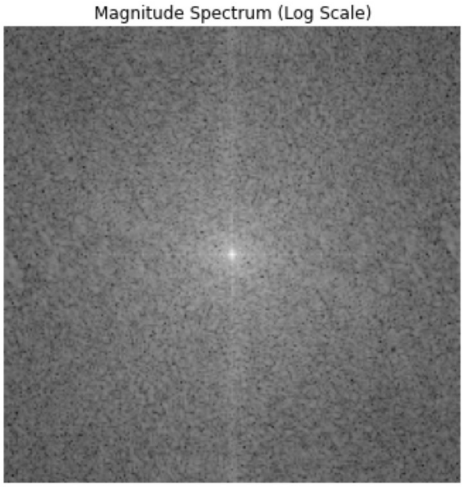
In [1]:

```python
# Import necessary libraries
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Step 1: Load Image
image_path = 'C:/Users/Student/Documents/Expimage.jpg'
img = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)

# Step 2: Apply Discrete Fourier Transform (DFT)
# Perform 2D Fourier Transform
dft = np.fft.fft2(img)
# Shift the zero frequency component to the center
dft_shifted = np.fft.fftshift(dft)

# Step 3: Compute the Magnitude Spectrum
# Compute the magnitude of the DFT
magnitude = np.abs(dft_shifted)
# Use log scaling for better visualization
magnitude_spectrum = np.log(1 + magnitude)

# Step 4: Display the Results
# Create a subplot with 2 images: original and magnitude spectrum
fig, axes = plt.subplots(1, 2, figsize=(15, 5))

# Display the original image
axes[0].imshow(img, cmap='gray')
axes[0].set_title('Original Image')
axes[0].axis('off')

# Display the magnitude spectrum
axes[1].imshow(magnitude_spectrum, cmap='gray')
axes[1].set_title('Magnitude Spectrum (Log Scale)')
axes[1].axis('off')

# Show the plots
plt.tight_layout()
plt.show()

# Step 5: (Optional) Inverse DFT (to recover the image)
# Inverse shift to revert the zero-frequency component to its original position
dft_ishifted = np.fft.ifftshift(dft_shifted)
# Perform Inverse 2D Fourier Transform
img_back = np.fft.ifft2(dft_ishifted)
# Take the real part of the inverse transformed image
img_back = np.real(img_back)

# Display the recovered image from the inverse DFT
plt.figure(figsize=(7, 7))
plt.imshow(img_back, cmap='gray')
plt.title('Recovered Image from Inverse DFT')
plt.axis('off')
plt.show()
```


Original Image


Magnitude Spectrum (Log Scale)


Recovered Image from Inverse DFT

In [ ]: