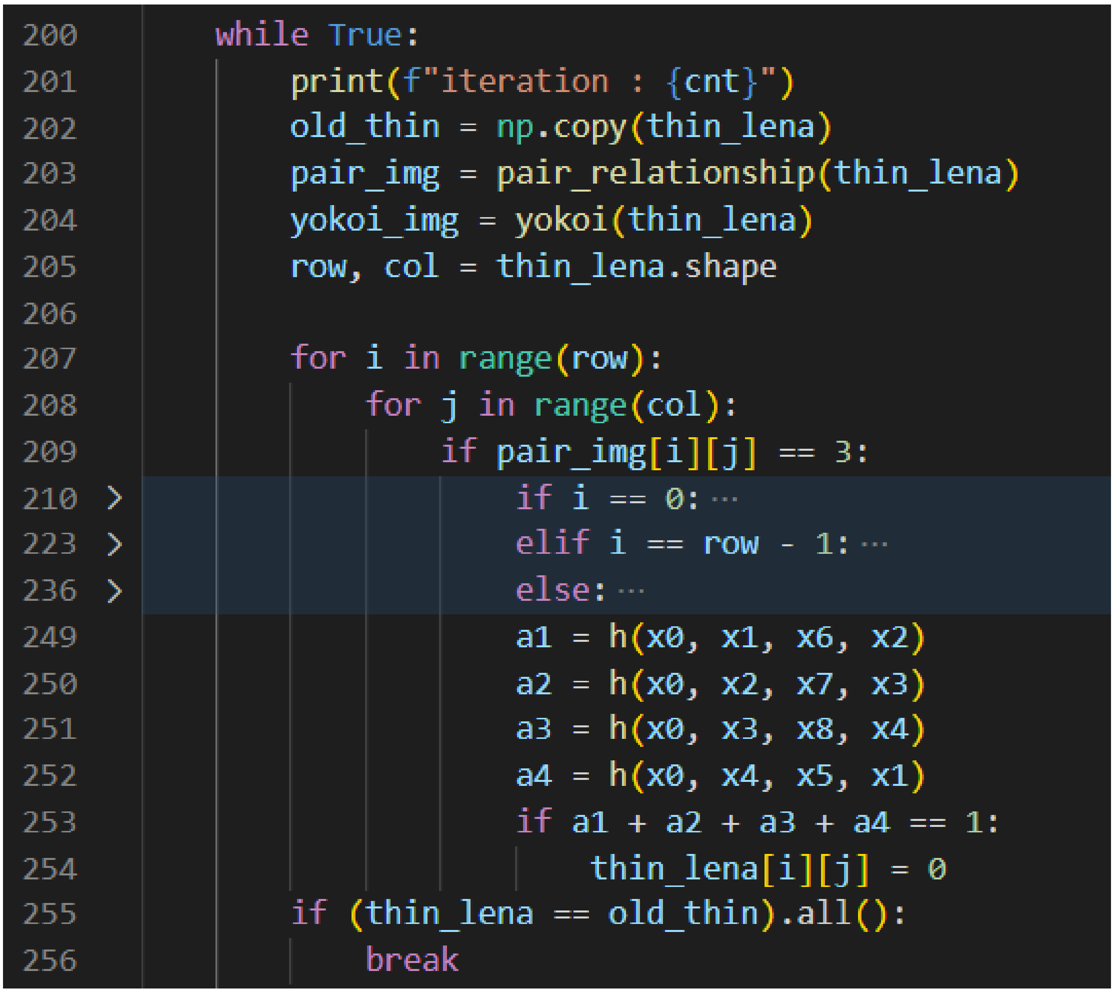
R10922123　周昱豪

Code:

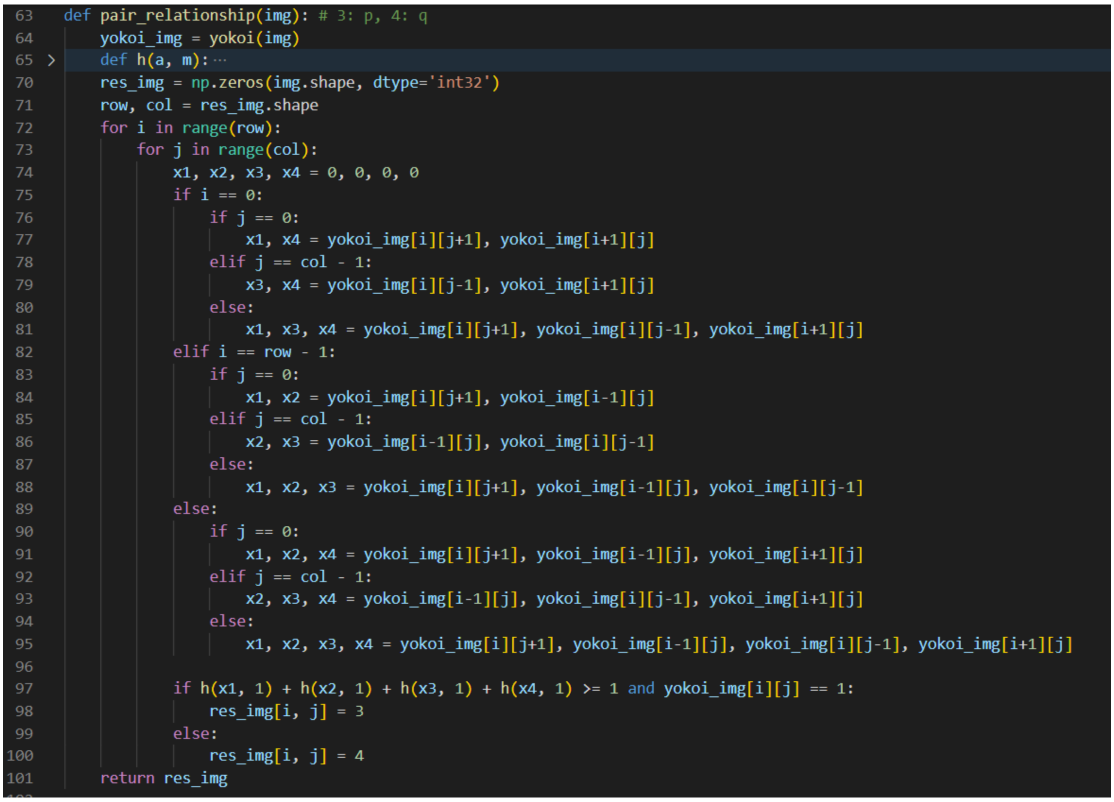
1. 首先先將原圖做 binarize 再做 downsample，那因為前幾份作業有做過在這邊就不附上 code 了。
2. 整體大架構為下圖



```
while True:
    print(f"iteration : {cnt}")
    old_thin = np.copy(thin_lena)
    pair_img = pair_relationship(thin_lena)
    yokoi_img = yokoi(thin_lena)
    row, col = thin_lena.shape

    for i in range(row):
        for j in range(col):
            if pair_img[i][j] == 3:
                if i == 0: ···
                elif i == row - 1: ···
                else: ···
                a1 = h(x0, x1, x6, x2)
                a2 = h(x0, x2, x7, x3)
                a3 = h(x0, x3, x8, x4)
                a4 = h(x0, x4, x5, x1)
                if a1 + a2 + a3 + a4 == 1:
                    thin_lena[i][j] = 0
    if (thin_lena == old_thin).all():
        break
```

3. 先將結果圖(thin_lena)利用 np.copy 給 old_thin，藉此來判斷兩個 iteration 是否有變化
4. 再做 pair_relationship，程式碼如下，主要就是先得到上次作業的 yokoi_img 後再依照上課 PPT 來實作出 pair_relationship operator



```python
def pair_relationship(img): # 3: p, 4: q
    yokoi_img = yokoi(img)
    def h(a, m):···
    res_img = np.zeros(img.shape, dtype='int32')
    row, col = res_img.shape
    for i in range(row):
        for j in range(col):
            x1, x2, x3, x4 = 0, 0, 0, 0
            if i == 0:
                if j == 0:
                    x1, x4 = yokoi_img[i][j+1], yokoi_img[i+1][j]
                elif j == col - 1:
                    x3, x4 = yokoi_img[i][j-1], yokoi_img[i+1][j]
                else:
                    x1, x3, x4 = yokoi_img[i][j+1], yokoi_img[i][j-1], yokoi_img[i+1][j]
            elif i == row - 1:
                if j == 0:
                    x1, x2 = yokoi_img[i][j+1], yokoi_img[i-1][j]
                elif j == col - 1:
                    x2, x3 = yokoi_img[i-1][j], yokoi_img[i][j-1]
                else:
                    x1, x2, x3 = yokoi_img[i][j+1], yokoi_img[i-1][j], yokoi_img[i][j-1]
            else:
                if j == 0:
                    x1, x2, x4 = yokoi_img[i][j+1], yokoi_img[i-1][j], yokoi_img[i+1][j]
                elif j == col - 1:
                    x2, x3, x4 = yokoi_img[i-1][j], yokoi_img[i][j-1], yokoi_img[i+1][j]
                else:
                    x1, x2, x3, x4 = yokoi_img[i][j+1], yokoi_img[i-1][j], yokoi_img[i][j-1], yokoi_img[i+1][j]

            if h(x1, 1) + h(x2, 1) + h(x3, 1) + h(x4, 1) >= 1 and yokoi_img[i][j] == 1:
                res_img[i, j] = 3
            else:
                res_img[i, j] = 4
    return res_img
```

5. 之後開始檢查如果(row, col)這個位置的 pair_relationship 結果是 p 的話，再去計算 shrink operator，判斷說現在的點是否符合可以刪除的標準，在程式碼 249-254 行

Result: