

1. Write a MATLAB program to implement LSB steganography

Program

```
cover_image = imread('cameraman.tif');
binary_message = randi([0 1], 1, numel(cover_image));
cover_image = double(cover_image);
binary_message = double(binary_message);
num_bits_needed = numel(binary_message);
if numel(cover_image) < num_bits_needed
    error('Cover image is too small to store the entire message.');
```

end

```
modified_image = cover_image;
bit_count = 1;
for i = 1:size(cover_image, 1)
    for j = 1:size(cover_image, 2)
        pixel_value = cover_image(i, j);
        if bit_count <= num_bits_needed
            bit_to_embed = binary_message(bit_count);
        else
            break;
        end
        modified_pixel = bitset(pixel_value, 1, bit_to_embed);
        modified_image(i, j) = modified_pixel;
        bit_count = bit_count + 1;
    end
end
subplot(1, 2, 1);
imshow(uint8(cover_image));
title('Original Image');
```

subplot(1, 2, 2);

```
imshow(uint8(modified_image));
title('Stego Image');
```

imwrite(uint8(modified_image), 'stego_image.tif');

```
extracted_message = zeros(1, num_bits_needed);
bit_count = 1;
for i = 1:size(cover_image, 1)
    for j = 1:size(cover_image, 2)
        modified_pixel = modified_image(i, j);
        extracted_bit = bitget(modified_pixel, 1);
        if bit_count <= num_bits_needed
            extracted_message(bit_count) = extracted_bit;
        else
            break;
        end
        bit_count = bit_count + 1;
    end
end
disp('Extracted Message:');
disp(extracted_message);
```

Output

Original Image



Stego Image



2. Write a Program to implement Difference Expansion

Program

```
a = imread('cameraman.tif');
a = double(a);
block = a(1:256, 1:256);
p = zeros(256, 256);
d = zeros(256, 256);
for i = 1:256
    for j = 1:256
        if i == 1 && j == 1
            p(i, j) = block(i, j);
        elseif i == 1
            p(i, j) = block(i, j) + p(i, j-1);
        elseif j == 1
            p(i, j) = block(i, j) + p(i-1, j);
        else
            p(i, j) = block(i, j) + (p(i-1, j) + p(i, j-1)) / 2;
        end
    end
end
for i = 1:256
    for j = 1:256
        if i == 1 && j == 1
            d(i, j) = block(i, j) - p(i, j);
        elseif i == 1
            d(i, j) = block(i, j) - (block(i, j-1) + p(i, j-1)) / 2;
        elseif j == 1
            d(i, j) = block(i, j) - (block(i-1, j) + p(i-1, j)) / 2;
        else
            d(i, j) = block(i, j) - (block(i-1, j) + block(i, j-1)) / 2;
        end
    end
end
expanded_block = zeros(256, 256);
for i = 1:256
    for j = 1:256
        if i == 1 && j == 1
```

```

expanded_block(i, j) = p(i, j) + d(i, j);
elseif i == 1
expanded_block(i, j) = block(i, j-1) + p(i, j-1) + d(i, j);
elseif j == 1
expanded_block(i, j) = block(i-1, j) + p(i-1, j) + d(i, j);
else
expanded_block(i, j) = block(i-1, j) + block(i, j-1) + d(i, j);
end
end
end
expanded_block = uint8(expanded_block);
figure;
subplot(1, 2, 1);
imshow(uint8(block));
title('Original 4x4 Block');
subplot(1, 2, 2);
imshow(expanded_block);
title('Expanded 4x4 Block');

```

Output

Original 4x4 Block Expanded 4x4 Block

