HW5
r10944013 網媒所碩一 馮啟倫

HW5-1:
目標：Gray-scale dilation

演算法：
跟上次作業類似，只是這次是抓去kernel中最大值，去填補kernel中間位置的數值

Code segment

```
#hw5-1
def dilation(img, kernel):
    print("Process for gray-scale dilation image...")
    img_dilation = np.zeros((512,512))
    for row in range(img.shape[0]):
        for col in range(img.shape[1]):
            if img[row][col] > 0: #kernel中心點跟img對齊, 且大於0
                MaxValue=-1
                #找max數值
                for y,x in kernel:
                    if (y+row) >=0 and (y+row) < 512 and (x+col) >= 0 and (x+col) <512:
                        if img[y+row][x+col] > MaxValue:
                            MaxValue = img[y+row][x+col]
                for y,x in kernel:
                    if (y+row) >=0 and (y+row) < 512 and (x+col) >= 0 and (x+col) <512:
                        img_dilation[y+row][x+col] = MaxValue
    return img_dilation
```

Result picture:

HW5-2:
目標：gray-scale erosion

演算法：

針對目前的kernel所覆蓋到的像素點，選取裡面其中的最小值，把它當作kernel中間的數值並填補到原本圖片上面

Code segment:

```
#hw5-2
def erosion(img, kernel):
    print("Process for erosion image...")
    img_erosion = np.zeros((512,512))
    for row in range(img.shape[0]):
        for col in range(img.shape[1]):
            MinValue = 256
            for y,x in kernel:
                if (y+row) (variable) y: Any 512 and (x+col) >= 0 and (x+col) <512:
                    if img[y+row][x+col] < MinValue:
                        MinValue = img[y+row][x+col]
            img_erosion[row][col] = MinValue
    return img_erosion
```

Result picture:

HW5-3:

目標：gray-scale opening

演算法：
先做erosion, 再做dilation

Code segment:

```python
#hw5-3
#先做erosion, 再做dilation
def opening(img, kernel):
    print("Process for opening image...")
    return dilation(erosion(img, kernel), kernel)
```

Result picture:

HW5-4:

目標：gray-scale closing

演算法：
先做dilation, 再做erosion

Code segment:

```python
#hw5-4
#先做dilation, 再做erosion
def closing(img, kernel):
    print("Process for closing image...")
    return erosion(dilation(img, kernel), kernel)
```

Result picture: