MACHINE LEARNING INTERNSHIP
QUALIFICATION TASK 1

Title: **Contour Detection and Analysis for Line and Rectangle Shapes**

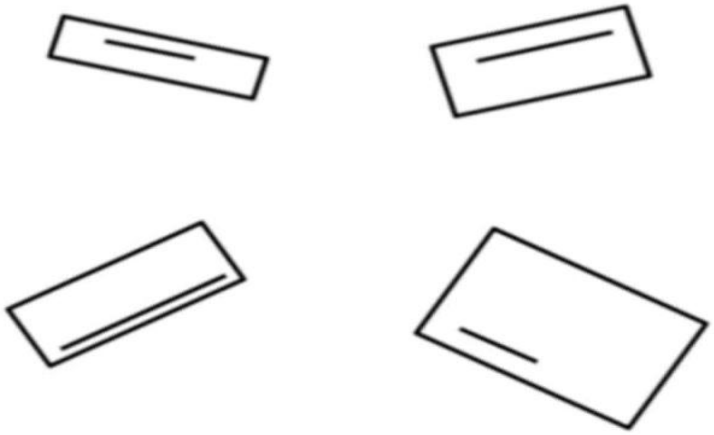
Summary:

This report presents an approach for detecting and analyzing line and rectangle contours in images using OpenCV. The code implementation involves preprocessing steps such as Gaussian blur, grayscale conversion, and Canny edge detection to enhance contour visibility. The contours are then extracted using OpenCV's `findContours` function and analyzed based on their shape properties.

Problem statement:

Assign the number (1 to 4) below the image of the rectangle with respect to its length inside the rectangle. The shorter the line lower the number (No need to reorder the image of the rectangle, only give numbering)

Image:



Tools & Technology:

- Python
- OpenCV
- vs-code

Approach:

Preprocessing: The image is loaded and preprocessed by applying Gaussian blur to reduce noise and converting it to grayscale.

1. **Contour Detection**: Canny edge detection is applied to the grayscale image to obtain the contours.
2. **Contour Analysis**: The `getContours` function is used to extract line contours and rectangle contours separately. The function calculates properties like area, bounding box, center, size, and angle for each contour.
3. **Line Contour Processing**: Line contours with small area and size are filtered out, and similar line contours are grouped together. The resulting line contours are sorted based on their lengths.
4. **Rectangle Contour Processing**: The rectangle contours are processed individually, and their centers are used to determine their numbering order based on the sorted line contours.

Output: