

## HW7 Report

語言環境: python3 + PIL+numpy

執行: python3 hw7.py

檔案:

程式	----- hw7.py
圖檔	----- binarized.bmp
	----- output.bmp

說明:

hw7.py

```
9 def val(pos):
10     r,c = pos
11     return -1 if r<0 or c<0 or r>=R or c>=C else arr[r][c]
12
13 def h(a, b, c, d):
14     va, vb, vc, vd = val(a), val(b), val(c), val(d)
15     if va!=vb: return 's'
16     return 'r' if vb==vc and vc==vd else 'q'
17
18 def f(r, c):
19
20     ret = [h((r,c), (r,c+1), (r-1,c+1), (r-1,c)), \
21            h((r,c), (r-1,c), (r-1,c-1), (r,c-1)), \
22            h((r,c), (r,c-1), (r+1,c-1), (r+1,c)), \
23            h((r,c), (r+1,c), (r+1,c+1), (r,c+1))]
24     return 5 if (ret.count('r')==4) else ret.count('q')
25
26 def ispair(r,c,yokois):
27     if r>0 and yokois[r-1][c]==1: return True
28     if c>0 and yokois[r][c-1]==1: return True
29     if r<(R-1) and yokois[r+1][c]==1: return True
30     if c<(C-1) and yokois[r][c+1]==1: return True
31     return False
```

前三個函式用來找 yokoi number:

1. val 負責判斷出界並選擇回傳-1(代表出界)或二元資料陣列中對應的亮度值(1 或 0)
2. h 和 f 是照投影片的寫法，最後 f 以四次 h 的 q 數量作為 yokoi number 回傳

ispair 接受一個點座標與一個 RXC 的陣列 yokois(儲存整張圖所有點的 yokoi number)，並判斷這個點(r,c)是否有 yokoi number 為 1 的 4-connected neighbor。

```

32 while True:
33     yokoiarr = np.array([[f(r,c) if arr[r][c]!=0 else -1 for c in range(C)] for r in range(R)])
34     pairarr = np.array([(r,c) for r in range(R) for c in range(C) if yokoiarr[r][c]==1 and ispair(r,c,yokoiarr)])
35     if len(pairarr)==0: break
36     for r,c in pairarr:
37         if(f(r,c)==1): arr[r][c] = 0
38 Image.fromarray(arr).save('output.bmp')

```

主程式由兩層迴圈組成，第一層迴圈開頭先找出所有點的 **yokoi number** 並存在 **yokoiarr** 裡，接下來再找出所有 **yokoi number** 是 1 而且也有 4-connected neighbor 它的 **yokoi number** 也是 1 的點，並將這些點的座標存入列表 **pairarr** 中。如果 **pairarr** 為空，則跳出迴圈輸出新圖檔。

接下來第二層迴圈從這些 **ispair** 的點當中找出 **yokoi number** 還是 1 的點來消除(因為有些點在改動完後 **yokoi number** 會變，所以要再判斷一次自己是不是 **yokoi number** 仍為 1)。

一直反覆操作直到沒有東西可消，輸出即為 **sekeleton** 的圖檔。