```
% Problem 1: Extraction Bitplanes from Image and Image Reconstruction from
% Bitplanes.
clear all;

%-----Part 1 : Read the original image :slope.tif', which is an 8 bits/pixel gray scale image.
slope_image = imread('slope.tif'); % reading the image from an external source with the help of imread funciton.

%-----Part 2 : Extract its 8 bitplanes as 8 binary images. storing image
%information in new variable
converted_slope_image = double(slope_image); % converting each pixel of the image in double.

% extracting all bit one by one from 1st to 8th in variable from bitplane_1 to bitplane_8 respectively.
% We are converting all the images into binary format. That is why we have
% used mod here.
bit_plane_8 = mod(converted_slope_image,2); % Least Significant Bitplane
bit_plane_7 = mod(floor(converted_slope_image/2),2);
bit_plane_6 = mod(floor(converted_slope_image/4),2);
bit_plane_5 = mod(floor(converted_slope_image/8),2);
bit_plane_4 = mod(floor(converted_slope_image/16),2);
bit_plane_3 = mod(floor(converted_slope_image/32),2);
bit_plane_2 = mod(floor(converted_slope_image/64),2);
bit_plane_1 = mod(floor(converted_slope_image/128),2); % Most Significant Bitplane

%imshow(a)

%-----Part 3 : Reconstruct 8 versions of the original image.
% Reconstructing the image using the most significant (upper) bitplane only.
reconstructed_image_1 = (2 * (2 * (2 * (2 * (2 * (2 * (2 * bit_plane_1)))))); % 128 * bitplane1
reconstructed_image_1 = uint8(reconstructed_image_1); % It will convert each and every pixel value of reconstructed_image_1 into the range of 0 to 255.

%reconstructing the image using upper 2 bitplanes only
reconstructed_image_2 = (2 * (2 * (2 * (2 * (2 * (2 * (2 * bit_plane_1 + bit_plane_2)))))); % 128 * bitplane1 + 64 * bitplane2
reconstructed_image_2 = uint8(reconstructed_image_2); % It will convert each and every pixel value of reconstructed_image_2 into the range of 0 to 255.

%reconstructing the image using upper 3 bitplanes only
reconstructed_image_3 = (2 * (2 * (2 * (2 * (2 * (2 * (2 * bit_plane_1 + bit_plane_2) + bit_plane_3)))))); % 128 * bitplane1 + 64 * bitplane2 + 32 * bitplane3
reconstructed_image_3 = uint8(reconstructed_image_3); % It will convert each and every pixel value of reconstructed_image_3 into the range of 0 to 255.

%reconstructing the image using upper 4 bitplanes only
reconstructed_image_4 = (2 * (2 * (2 * (2 * (2 * (2 * (2 * bit_plane_1 + bit_plane_2) +
```

```
bit_plane_3) + bit_plane_4))))); % 128 * bitplane1 + 64 * bitplane2 + 32 * bitplane3 +↙
16 * bitplane4
reconstructed_image_4 = uint8(reconstructed_image_4); % It will convert each and every↙
pixel value of reconstructed_image_4 into the range of 0 to 255.

%reconstructing the image using upper 5 bitplanes only
reconstructed_image_5 = (2 * (2 * (2 * (2 * (2 * (2 * (2 * bit_plane_1 + bit_plane_2) +↙
bit_plane_3) + bit_plane_4) + bit_plane_5) ))); % 128 * bitplane1 + 64 * bitplane2 + 32↙
* bitplane3 + 16 * bitplane4 + 8 * bitplane5
reconstructed_image_5 = uint8(reconstructed_image_5); % It will convert each and every↙
pixel value of reconstructed_image_5 into the range of 0 to 255.

%reconstructing the image using upper 6 bitplanes only
reconstructed_image_6 = (2 * (2 * (2 * (2 * (2 * (2 * (2 * bit_plane_1 + bit_plane_2) +↙
bit_plane_3) + bit_plane_4) + bit_plane_5) + bit_plane_6) )); % 128 * bitplane1 + 64 *↙
bitplane2 + 32 * bitplane3 + 16 * bitplane4 + 8 * bitplane5 + 4 * bitplane6
reconstructed_image_6 = uint8(reconstructed_image_6); % It will convert each and every↙
pixel value of reconstructed_image_6 into the range of 0 to 255.

%reconstructing the image using upper 7 bitplanes only
reconstructed_image_7 = (2 * (2 * (2 * (2 * (2 * (2 * (2 * bit_plane_1 + bit_plane_2) +↙
bit_plane_3) + bit_plane_4) + bit_plane_5) + bit_plane_6) + bit_plane_7) ); % 128 *↙
bitplane1 + 64 * bitplane2 + 32 * bitplane3 + 16 * bitplane4 + 8 * bitplane5 + 4 *↙
bitplane6 + 2 * bitplane7
reconstructed_image_7 = uint8(reconstructed_image_7); % It will convert each and every↙
pixel value of reconstructed_image_7 into the range of 0 to 255.

%reconstructing the image using upper 8 bitplanes only
reconstructed_image_8 = (2 * (2 * (2 * (2 * (2 * (2 * (2 * bit_plane_1 + bit_plane_2) +↙
bit_plane_3) + bit_plane_4) + bit_plane_5) + bit_plane_6) + bit_plane_7) +↙
bit_plane_8); % 128 * bitplane1 + 64 * bitplane2 + 32 * bitplane3 + 16 * bitplane4 + 8↙
* bitplane5 + 4 * bitplane6 + 2 * bitplane7 + 1 * bitplane8
reconstructed_image_8 = uint8(reconstructed_image_8); % It will convert each and every↙
pixel value of reconstructed_image_8 into the range of 0 to 255.

%-----Part 4 : Create 4 figures, each with 4 sub-figures of 2x2 layout
% In Figure 1, show the upper 4 bitplanes in 4 sub-figures.
figure; % figure creates figure graphics objects. figure objects are the individual↙
windows on the screen in which MATLAB displays graphical output.
subplot(2,2,1) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=1.
imshow(bit_plane_1) % It will display the gray-scale image in the figure.
title('Bitplane_1 image') % It will add the specified title for the current plot.
subplot(2,2,2) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=p=2.
imshow(bit_plane_2) % It will display the gray-scale image in the figure.
title('Bitplane_2 image') % It will add the specified title for the current plot.
subplot(2,2,3) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=3.
imshow(bit_plane_3) % It will display the gray-scale image in the figure.
```

```
title('Bitplane_3 image') % It will add the specified title for the current plot.
subplot(2,2,4) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=4.
imshow(bit_plane_4) % It will display the gray-scale image in the figure.
title('Bitplane_4 image') % It will add the specified title for the current plot.

% In Figure 2, show the lower 4 bitplanes in 4 sub-figures.
figure; % figure creates figure graphics objects. figure objects are the individual↙
windows on the screen in which MATLAB displays graphical output.
subplot(2,2,1) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=1.
imshow(bit_plane_5) % It will display the gray-scale image in the figure.
title('Bitplane_5 image') % It will add the specified title for the current plot.
subplot(2,2,2) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=p=2.
imshow(bit_plane_6) % It will display the gray-scale image in the figure.
title('Bitplane_6 image') % It will add the specified title for the current plot.
subplot(2,2,3) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=3.
imshow(bit_plane_7) % It will display the gray-scale image in the figure.
title('Bitplane_7 image') % It will add the specified title for the current plot.
subplot(2,2,4) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=4.
imshow(bit_plane_8) % It will display the gray-scale image in the figure.
title('Bitplane_8 image') % It will add the specified title for the current plot.

% Figure 3, show the first 4 reconstructed images in 4 sub-figures, corresponding to↙
using the upper 1, 2, 3, and 4 bitplanes, respectively.
figure; % figure creates figure graphics objects. figure objects are the individual↙
windows on the screen in which MATLAB displays graphical output.
subplot(2,2,1) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=1.
imshow(reconstructed_image_1) % It will display the gray-scale image in the figure.
title('Reconstructed Image using bitplane 1') % It will add the specified title for the↙
current plot.
subplot(2,2,2) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=p=2.
imshow(reconstructed_image_2) % It will display the gray-scale image in the figure.
title('Reconstructed Image using bitplanes 1-2') % It will add the specified title for↙
the current plot.
subplot(2,2,3) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=3.
imshow(reconstructed_image_3) % It will display the gray-scale image in the figure.
title('Reconstructed Image using bitplanes 1-3') % It will add the specified title for↙
the current plot.
subplot(2,2,4) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=4.
imshow(reconstructed_image_4) % It will display the gray-scale image in the figure.
title('Reconstructed Image using bitplanes 1-4') % It will add the specified title for↙
the current plot.
```

```
% In Figure 4, show the remaining 4 reconstructed images in 4 sub-figures,↙
corresponding to using the upper 5, 6, 7 and 8 bitplanes, respectively.
figure; % figure creates figure graphics objects. figure objects are the individual↙
windows on the screen in which MATLAB displays graphical output.
subplot(2,2,1) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=1.
imshow(reconstructed_image_5) % It will display the gray-scale image in the figure.
title('Reconstructed Image using bitplanes 1-5') % It will add the specified title for↙
the current plot.
subplot(2,2,2) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=p=2.
imshow(reconstructed_image_6) % It will display the gray-scale image in the figure.
title('Reconstructed Image using bitplanes 1-6') % It will add the specified title for↙
the current plot.
subplot(2,2,3) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=3.
imshow(reconstructed_image_7) % It will display the gray-scale image in the figure.
title('Reconstructed Image using bitplanes 1-7') % It will add the specified title for↙
the current plot.
subplot(2,2,4) % subplot(m, n, p) divides the current figure into an m-by-n grid and↙
creates axes in the position specified by p. Here, m=n=2, p=4.
imshow(reconstructed_image_8) % It will display the gray-scale image in the figure.
title('Reconstructed Image using bitplanes 1-8') % It will add the specified title for↙
the current plot.
```