:**

**Analysis:** Gamma correction is a nonlinear technique used to adjust image brightness based on a power-law relationship. When **gamma < 1** (e.g., 0.4), the image becomes brighter, enhancing details in dark regions. When **gamma > 1** (e.g., 2.0), the image becomes darker, which helps tone down overly bright areas. This method is especially useful in display systems and image preprocessing to match human visual perception.