Read an image and then convert all it pixel values to gray scale. Follow the steps of the following functions:

**Function: sCircleDiameterF()**

Input: Input metering area % and meter scaling factor

Output: Diameter of spot metering circle

Methods:

1. Step1: Area of the spot metering circle is measured by taking a predefined percentage (input metering area) of pixel from total pixel of an image and then divided it by scaling factor.
2. Step 2: We then measure the *diameter of the circle* by using equation where is represented by total number of spot metering circle.

**Function: trimImageF()**

Input: Grayscale image data and the diameter of spot metering circle.

Output: Grayscale trim image data, index of all A squares and length of final A square.

Methods:

1. Step1: If diameter of the spot metering is odd we change it to even by *subtracting* a pixel.
2. Step2: Store all *even diameter length* of spot metering circle which is the 100% to 75%-pixel value of original diameter and count them as the side of A square.
3. Step 3: Calculate the total number of *dropped pixels* for the different side length of A square and the *final length of A square* will be a length that produce minimum pixel loss.
4. Step 4: Drop the pixel from left, right, top and bottom of an image and we will get a grayscale *trim image*.
5. Step 5: Calculate the *index of all A square side* (a 1 by 4 matrix).

**Function: gScaleAveragingF()**

Input: Grayscale trim image data, final A square side length and a linear averaging table.

Output: Gray scale trim image weighted value and average weighted pixel value in different A squares.

Methods:

1. Step1: For every A square of the trimmed image

* Separate all its pixels
* Count the number of pixels in A square
* Multiply a predefined weight to all pixel value
* Sum all the weighted pixel value and then take their average.
* Replace the calculated average to all the original pixels of A square.

**Function: gScaleAvgEntropyF()**

Input: Weighted value of gray scale trim image

Output: Total entropy of weighed gray scale trim image, entropy of individual (weighted) pixel value and frequency of individual (weighted) pixel value.

Methods:

1. Step1: For every weighted pixel value

- Count the number of pixel in the trim image and then measure its probability

- Measure the entropy of every individual pixel value

- Measure the average entropy for all weighted pixel.

**Function: ASquareZTableAvgF()**

Input: Grayscale trim image data, final A square side length and zone table value.

Output: Zone mapped image value and average zone mapped value in different A squares.

Methods:

1. Step1: For every A square of the trimmed image

* Separate all its pixels
* Count the number of pixels
* Replace every pixel value by its zone mapped value
* Sum all zone mapped value and then take their average.
* Replace the average of zone mapped value to all the original pixels of A square.

**Function: zTableEntropyF()**

Input: Zone mapped trim image value

Output: Total entropy of zone mapped trim image, entropy of individual zone mapped value and the frequency of individual zoned mapped value.

Methods:

1. Step1: For every zone mapped value

- Count the number of pixel in the image that has that zone mapped value and measure its probability

- Measure the entropy for every zone mapped value

- Measure the average entropy for all zone mapped value.

**Function: wZoneTableAvgEntF()**

Input: Zone mapped image value, final A square side length and the average zone mapped value of different squares.

Output: Average entropy for all zone mapped block, entropy of individual zone mapped block, and the frequency of individual zone mapped block.

Methods:

1. Step1: Count the total number of possible A square from zone mapped image by

- Dividing the length of A square side by 2 which will be the length of every block

- Count how many possible A square block in a unit width and unit height of the image. If there are total x A square block in unit width and total y A square block in unit height then total no of possible A square block is (2x-1)\*(2y-1).

2. Step1:

1. Step2: Assign the average zone mapped value to all its corresponding block
2. Step 3: For every zone mapped value

-Count how many possible A square block has same value. We can do it by checking every block and its 3 neighbor bocks right, bottom and diagonal

- Measure the probability

- Calculate the entropy for every zone mapped value based on that probability

- Also measure the average entropy.