

程式與科學計算 Python 期中考（佔原始分數 10%，10/17）

考試時間： 15:30–17:20

注意事項（請看完再開始作答！）

- 使用**系內**網路時，連線 ip 是 **140.112.66.23**；使用**系外**網路時，連線 ip 是 **140.112.66.200**
- 登入 study 主機，並在你的家目錄下建立子目錄 **mid_p**。
- 答題相關的所有檔案都請存放在你的 **~/mid_p/** 下；若檔案名稱、位置錯誤則不予計分。
- 考試**開放**參考上課講義、筆記、範例檔、作業、作業解答、程式相關書籍、網路搜尋。
- 所有答題程式碼的撰寫、編譯、執行必須由**本人獨立完成**。考試期間**不可與他人交談、傳遞訊息**（包括網路、手機）、**不可傳遞檔案**，或進行其他影響考試公平性的行為。
違反規定者直接取消應考資格。
- 遠距考試者，請連線到 gather，全程開啟視訊鏡頭與麥克風，有任何狀況請用文字訊息框聯繫
- 如果不確定自己的行為是否符合規定，請**事先**詢問監考老師或助教。
- 先完成考試作答者通知助教後，待助教將主機上的答題程式確認並備份完畢即可提早離場。
- 完成的程式，只要編譯後執行結果正確，就可獲得滿分。
如果程式碼排版整齊，或附有清楚的註解說明，會酌予加分，最多加該題分數的 2%

(a) Iteration Function System (16%)

有一個遞迴函數組如下，其中 A,B...F 為常數係數：

$$X(n+1) = A \times X(n) + B \times Y(n) + C$$

$$Y(n+1) = D \times X(n) + E \times Y(n) + F$$

(提示： $X(n+1)$ 要由 $X(n)$ 與 $Y(n)$ 共同決定， $Y(n+1)$ 也一樣)

請寫出 **mida.py**，進行下面的計算與輸出。

- 建立長度為 10 的兩個 **numpy** 陣列 X 與 Y。(2%)
- 接收使用者自鍵盤輸入兩個實數，分別為 $X[0]$ 與 $Y[0]$ 賦值。(4%)
- 然後利用迴圈依序計算出 $X[1] \sim X[9]$, $Y[1] \sim Y[9]$ ，遞迴公式中使用的係數如下：(5%)
 $A = 0.845$; $B = 0.035$; $C = 0.000$; $D = -0.030$; $E = 0.820$; $F = 1.600$;
- 計算完畢後將 n, X, Y 數列以格式化輸出到 **mida.txt** 如下：(5%)

(以下是輸入 $X[0]=0.1$, $Y[0]=0.2$ 的結果，各欄位之間有一個空格，n 欄位總寬度 3 格，X 欄位總寬度 5 格，Y 欄位總寬度 5 格)

#	n	X	Y
	0	0.100	0.200
	1	0.092	1.761
	2	0.139	3.041
	3	0.224	4.090
	4	0.332	4.947
	5	0.454	5.646
	6	0.581	6.216
	7	0.709	6.680
	8	0.833	7.056
	9	0.951	7.361

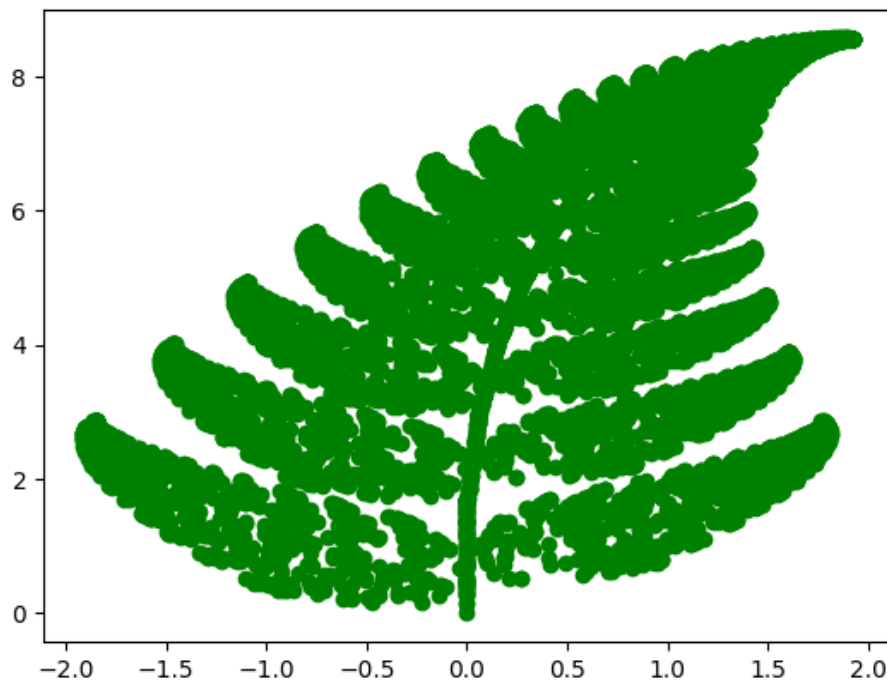
(b) Barnsley Fern (22%)

本題的概念是(a)小題的延續，請寫出 **midb.py**，依序下面步驟完成計算與繪圖：

- 設定一個常數 $nmax=10000$ 。以建立一個長度為 $(nmax-1)$ 的 numpy 陣列 P ，其元素為介於 $0\sim 1$ 之間的隨機數。（提示：利用 `np.random.rand(nmax-1)`指令）(4%)
- 建立長度為 $nmax$ 的 numpy 陣列 X 與 Y ，讓 $X[0]=0, Y[0]=0$ 。(2%)
- 以迴圈，進行 $X[1]\sim X[nmax-1], Y[1]\sim Y[nmax-1]$ 遞迴數列的運算。迴圈每前進一次，判斷式會以 $P(n)$ 的值決定要使用的 $A\sim F$ 係數值（參考下表），用(a)小題的公式計算出 $X(n+1), Y(n+1)$ 。(10%)

	$0.00 \leq P(n) < 0.01$	$0.01 \leq P(n) < 0.86$	$0.86 \leq P(n) < 0.93$	$0.93 \leq P(n) \leq 1.00$
A	0.000	0.845	0.200	-0.150
B	0.000	0.035	-0.260	0.240
C	0.000	0.000	0.000	0.000
D	0.000	-0.030	0.255	0.250
E	0.200	0.820	0.245	0.200
F	-0.012	1.600	0.290	0.680

- 迴圈結束後，用 `matplotlib.pyplot` 的 `plot` 畫出 X, Y 數列的散布圖如下， X 為橫軸， Y 為縱軸，使用綠點，座標軸範圍與刻度使用預設不用調整。輸出圖檔為 **midb.png**。(6%)
(由於陣列 P 的元素為隨機產生，每次畫出來的圖會有些微差異，不需要跟題目上的範例圖完全一模一樣，只要程式碼正確就會滿分)



說明：此遞迴函數組稱為 Barnsley Fern。數學家 Barnsley 發現看似複雜的蕨類葉片其實遵循了簡單的數學規則，利用相同形式的遞迴方程組，但是隨機代入不同的係數，就可以產生不同尺度下具有自我相似（self similarity）特性的碎形圖案，與真實的蕨葉非常接近。使用不同係數組合可以畫出不同種類的蕨葉。wikipedia “Barnsley Fern”有更詳細的介紹。

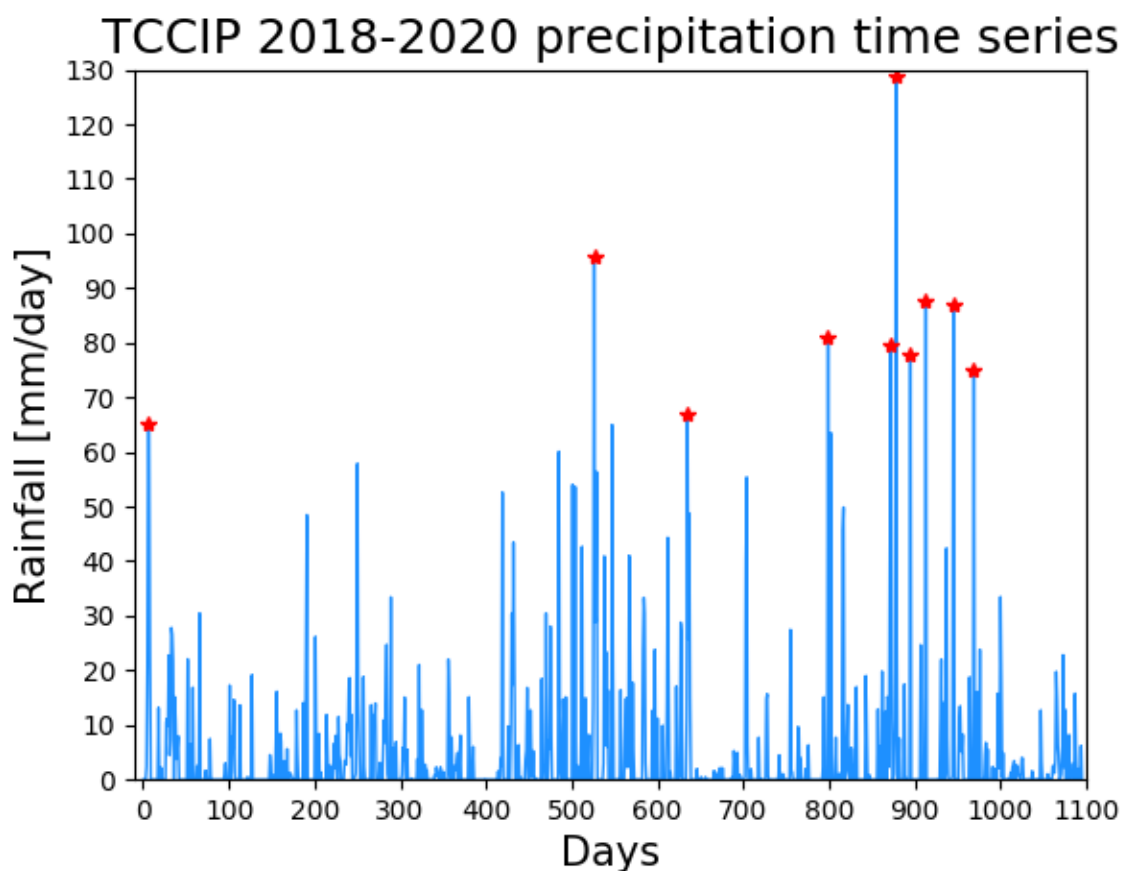
(c) Precipitation Intensity Spectrum (30%)

請複製 `/home/teachers/fortran_ta/data/PSC2022/mid_p/TCCIP_daily_precip_2018-2020.txt` 到你的 `~/mid_p/` 下。這個純文字檔是從 TCCIP 5 km 解析度網格化日降水資料中抽取出距離台大最近的網格，所得到 2018/01/01~2020/12/31 的逐日平均降水強度 (mm/day)，共 1096 天。

本題目標是對此日雨量時間序列進行統計，請寫出 `midc.py` 完成下面的運算與繪圖：

(c.1) 畫出時間序列及前十大資料 (12%)

- 讀取檔案中的降水時間序列到 numpy 陣列 `prec`
- 將 `prec` 隨時間的變化以淺藍線(dodgerblue)畫出，線條粗細 1 點 (`linewidth = 1`)。
- 利用 `sort` 函數將 `prec` 由大到小排序，用紅色*號在同一張圖中點出日雨量最大的前十筆。
- 加上 x 軸、y 軸標題（文字大小 15 點）、圖片標題（文字大小 18 點）；座標軸刻度、範圍如下圖（x 軸從 -10 開始）。輸出為 `midc1.png`



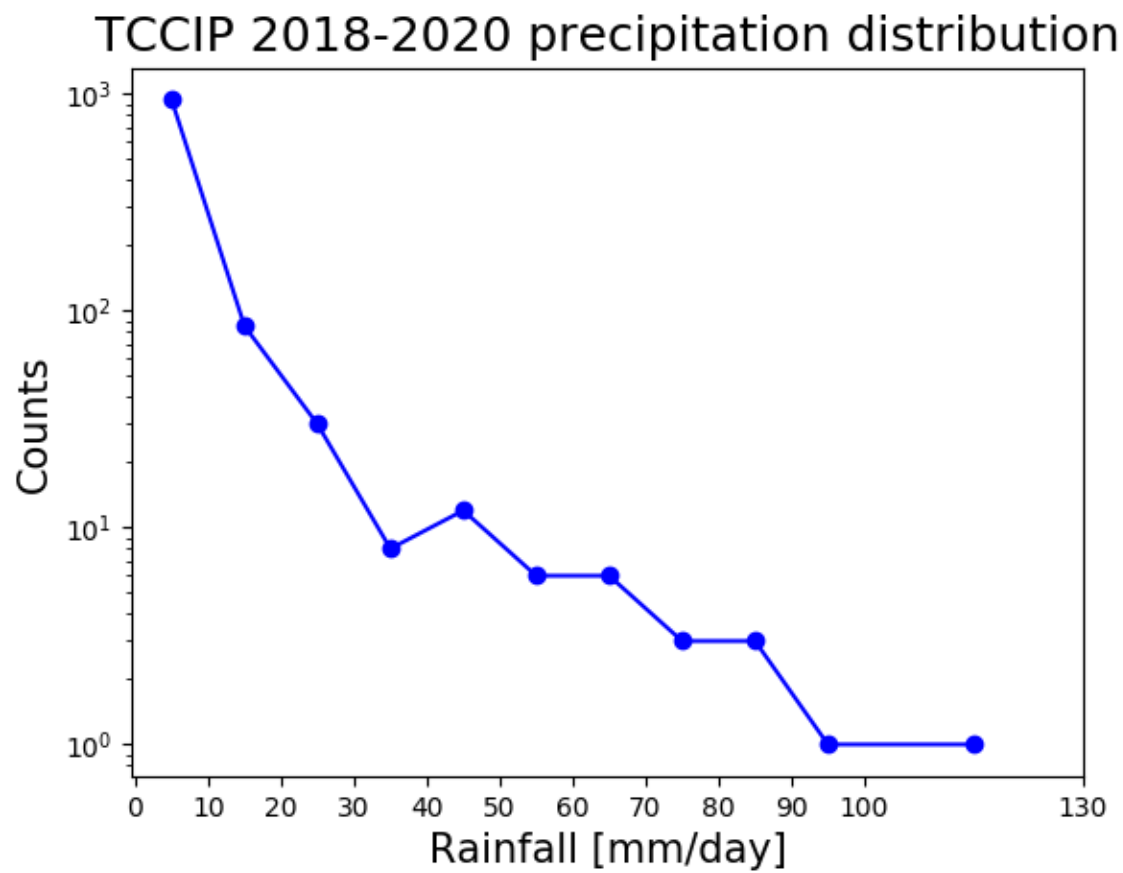
(c.2) 統計日雨量在不同強度區間發生的次數 (12%)

- 建立日降雨量區間 numpy 陣列 `bins = 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 130`
建立 numpy 陣列 `counts`，長度比 `bins` 陣列少一個元素，用來儲存發生在各區間的次數
- 利用迴圈與 numpy 陣列的 `mask` 語法，找出 `prec` 資料在各降雨區間的發生次數
($0 \leq \text{prec} < 10$ 的次數記錄於 `count[0]`，依此類推)

(c.3) 畫出日雨量強度譜 (6%)

- 將 `count` 隨不同降水強度的變化以藍色圓點及線條畫出，注意每個圓點在 x 軸的位置對應到 `bins` 陣列元素兩兩平均的值（第一個圓點在 5 mm/day，依此類推），y 軸為對數座標。

- 加上 x 軸、y 軸標題（文字大小 15 點）、圖片標題（文字大小 18 點）；座標軸刻度、範圍如下圖。線條粗細預設不用調整。輸出為 **midc2.png**



(d) Saturation Vapor Pressure (32%)

飽和水氣壓是位溫及氣壓的函數，其關係如以下公式：

$$e_s = 6.11 \times \exp \left[\frac{L_v}{R_v} \left(\frac{1}{273} - \frac{1}{T} \right) \right], \quad T = \theta \times \left(\frac{P}{P_0} \right)^{\frac{R_d}{C_p}}$$

e_s ：飽和水氣壓 (hPa) T ：溫度 (K) θ ：位溫 (K) P ：氣壓 (hPa)
 L_v ：水的蒸發潛熱 (2.5×10^6 J/kg) R_v ：水的氣體常數 (461.0 J/K kg)
 R_d ：乾空氣氣體常數 (287.0 J/K kg) C_p ：乾空氣定壓比熱 (1004.0 J/K kg)
 P_0 ：參考層氣壓 (1000.0 hPa)

本題目標為觀察改變 θ 和 P 對 e_s 的影響。請完成下面的運算與繪圖：

(d.1) 利用自訂函數計算飽和水氣壓 (12%)

- 建立程式檔案 **func.py**，將飽和水氣壓的計算方式寫成自訂函數
- 函數名稱設定為 **func_SATQ**，並依序接收 θ 和 P ，計算回傳 e_s
(可以選擇傳入純量、回傳純量並在主程式用迴圈計算；抑或是傳入陣列，在函數中用 meshgrid 計算並回傳整個二維陣列。只要能正確完成計算都給分)

(d.2) 主程式與計算 (10%)

- 創立主程式 **midd.py**。建立 θ 向量，由 275 遞增至 310，間距為 1。建立 P 向量，由 850 遞增至 1050，間距為 10。
- 用自訂函數計算各個 θ 和 P 組合對應的 e_s 結果。

(d.3) 繪製等值線(10%)

- 在主程式以 θ 為橫坐標， P 為縱座標，畫出 e_s 的等值線圖如下，輸出為 **midd.png**
- 色系設定為 **cm.hsv**，數值設定為 5 至 55，間隔為 5，等值線標籤字體大小 12 點
- 加上 x 軸標題、y 軸標題（注意下標，字串前加上 r 才能正確顯示）、圖片標題；刻度、範圍如下圖，字體大小與線條粗細為預設不用調整。

