Q1

The formula for computing the negative of a grayscale image is:

$$s = (L - 1) - r$$

where:

- r is the original pixel intensity.
- $L - 1$ is the maximum intensity in the image (255 for an 8-bit grayscale image).
- s is the transformed (negative) pixel intensity.

Applying this formula to each element in the given matrix:

$$\begin{bmatrix} 50 & 100 & 150 \\ 200 & 225 & 255 \\ 30 & 90 & 180 \end{bmatrix}$$

We calculate:

$$\begin{bmatrix} (255-50) & (255-100) & (255-150) \\ (255-200) & (255-225) & (255-255) \\ (255-30) & (255-90) & (255-180) \end{bmatrix}$$

$$\begin{bmatrix} 205 & 155 & 105 \\ 55 & 30 & 0 \\ 225 & 165 & 75 \end{bmatrix}$$

B

Analysis of What Happened to the Given Image Patch

1. **Bright Pixels Became Dark, and Dark Pixels Became Bright:**
   - The highest intensity pixel (255) turned into the lowest (0), and vice versa.
   - For example:
     - 50→205(a darker region became much brighter)
     - 225→30(a bright region became very dark)

Contrast Is Reversed:

2. Lighter areas in the original image now appear dark in the negative, and darker areas now appear bright.
3. If this image represents a grayscale scene, objects that were once highlighted by light reflections may now look like shadows, and vice versa.

Question2

(a) Compute the Transformed Values

We compute s for each given intensity r:

For $r = 5$:

$$s = 50 \cdot \log(1 + 5) = 50 \cdot \log(6)$$

Using $\log(6) \approx 1.7918$,

$$s = 50 \times 1.7918 = 89.59$$

For $r = 50$:

$$s = 50 \cdot \log(1 + 50) = 50 \cdot \log(51)$$

Using $\log(51) \approx 3.9318$,

$$s = 50 \times 3.9318 = 196.59$$

For $r = 150$:

$$s = 50 \cdot \log(1 + 150) = 50 \cdot \log(151)$$

Using $\log(151) \approx 5.0173$,

$$s = 50 \times 5.0173 = 250.86$$

For $r = 250$:

$$s = 50 \cdot \log(1 + 250) = 50 \cdot \log(251)$$

Using $\log(251) \approx 5.5255$,

$$s = 50 \times 5.5255 = 276.27$$

(b)

The values **250.86 and 276.27** are saturated to **200**.

Low intensities are enhanced, making dark details more visible.
High intensities get compressed and clipped, reducing contrast in bright regions.
Overall effect: Better visibility in darker parts of the image but loss of distinction in very bright regions.

Question3

(a) Compute the New Values for Given Intensities

1. **For $r = 85$:**

$$s = \frac{(85 - 80)}{(180 - 80)} \times 255 = \frac{5}{100} \times 255 = 12.75 \approx 13$$

2. **For $r = 120$:**

$$s = \frac{(120 - 80)}{(180 - 80)} \times 255 = \frac{40}{100} \times 255 = 102$$

3. **For $r = 160$:**

$$s = \frac{(160 - 80)}{(180 - 80)} \times 255 = \frac{80}{100} \times 255 = 204$$

4. **For $r = 175$:**

$$s = \frac{(175 - 80)}{(180 - 80)} \times 255 = \frac{95}{100} \times 255 = 242.25 \approx 242$$

(b) Handling Out-of-Range Intensity $r = 70$

The given transformation is defined only for pixel values within the range $[80, 180]$. If a pixel falls **outside** this range:

- **If $r < R_{\min}$** (e.g., $r = 70$):
  - Since 70 is below $R_{\min} = 80$, it is **clipped** to the minimum output value, which is **0**.
- **If $r > R_{\max}$**:
  - If a pixel was greater than 180, it would be mapped to the maximum intensity (255).

Thus, for $r = 70$, **the transformed intensity should be 0**.

Question4

- $g[0] = 20$
- $g[1] = 25 - 20 = 5$
- $g[2] = 30 - 25 = 5$
- $g[3] = 35 - 30 = 5$
- $g[4] = 100 - 35 = 65$
- $g[5] = 200 - 100 = 100$
- $g[6] = 210 - 200 = 10$
- $g[7] = 215 - 210 = 5$
- $g[8] = 220 - 215 = 5$
- $g[9] = 50 - 220 = -170$

(We stop at $g[9]$ because we ignore a gradient calculation for the last pixel.)

Step 2 – Identify Regions of High Gradient:

Looking at the absolute values:

- The jump from 35 to 100 (gradient 65)
- A larger jump from 100 to 200 (gradient 100)
- And a very large drop from 220 to 50 (gradient -170, absolute value 170)

In the context of lane detection, high gradients indicate sharp intensity changes—likely corresponding to edges of lane markings or boundaries between the road and an obstacle.

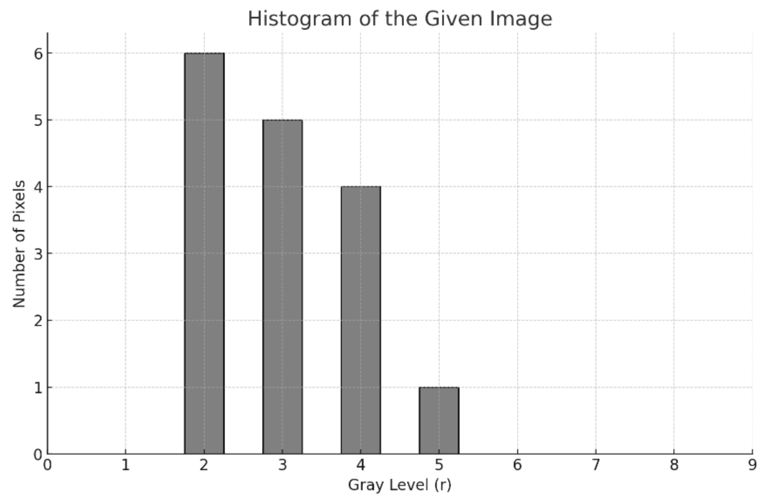
- The gradient of 100 suggests a strong edge (possibly the edge of a lane marking).
- The very high gradient of 170 (in absolute terms) likely marks a very distinct boundary (for example, from a bright road marking to a dark surrounding area).

b) Choosing the Right Edge Detector for Rainy Conditions:

Under rainy conditions, images tend to be blurred and noisy.

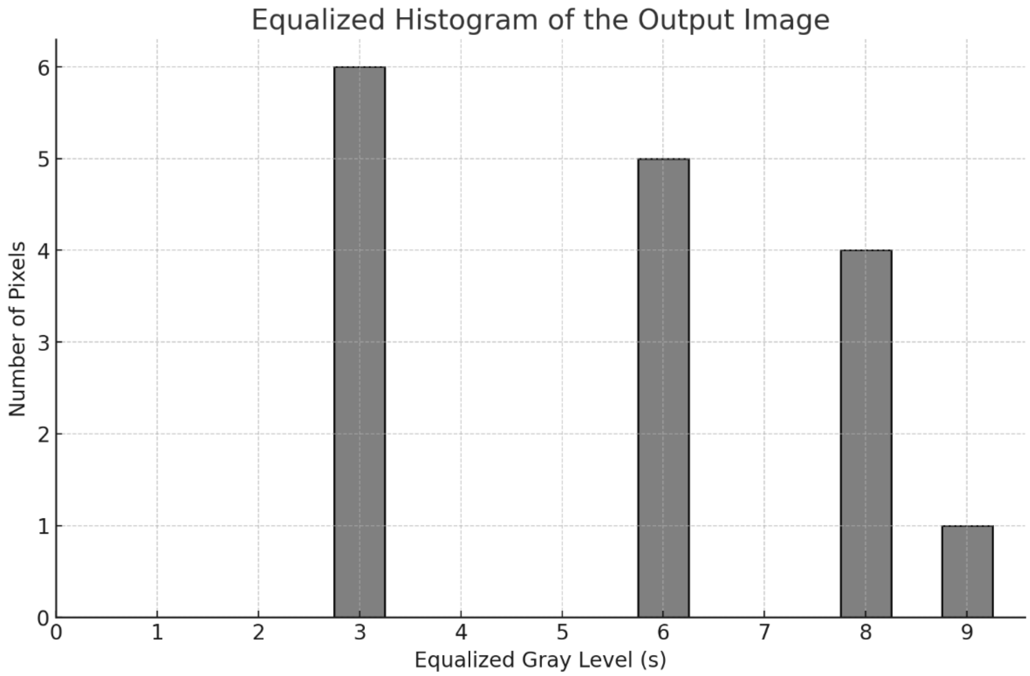
- **Roberts Cross:** Uses a very small (2×2) kernel; it is very sensitive to noise and less robust to blurring.
- **Prewitt:** Slightly larger kernels are used, but it still does not provide much smoothing.
- **Sobel:** Uses a 3×3 kernel that not only computes the gradient but also incorporates a smoothing effect due to the weighting of the center row/column. This makes Sobel more robust against noise and blurring.

Question5



Gray Level (r)	2	3	4	5
No. of Pixels	6	5	4	1
Cumulative Sum	6	11	15	16
Normalized CDF	0.375	0.6875	0.9375	1
s x 9	3	6	8	9

3	**6**	**6**	**3**
8	**3**	**8**	**6**
6	**3**	**6**	**9**
3	**8**	**3**	**8**


Equalized Histogram of the Output Image
Number of Pixels
Equalized Gray Level (s)
0
1
2
3
4
5
6
7
8
9