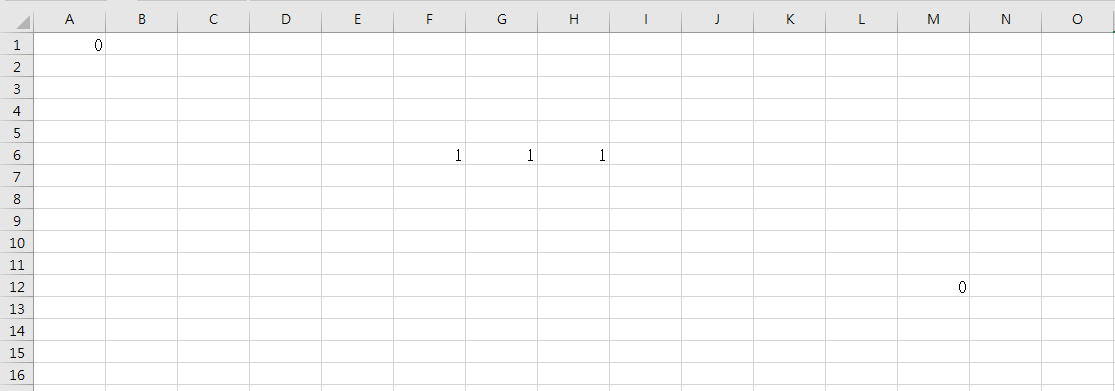
使用csv檔作為世界編排資訊，方便修改。規則如下：



1. 「字元1」被當作是生命（在此用1表示）
2. 「非1與空字元」為非生命/死亡細胞
3. 世界的邊界大小分別以「列數位置最大的0」與「行數位置最大的0」作為邊界（這裡是13x12）

程式碼使用python編寫：

程式碼的部分使用兩個檔案，一個為處理csv的檔案：

import csv #導入csv模組

def gotWorld():

    world = []

    with open("worldData.csv",encoding="UTF-8-sig") as csvFile:

        #建立串列儲存世界資訊之資料，判斷非1字元，改為0，1則不更改

        csvReader = csv.reader(csvFile)

        listReport = list(csvReader)

        for index in listReport:

            for j in range(len(index)):

                if index[j] != "1":

                    index[j] = "0"

            world.append( list(map(int,index)) )

    #回傳世界、寬度、長度資料

    width = len(world[0])

    length = len(world)

    return world,width,length

另一個為處理world與視覺化的檔案：

import numpy as np

import matplotlib.pyplot as plt

import matplotlib.animation as ani

from csvReader import gotWorld

#在此做了兩個模式，輸入1是隨機生成，輸入2是使用CSV檔生成世界

mode = int(input("Please input your mode : "))

if mode == 1:

    WIDTH = int(input("Please input your row and col : "))

LENGTH = WIDTH

    world = []

    for i in range(LENGTH):

        world.append( [random.randint(0,1) for j in range(WIDTH)] )

elif mode == 2 :

    world,WIDTH,LENGTH = gotWorld()

day = 0 #第x世代

generationText = "Generation"

def next\_generation(): #下個世代

    for i in range(LENGTH):

        for j in range(WIDTH):

            #串列走訪

            life\_nearby = 0 #此變數用來表示附近生物數量

            if i != 0:

                if world[i-1][j] in [1,-1]:

                    life\_nearby += 1

            if i != LENGTH-1:

                if world[i+1][j] in [1,-1]:

                    life\_nearby += 1

            if j != WIDTH-1:

                if world[i][j+1] in [1,-1]:

                    life\_nearby += 1

            if j != 0:

                if world[i][j-1] in [1,-1]:

                    life\_nearby += 1

            if i != 0 and j != 0:

                if world[i-1][j-1] in [1,-1]:

                    life\_nearby += 1

            if i != 0 and j != WIDTH-1:

                if world[i-1][j+1] in [1,-1]:

                    life\_nearby += 1

            if i != LENGTH-1 and j != 0:

                if world[i+1][j-1] in [1,-1]:

                    life\_nearby += 1

            if i != LENGTH-1 and j != WIDTH-1:

                if world[i+1][j+1] in [1,-1]:

                    life\_nearby += 1

            #即將死亡改成「-1」 即將出生改成「2」

            if life\_nearby<2 and world[i][j]==1:

                world[i][j] = -1

            elif 2<=life\_nearby<=3 and world[i][j]==1:

                pass

            elif life\_nearby>3 and world[i][j]==1:

                world[i][j] = -1

            elif life\_nearby==3 and world[i][j]==0:

                world[i][j] = 2

def finish():          #世代結束，將-1改成0 將2改成1

    for i in range(LENGTH):

        for j in range(WIDTH):

            if world[i][j] == 2:

                world[i][j] = 1

            elif world[i][j] == -1:

                world[i][j] = 0

def animate(self): #matplotlib animation用法

    global day,world

    generationText = "Generation " + str(day) #generationText字串，用於顯示

    next\_generation()

    finish()

    plt.cla() #清除上次的圖像

    plt.title("Conway's Game of Life - NSYSU")

    plt.text(-2,2,generationText,ha="right",wrap=True) #顯示generationText

    world\_np = np.array(world) #創建np.array以利matplotlib使用

    plt.imshow(world\_np)

    print("Generation",day)

    day += 1

fig = plt.figure() #創建figure

world\_np = np.array(world) #創建np.array以利matplotlib使用

animator = ani.FuncAnimation(fig, animate, interval = 500) #matplotlib animation用法

plt.imshow(world\_np)

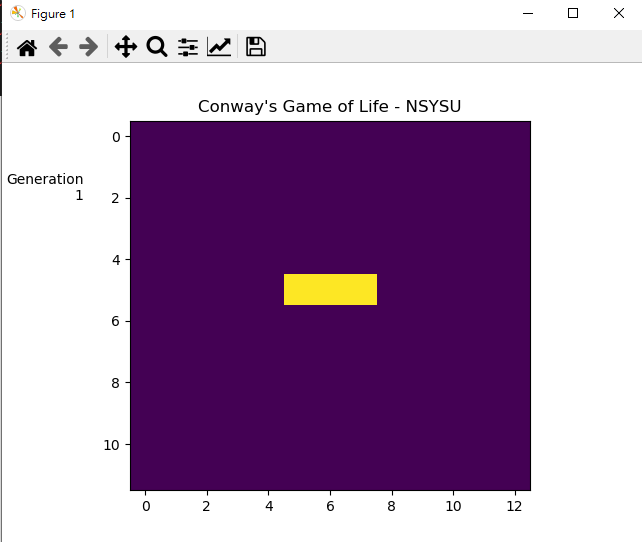
plt.title("Conway's Game of Life - NSYSU") #標題名稱

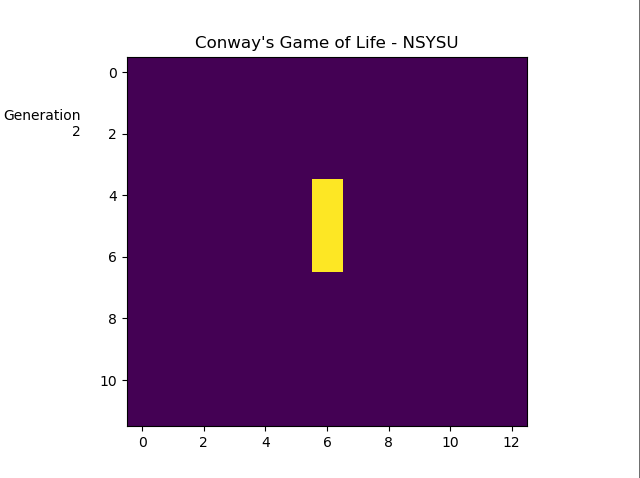
generationText = "Generation " + str(day) #generationText字串，用於顯示

plt.text(-2,2,generationText,ha="right",wrap=True) #顯示generationText

plt.show() #顯示圖表

成果如下：





以下是程式碼連結：

<https://reurl.cc/1ozLxX>

