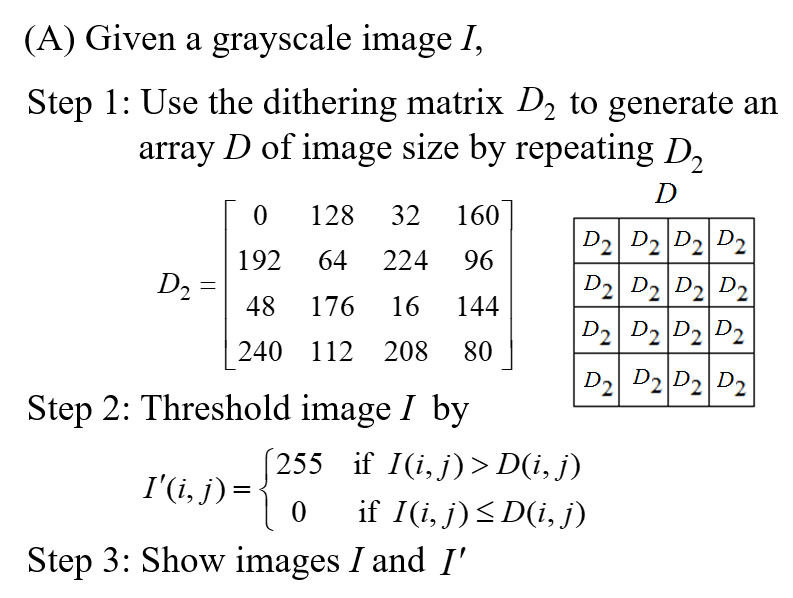
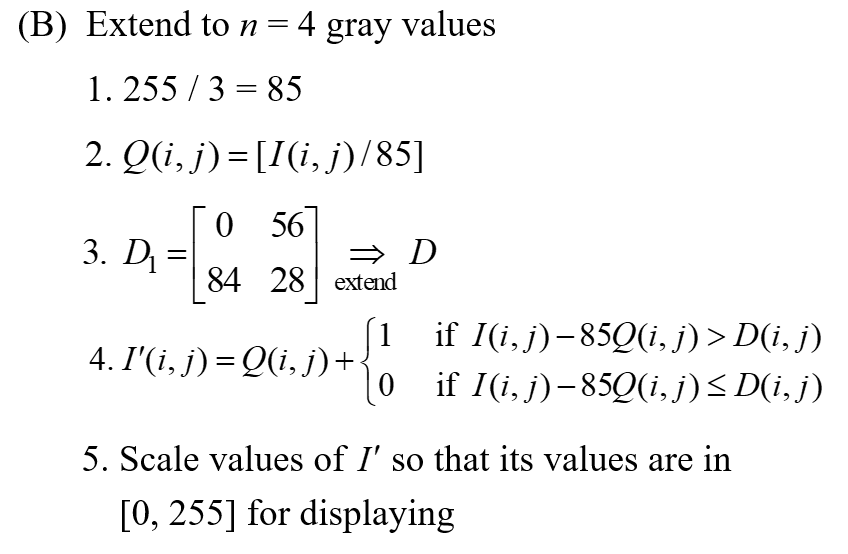
**40847041S朱自宇 hw2**

**Problem statement:**

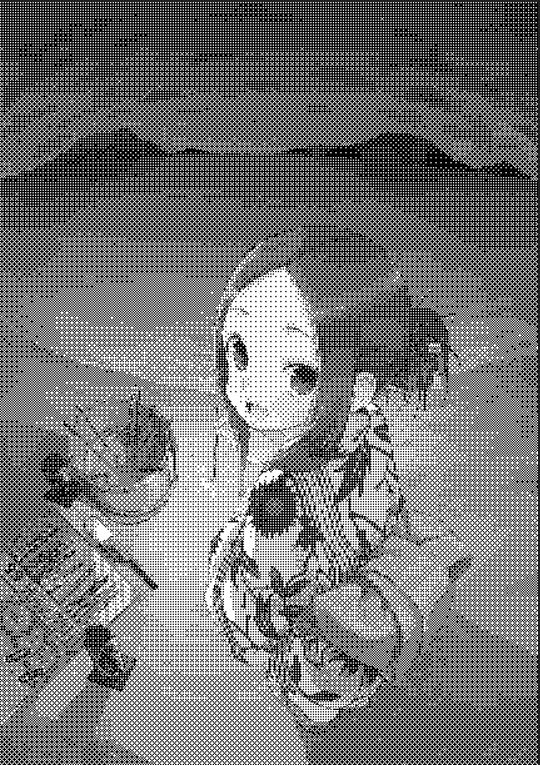
**Experimental results:**

* **Input/Output images:**

**Input grayscale image I**



**Output grayscale image I’ in partA:**

****

**Output grayscale image I’ in partB:**

****

* **Source code:** **import numpy as np**

import cv2

from PIL import Image

# read the image

grayimg = cv2.imread("gray\_takagisan.png", cv2.IMREAD\_GRAYSCALE)

grayimgArr = np.array(grayimg)

# print out the original picture size

# print(img.shape)

"""

Part A:

"""

# show the picture

cv2.imshow("Original gray image", grayimg)

"""

Step1: generate an array D of image size

"""

# image size

(height, width) = grayimg.shape

# dithering matrix D2

ditherMat = np.array([[0, 128, 32, 160],

[192, 64, 224, 96],

[48, 176, 16, 144],

[240, 112, 208, 80]])

# ///////////////////////////#

# set up the dithering array #

#////////////////////////////#

# fill the image with 0s to fit the dithering Matrix size

qx = height // 4

qy = width // 4

resizeImg = np.zeros([(qx+1)\*4, (qy+1)\*4], dtype='uint8')

resizeImg[0:height,0:width] = grayimg

# now let's generate the array D for thersholding

dithertable = np.tile(ditherMat,(qx+1, qy+1))

"""

Step2: Threshold the image

"""

# thresholding

threshold = (resizeImg > dithertable )\*255

outsize = threshold[0:height,0:width]

# output the image I'

out = Image.fromarray(outsize)

out.show(title="PartA")

"""

Part B:

"""

# set up another picture

qx2 = height // 2

qy2 = width // 2

resizeImg2 = np.zeros([(qx2+1)\*2, (qy2+1)\*2], dtype='uint8')

resizeImg2[0:height,0:width] = grayimg//85

ditherMat2 = np.array([[0, 56],

[84, 28]])

# now let's generate the array D for thersholding

dithertable2 = np.tile(ditherMat2,(qx2+1, qy2+1))

# do threshold for partB

threshold2 = resizeImg2 + ( (resizeImg - resizeImg2\*85) > dithertable2 )\*1

outsize2 = threshold2[0:height,0:width]\*85

out2 = Image.fromarray(outsize2)

out2.show(title="PartB")

# wait until the user press any button to close the img window

cv2.waitKey(0)

cv2.destroyAllWindows()

* **Comments:**

To modify the picture data, I have to use the “numpy” library to change the image to an array. There are some special usage about it, it took me some time, but I finally made it：).