Chapter 7

用Python學數學

本章大綱

☆ 7-1　內建數學公式
☆ 7-2　數學函數繪圖
☆ 7-3　實例探討

Python 內建很多數學工具，通通寫在 numpy 模組，本章將介紹這些功能，這樣可以幫助理解或解決國中、高中的數學問題。

7-1 內建數學公式

一元多次方程式

解一元多次方程式是使用 roots() 方法，可解一元一次、一元二次、一元三次等方程式。例如，若有一元一次方程式如下：

$x+2=0$

則求其解的程式如下：

```
import numpy as a
b=a.roots([1,2])#x+2=0
print(b)#-2
```

又例如，一元二次方程式如下：

x^2-4x-12=0

求其解的程式如下：

```
import numpy as a
b=a.roots([1,-4,-12])#x**2-4x-12=0
print(b)#6,-2
```

又例如，一元三次方程式如下：

x^3-3x^2-16x-12=0

求其解的程式如下：

```
import numpy as a
b=a.roots([1,-3,-16,-12])#x**3-3x**2-16x-12=0
print(b)#6,-2,-1
```

二元一次或三元一次

二元一次或三元一次要使用 linalg 類別的 solve 方法。例如，若有二元一次方程式如下：

3x+y=5

x-2y=-3

則求其解的 Python 程式如下：

```
import numpy as a
c=a.array([[3,1],[1,-2]])
d=a.array([5,-3])
ans=a.linalg.solve(c,d)
print(ans)#1,2
```

又例如，三元一次方程式如下：

$x+y-z=-2$

$x+z=2$

$x-y+2z=5$

則求其解的 Python 程式如下：

```
import numpy as a
c=a.array([[1,1,-1],[1,0,1],[1,-1,2]])
d=a.array([-2,2,5])
ans=a.linalg.solve(c,d)
print(ans)#1,-2,1
```

向量內積（高中數學）

若有兩個向量如下：

```
m=(a1,a2,a3)
n=(b1,b2,b3)
```

向量內積公式是

```
m*n=(a1*b1+a2*b2+a3*b3)
```

運算結果是純量。例如，若有向量

```
m=(1,1,-1)
n=(2,0,1)
```

則其內積程式如下：

```
import numpy as a
m=a.array([1,1,-1])
n=a.array([2,0,1])
d=a.inner(m,n)
print(d)#1
```

向量外積（高中數學）

向量外積也是高中空間向量常用計算，過程有點繁索，把程式寫好，那隨時可代入求答案。

```
m=(a1,a2,a3)
n=(b1,b2,b3)
```

則向量外積 $m*n$=($a2*b3$-$a3*b2$,$a3*b1$-$a1$-b3,$a1*b2$-$a2*b1$)。例如：

```
m=(1,1,-1)
n=(2,0,1)
```

則求解所需程式如下：

```
import numpy as a
m=a.array([1,1,-1])
n=a.array([2,0,1])
d=a.cross(m,n)
print(d)#1 -3 -2
```

矩陣相乘（高中數學）

若有矩陣 A_{23} 如下，我們稱爲 2 列 3 行、記爲 A_{23}，其中列與行共有 6 個元素，每個元素編號如下：

$$A_{23} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix}$$

矩陣要能進行相乘，必須第一個矩陣的行數等於第二個矩陣的列數，且相乘結果的列數會等於第一個矩陣的列數、行數會等於第二個矩陣的行數。如下式爲 A 矩陣乘以 B 矩陣的『列』『行』數目表示法。

$$A_{nm} \times B_{mp} = C_{np}$$

若有矩陣如下：

$$C_{np} = A_{nm} \times B_{mp}$$

則矩陣 C_{np} 的所有元素如下：

$$C_{np} = \sum_{k=1}^{m} A_{nk} \times B_{kp}$$

例如，若有矩陣 A_{23} 如下：

$$A_{23} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix}$$

矩陣 B_{32} 如下：

$$B_{32} = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{pmatrix}$$

則兩個矩陣相乘的結果表示如下：

$$A_{23} \times B_{32} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ & & \\ a_{21} & a_{22} & a_{23} \end{pmatrix} \times \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{pmatrix}$$

$$= \begin{pmatrix} a_{11} \times b_{11} + a_{12} \times b_{21} + a_{13} \times b_{31} & a_{11} \times b_{12} + a_{12} \times b_{22} + a_{13} \times b_{32} \\ a_{21} \times b_{11} + a_{22} \times b_{21} + a_{23} \times b_{31} & a_{21} \times b_{12} + a_{22} \times b_{22} + a_{23} \times b_{32} \end{pmatrix}$$

$$= \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} = C_{22}$$

例如：

$$\begin{pmatrix} -1 & 0 & 3 \\ 2 & 1 & -2 \end{pmatrix} \times \begin{pmatrix} 1 & 0 \\ 2 & -3 \\ -1 & 4 \end{pmatrix} = \begin{pmatrix} -4 & 12 \\ 6 & -11 \end{pmatrix}$$

計算過程如下：

$$C_{11} = a_{11} \times b_{11} + a_{12} \times b_{21} + a_{13} \times b_{31} = -1 \times 1 + 0 \times 2 + 3 \times -1 = -4$$
$$C_{12} = a_{11} \times b_{12} + a_{12} \times b_{22} + a_{13} \times b_{32} = 12$$
$$C_{21} = a_{21} \times b_{11} + a_{22} \times b_{21} + a_{23} \times b_{31} = 6$$
$$C_{22} = a_{21} \times b_{12} + a_{22} \times b_{22} + a_{23} \times b_{32} = -11$$

※範例7-1a

示範矩陣相乘。

✍資料結構

此題就要用二維 list 儲存，這樣再配合三層迴圈，則可以擴展任何列行的矩陣運算。

✍輸出結果

```
[[0, 0], [0, 0]]
 -1  0  3
  2  1 -2

  1  0
  2 -3
 -1  4

 -4 12
  6-11
```

✍程式列印

```
a=[[-1,0,3],[2,1,-2]]
b=[[1,0],[2,-3],[-1,4]]
c = [[0 for i in range(2)] for j in range(2)]
print(c)
n=2
m=3
p=2
for i in range(n):
    for j in range(p):
        for k in range(m):
            c[i][j]=c[i][j]+a[i][k]*b[k][j]
#print(a)
#print(b)
#print(c)
for i in range(n):
    for j in range(m):
        print('%3d'%a[i][j],end='')
    print()
print()
for i in range(m):
    for j in range(p):
        print('%3d'%b[i][j],end='')
    print()
print()
for i in range(n):
    for j in range(p):
        print('%3d'%c[i][j],end='')
    print()
```

其次，numpy 模組已經將以上演算內建爲『@』運算子，以上計算的 Python 程式如下：

```
import numpy as a
m=a.array([[-1,0,3],[2,1,-2]])
n=a.array([[1,0],[2,-3],[-1,4]])
p=m @ n
print(p)
```

集合的運算

關於集合的運算，請看 7-4 節。例如：

```
a={1,2,3}
b={2,3,5,6}
```

若要求其交集、聯集、差集，則其程式如下：

```
a={1,2,3}#表示法竟然和集合的列舉法相同
b={2,3,4,5}
c={1,2,3,4}
print(a& b)#交集 2,3
print(a|b)#聯集1,2,3,4,5
print(a-b)#差集1
print(a^b)#1,4,5
print(1 in a)#True
print(a==b)#False
print(5 not in b)#False
print(a < c)#包含於 True
print(a < b)#False
```

複數的運算

Python 的資料型態有複數，且運算子『+、-、*、/、abs』也適用複數的加減乘除的運算。請鍵入以下程式，並觀察執行結果。

```
a=4+1j#1不能省
b=2-1j
print(a+b)
print(a-b)
print(a*b)
print(a/b)
print(abs(a))
```

7-2 數學函數繪圖

國高中數學有很多數學函數，本節就使用 matplotlib.pyplot 繪圖類別來繪製這些函數圖形，尤其是高一數學的指數、對數、三角函數，您只要多畫幾次，那就會更瞭解其內涵。其次，matplotlib.pyplot 繪圖類別還要使用 numpy 模組，來存放所需要的連續數據。（補充說明：若使用 Anaconda3，那以上模組都是預設模組，通通不用再額外安裝）

直線

我們人類繪製直線是使用直尺，但電腦並沒有直尺，那電腦如何繪製直線呢？答案就是要先建立直線方程式，然後再根據直線方程式一點一點密集繪製，這些密集的點，看起來就是直線。例如，設有直線方程式如下：

$y=ax+b$

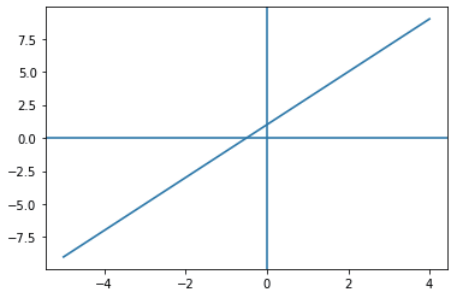
那要繪製 x 從 -5 到 5 的直線，程式如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-5, 5, 1) #-5開始到5，間隔是1，不含結束點5，
且間隔可以實數
print(x)
y = 2*x +1
print(y)
plt.plot(x,y)
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

執行結果如下，arange() 方法是產生一個 list（一群數字，不是一個，請看上一章），本例從 -5 開始到 5，間隔是 1，不含結束點 5，且間隔可以實數。請留意下圖第一、第二列數字是以下程式的輸出結果。

```
x = np.arange(-5, 5, 1) #-5開始到5，間隔是1，不含結束點5，且間隔可以實數
print(x)
y = 2*x +1
print(y)
```

```
[-5 -4 -3 -2 -1  0  1  2  3  4]
[-9 -7 -5 -3 -1  1  3  5  7  9]
```



✍自我練習

請將範例 6_5h 的迴歸直線使用 Python 繪圖。

二次曲線

前面 x 僅一次方 $y=ax+b$，那圖形是直線，若 x 是兩次 $y=ax^2+bx+c$、或三次 $y=ax^3+bx^2+cx+d$，那就形成曲線，x 二次會是拋物線，二次方係數爲正就凹向上，會有極小值；二次方係數爲負就凹向下，會有極大值，請觀察以下圖形。

$y=(x-3)^2+1$ （$x=3$ 有極小值 1）

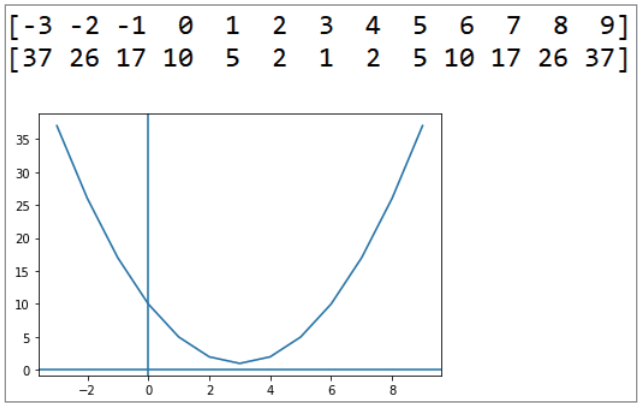
$=x^2-6x+10$

繪製以上圖形的程式如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-3, 10, 1) #數字是我多次修正的結果，這樣可以表現圖形
最有變化的部分
```

```
print(x)
y = x *x-6*x+10
print(y)
plt.plot(x,y)
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

以上程式執行結果如下：（x 從 -3 到 10，這是我慢慢觀察圖形，所調整出來的範圍，這樣才能畫出函數最有變化的範圍）



以下 x 是三次方，三次方圖形大部份有兩個臨界點，或稱相對極點。例如：

$$y=2x^3-3x^2-12x+3$$

將以上對 x 微分，得到斜率方程式如下：

$$dy/dx=6x^2-6x-12$$

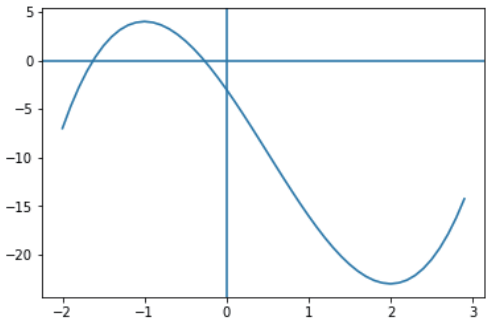
設其斜率爲零如下：

$$dy/dx=6x^2-6x-12=0$$

解出 x=-1 或 x=2 時斜率爲零，此點即爲臨界點，撰寫程式驗證如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-2, 3, 0.1)# 這是我慢慢觀察圖形，所調整出來的範圍
y=2*x**3-3*x**2-12*x-3
plt.plot(x,y)
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

執行結果如下：

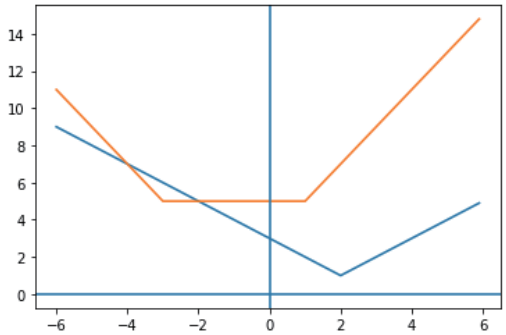


絕對值函數

方程式有一個絕對值，就會有一個轉折點，請看以下程式。

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-6, 6, 0.1)
y1=abs(x-2)+1#一個折點
y2=abs(x+3)+abs(x-1)+1#兩個折點
plt.plot(x,y1)
plt.plot(x,y2)
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

以上執行結果如下圖：



指數函數

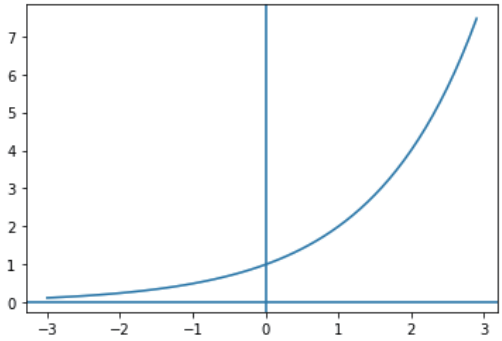
x 爲指數，例如：

$$y=a^x \quad a>0\ ,a \neq 1\ ,x \text{ 可爲任意實數}$$

稱爲指數函數，此爲高一數學。此一函數的第一個重點是，要討論定義域與值域，所有 x 值所成的集合稱爲定義域，其值可從負無限大到正無限大。第二個重點是，所有 y 值所成的集合稱爲值域，值域僅會大於 0，驗證程式如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-3, 3, 0.1)
y=2**x
plt.plot(x,y)
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

執行結果如下圖所示，請留意 x=0 時 y 一定等於 1。

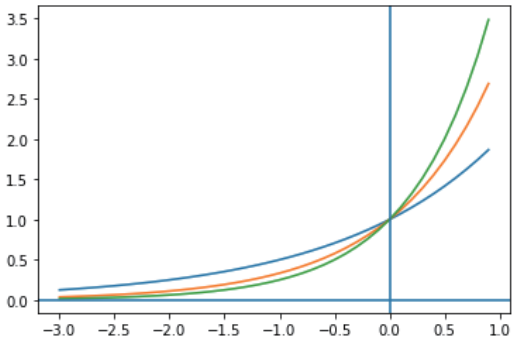


函數遞增與遞減

函數遞增與遞減是數學的考試重點，本節就是要用電腦繪圖，不斷繪圖，這樣就會是圖像式記憶法，這些圖形的特徵就會烙印腦海，考試很快可以寫出答案，把數學老師嚇昏。指數函數考試的重點在於 a>1 與 a<1，首先，先介紹 a>1 時，以下程式可比較 x>0 與 x<0，x>0，a 越大，當 x 值愈大時，y 值也愈大；x<0，a 越大，當 x 值愈大時，y 值反而愈小，驗證程式如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-3, 1, 0.1)
y1=2**x
y2=3**x
y3=4**x
plt.plot(x,y1)#藍
plt.plot(x,y2)#橘
plt.plot(x,y3)#綠
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

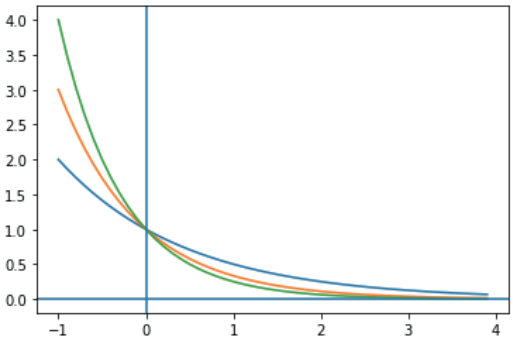
以上程式執行結果如下圖：（自己執行，就會有顏色）



當 a 小於 1 時，以下程式可比較 $x>0$ 與 $x<0$ 時，$y=(1/2)^x$, $y=(1/3)^x$, $y=(1/4)^x$ 的大小關係，此爲數學升學考試的重點，請自己將圖形映在腦海，那考試時，就很快可以寫出答案，這樣可以把數學老師嚇昏或冒冷汗，這就是圖像式記憶法。

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-1, 4, 0.1)
y1=(1/2)**x
y2=(1/3)**x
y3=(1/4)**x
plt.plot(x,y1)#藍
plt.plot(x,y2)#橘
plt.plot(x,y3)#綠
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

以上程式執行結果如下：（自己執行，就會有色彩）



✍自我練習

1. 請繪圖，並觀察 $y=2^x$ 與 $y=(1/4)*x^2$ 的交點個數。
2. 請繪圖，並觀察 $y=2^x$ 與 $y=x-3$ 的交點個數。
3. 請繪圖，並觀察 $y=2^x$ 與 $y=x+3$ 的交點個數。

對數

指數函數的反函數，就稱爲對數函數（什麼是反函數，例如，由座號得到姓名若是一個函數，那由姓名得到座號，就稱爲反函數）。例如：

$$y=log_a x$$

以上式子與

$$a^y=x$$

相同，x 稱爲眞數，a 稱爲底，a 僅能大於 0，且不能等於 1（請回想前面 1 的任何次方都是 1，所以對數的底就不能 1 了，因爲底數 =1 的對數值不唯一

$$log_1 1=1$$
$$log_1 1=2$$
$$log_1 1=100$$

其次，因爲指數函數的值域大於 0，現在反過來看，那就變成眞數 x 的定義域僅能大於 0。以下是 log 函數的 Python 表示法，請先鍵入，並觀察執行結果。

```
import numpy as np
import math
print (np.log(math.e))#base is e
print(np.log10(10))#base is 10
print(np.log2(2))#base is 2
```

其次，Python 對數函數的底僅有 10、e 與 2，若要其他數字當底數，則要使用換底公式。換底公式如下：

$log_a b = logb/loga$

例如，要得到 $log_3 9$，則程式如下：

```
print(np.log(9)/np.log(3))#  2
```

根據以上定義，若有對數函數如下

$y = log_2 x$

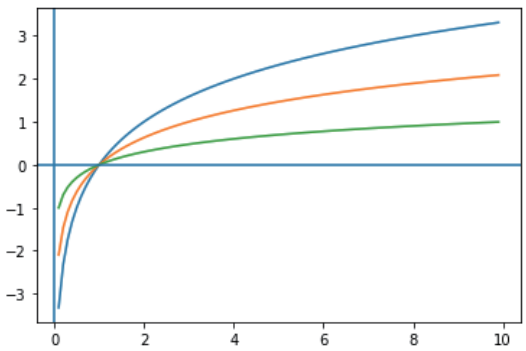
$y = log_3 x$

$y = log_{10} x$

則其繪圖程式如下，請留意 x 要大於 0

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(0.1, 10, 0.1)#不能從0開始
y1=np.log2(x)
y2=np.log(x)/np.log(3)#換底
y3=np.log10(x)
plt.plot(x,y1)#藍
plt.plot(x,y2)#橘
plt.plot(x,y3)#綠
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

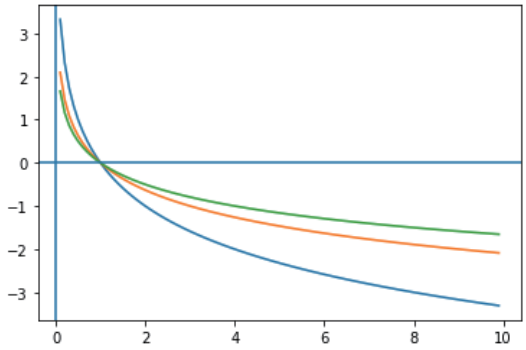
以上程式執行結果如下，圖形一定經過 (1,0) 。



上圖是 $a>1$ 的圖形，請留意 $x>1$ 與 $x<1$ 時三個函數的大小，這就要記圖形的輸出結果，數學考試時才能很快寫出答案。以下程式可繪出 $y=log_{1/2}x$、$y=log_{1/3}x$、$y=log_{1/4}x$，這些都是 $a<1$ 的圖形。

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(0.1, 10, 0.1)#不能從0開始
y1=np.log(x)/np.log(1/2)
y2=np.log(x)/np.log(1/3)
y3=np.log(x)/np.log(1/4)
plt.plot(x,y1)#藍
plt.plot(x,y2)#橘
plt.plot(x,y3)#綠
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

以上程式執行結果如下，圖形一定經過 (1,0) 。數學考試也是常考 $x>1$ 與 $x<1$ 時，三者的大小關係，也是記下以下執行結果，那考試很快就可寫出答案。

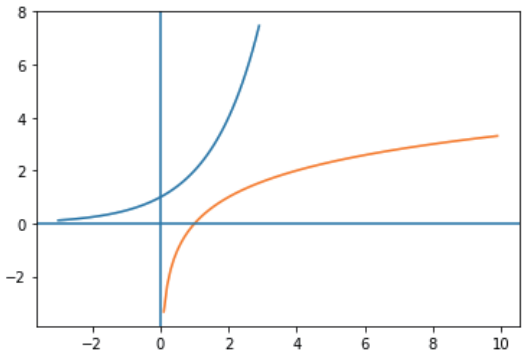


指數與對數的對稱驗證

以下程式可繪出 $y=2^x$（藍色）與 $y=log_2x$（橘色）的圖形，他們對稱 x-y=0。

```
import matplotlib.pyplot as plt
import numpy as np
x1 = np.arange(-3, 3, 0.1)
y1=2**x1
plt.plot(x1,y1)#藍
x2 = np.arange(0.1, 10, 0.1)#不能從0開始
y2=np.log2(x2)
plt.plot(x2,y2)#橘
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

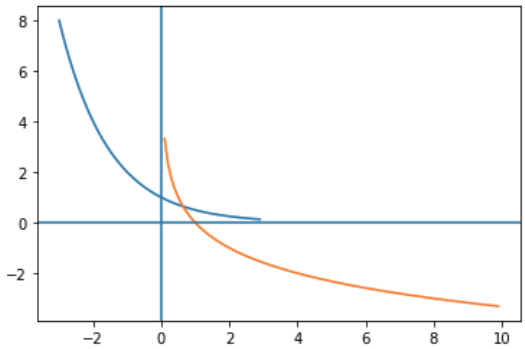
以上程式執行結果如下



以下程式可繪出 $y=(1/2)^x$（藍色）與 $y=log_{1/2}x$（橘色）的圖形，他們對稱 x-y=0。

```
import matplotlib.pyplot as plt
import numpy as np
x1 = np.arange(-3, 3, 0.1)
y1=(1/2)**x1
plt.plot(x1,y1)#藍
x2 = np.arange(0.1, 10, 0.1)#不能從0開始
y2=np.log(x2)/np.log(1/2)
plt.plot(x2,y2)#橘
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

以上程式執行結果如下。



✍自我練習

1. 請寫程式，繪出 $y=log_2x$ 與 $y=x$ 的圖形，觀察有幾個交點。
2. 請寫程式，繪出 $y=log_2x$ 與 $y=x^2$ 的圖形，觀察有幾個交點。
3. 方程式 $|log_2x|=2-x$ 有幾個實數解？（提示：同時繪出 $y=|log_2x|$ 與 $y=2-x$，再用肉眼觀察）

電容充放電函數

前面第 4 章已經說明 $e=2.718$ 的由來，因爲電容的充放電、生物界細菌的繁衍竟然都要用 e 才能解釋其現象，也因爲電容充放電和細菌繁殖都是相同的函數可以解釋，所以我們才會大膽假設電子的數量也是 10^{18} 等級的數量。電容放電函數如下，這是高職一年級基本電學實習的內容，使用一個電容器，讓其充電與放電，並用三用電表量電壓，就可驗證其內容。

$$y=e^{-x}$$

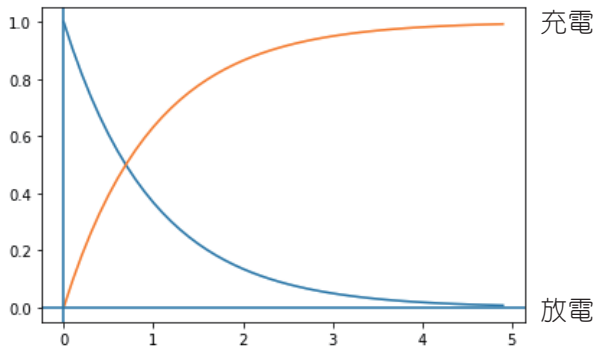
電容充電函數是

$$y=1-e^{-x}$$

所以撰寫程式如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(0, 5, 0.1)
y1=(np.e)**(-x)#放電
y2=1-(np.e)**(-x)#充電
plt.plot(x,y1)#藍
plt.plot(x,y2)#橘
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

以上程式執行結果如下。



三角函數

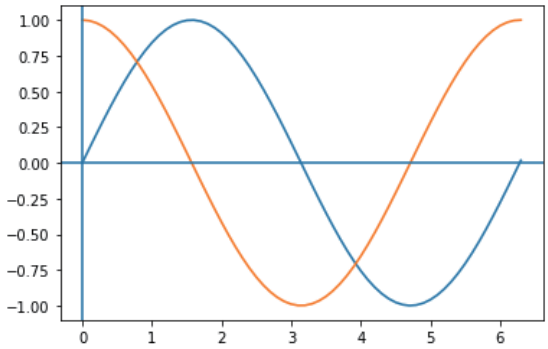
任意三角形有三個邊，三個角，科學家爲了找出其邊與角的關係，就定義出三角函數。例如已知三角形任兩邊與其所夾角度，就可透過餘弦定理找出第 3 邊的長度。所以現代人蓋六角庭，蓋橋樑、蓋房子，可以將所有木頭或鋼板長度通通事先裁切好，到了現場就是組裝而已。但是古代的工匠就窘了，他們沒有三角函數，所以是一面作，一面量下一根木頭長度，一面裁切，那速度當然慢。以下是 sin() 函數的定義，變數 x 長在 sin() 函數內

$$y=\sin(x)$$

x 定義域是 0 到 2π，值域在正負 1 之間，所以撰寫程式如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(0, 2*np.pi+0.1, 0.1)
y1 =np.sin(x)
y2=np.cos(x)
plt.plot(x,y1)#藍色
plt.plot(x,y2) #橘色
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

執行結果如下。

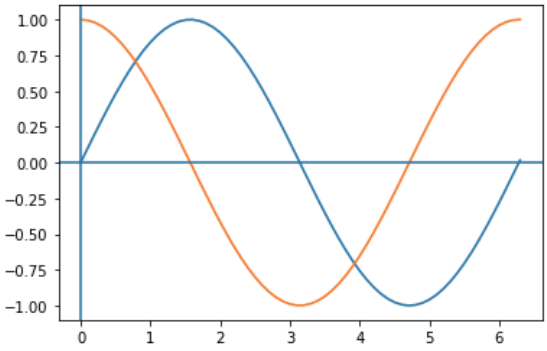


sin(x)與cos(x)

既生瑜何生亮，既然有 sin(x)，那爲什麼要有 cos(x)，那是因爲在物理上，我們有此需求，我們可用 cos(x) 表示其相位的變化。因爲上圖，我們將 sin(x) 圖形水平左移 π /2 單位，就是 cos(x)，也就是 cos(x) 其實就是 sin(x) 的相位往前推 90 度，cos(x)= sin(x+90)。例如，電學中交流電路的純電感電路，輸入信號電流是 sin(x)，我們發現電感端電壓信號會超前電流 90 度，這一現象如何用數學方程式表示呢？那就可用 cos(x) 來表示。這一問題，請在電學中再複習與反覆琢磨。

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(0, 2*np.pi+0.1, 0.1)
y1 =np.sin(x)
y2=np.sin(x+np.pi/2)
plt.plot(x,y1)#藍色
plt.plot(x,y2) #橘色
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

執行結果如下。

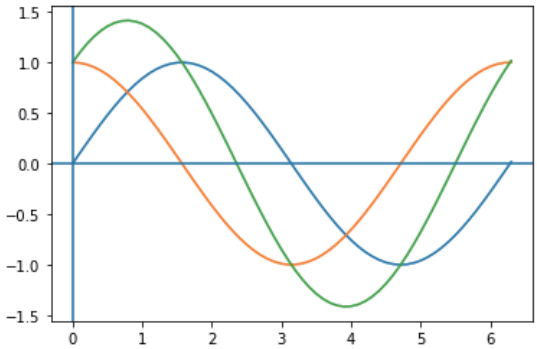


函數相加圖形

以下程式可繪出兩個函數的相加。

```
x = np.arange(0, 2*np.pi+0.1, 0.1)
y1 =np.sin(x)
y2=np.cos(x)
y3=np.sin(x)+np.cos(x)
plt.plot(x,y1)#藍色
plt.plot(x,y2) #橘色
plt.plot(x,y3)
```

執行結果如下：

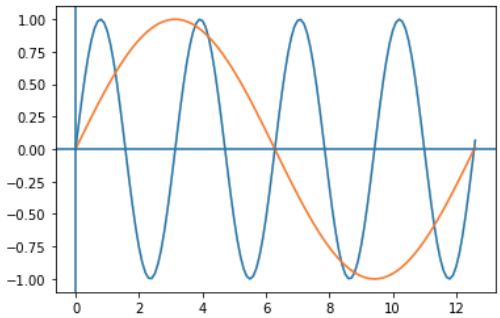


sin(2x)與sin(x/2)週期變化

sin(x) 週期是 2 π ，那 sin(2x)、sin(x/2) 週期爲何呢？這也是多繪幾次函數圖，就很容易記起來。繪製 sin(2x) 與 sin(x/2) 函數圖的程式如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(0, 4*np.pi+0.1, 0.1)
y1 =np.sin(2*x)
y2=np.sin(x/2)
plt.plot(x,y1)#藍色
plt.plot(x,y2)#橘色
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

執行結果如下：



由上圖可知，sin(2x) 函數週期變一半，sin(x/2) 函數週期變兩倍，這些問題，都是數學老師常考的題目。

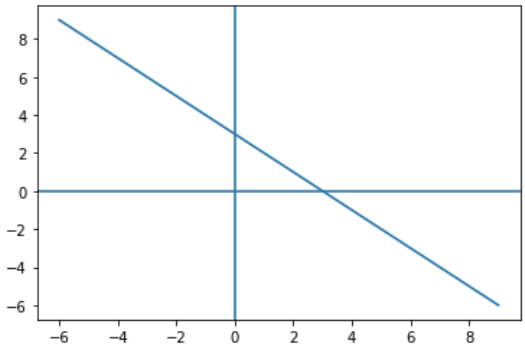
✍自我練習

1. 請自己練習繪出另 4 個三角函數圖形。
2. 請圖示 y=sin(x)，y=x+1，並觀察有幾個交點。

暴力法繪圖

所有實數，所有 $P(x,y)$ 所成的圖形 , 滿足 $x+y$-3=0，將會是一直線，請寫程式證明。

✍執行結果



✍程式列印

數學老師說『所有實數，所有 P(x,y) 所成的圖形』這句話，眞有如文言文一樣簡潔，程式設計就要使用『雙迴圈』來表示，所以程式如下：

```
import matplotlib.pyplot as plt
x=[]
y=[]
for i in range(-10,10,1):
    for j in range(-10,10,1):
        if i+j-3==0:
```

```
            x.append(i)
            y.append(j)
plt.plot(x,y)#藍
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

✍自我練習

1. 所有實數，所有$P(x,y)$所成的集合，F_1是(3,0)，F_2是(-3,0)，且$PF_1+PF_2=10$，將會是橢圓，上式以方程式表示爲$\sqrt{(x-3)^2+y^2}+\sqrt{(x+3)^2+y^2}=10$，請寫程式以暴力法繪出此圖形。
2. 所有實數，所有$P(x,y)$所成的集合，F_1是(3,0)，F_2是(-3,0)，且$|PF_1-PF_2|=4$，將會是雙曲線，請寫程式證明。

圓

人類必須靠圓規畫圓，但電腦沒有圓規，那電腦如何畫圓，此與直線一樣，也是要先造方程式，再依照方程式一點一點密集的點上去，看起來就是圓，圓的方程式有兩種，一種是直角座標。例如：

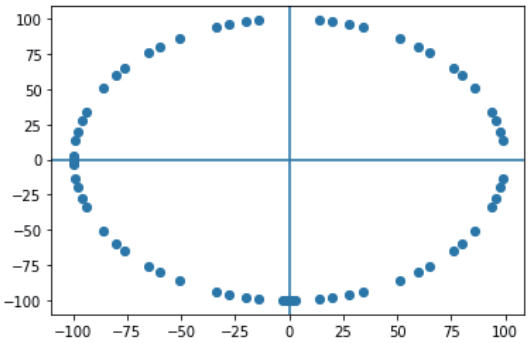
$$x^2+y^2=100$$

以上方程式有兩個變數，可以用以上暴力法繪製，程式如下：（若使用 if abs(i**2+j**2-10000)==0 則合乎條件的點太少了，請自己練習）

```
import matplotlib.pyplot as plt
x=[]
y=[]
for i in range(-100,100,1):
    for j in range(-100,100,1):
        if abs(i**2+j**2-10000)<10:
            x.append(i)
```

```
            y.append(j)
plt.scatter(x,y)#藍
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

以上程式執行結果如下。



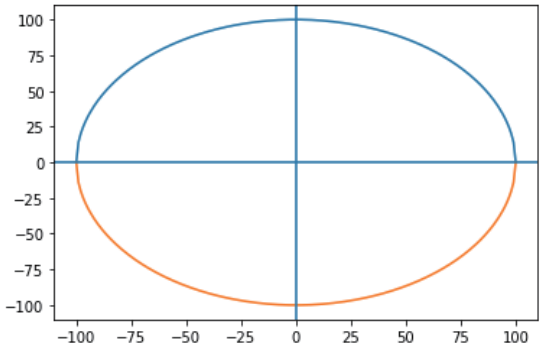
其次，雖然有兩個變數，但是，因爲兩個變數有關係，所以可以簡化變數如下：

$$y = \pm\sqrt{10000 - x^2}$$

根據以上方程式，程式撰寫如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-100, 100+1, 1)
y1 = (10000-x*x)**(1/2)
y2 = -(10000-x*x)**(1/2)
plt.plot(x,y1)#藍色
plt.plot(x,y2)#橘色
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

執行結果如下：



參數式

圓、橢圓、雙曲線都還有參數式，這通通可用來畫出圖形。例如，圓 $x^2+y^2=1$ 的參數式如下：

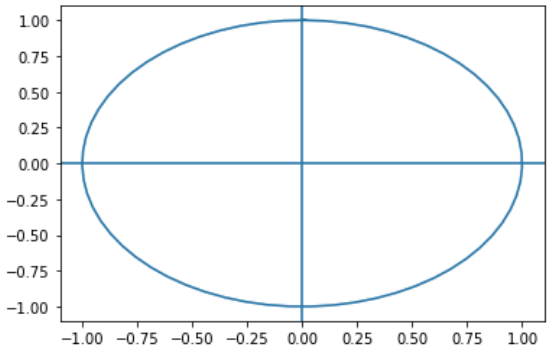
$x=\cos(x)$

$y=\sin(x)\quad 0<x\leq=2\pi$

利用以上參數式畫圖，程式如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(0, 2*np.pi+0.1, 0.1)
y1 =np.sin(x)
y2 =np.cos(x)
plt.plot(y1,y2)#藍色
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

執行結果如下：



✍自我練習

1. 假設橢圓的參數式如下，請寫一程式繪出圖形。

   $x=2\cos(x)$

   $y=3\sin(x)$

2. 假設雙曲線參數式如下，請寫一程式繪出圖形。

   $x=\text{atan}(x)$

   $y=\text{bsec}(x)\quad 0<x\leq=2\pi\quad x\neq\frac{\pi}{2},\frac{3}{2}\pi$

線性規劃

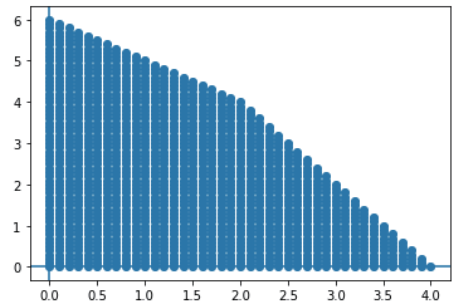
請繪出受制於 $x>=0,y>=0,x+y<=6,2x+y<=8$ 條件下，並求出 $f(x,y)=-2x+3y$ 的極大值。

✍程式列印

```
import matplotlib.pyplot as plt
x=[]
y=[]
s=10
max=-10000
for i in range(10*s):
    for j in range(10*s):
        if ((i/s+j/s)<=6) and ((2*i/s+j/s)<=8) :
            x.append(i/s)
            y.append(j/s)
```

```
            f=-2*i/s+3*j/s
            if f>max:
                max=f
plt.scatter(x,y)#藍色
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
print(max)
```

執行結果如下：




```
18.0
```

✍自我練習

1. 受制於 $x>=0,y>=0,x+y<=3,2x+y<=4$ 的條件下，請繪出滿足條件的區域，並求 $f(x,y)=x+3y$ 的最大值。(答案是 9)
2. 請寫程式圖示 x,y 爲正實數，且滿足 $x+y=4$ ，則繪出 $f(x,y)=1/x+1/y$ 的圖形，其最小值爲何？（答案是 1）
3. 請寫程式圖示 $|x-1|+|y-3|<=2$ 的區域，並求 $f(x,y)=x+2y$ 的最大值。(答案是 11)
4. 在座標平面上，設 S,T 爲不等式 $|x|+3|y|<=7$ 的圖形上任意兩點，請圖示其範圍，且求 S,T 距離最大值爲何？
5. 請寫一程式，圖示 $|x+y|<=8$,$|x-y|<=8$ 的區域。
6. 請寫程式，圖示不等式 $2|x+1|<=|x-2|$ 的範圍。
7. 請寫程式，圖示不等式 $1<=|2x-1|<3$ 的範圍。

7-3 實例探討（高中第四冊空間向量）

空間中直線交點、距離的電腦化計算

專題學生：陳蘊慈
高雄市新莊高中

摘要

將『檢驗空間中的兩條直線相交或平行或歪斜，若相交計算其交點、歪斜計算其最短距離』的過程，以 Python 寫成程式，只要輸入兩條直線的參數，就可完成以上判斷與計算。

研究動機

空間的直線相交或歪斜的判別、交點座標、距離的計算都非常繁瑣，我們學了一年的程式設計，讓我想到若將這些計算，寫成程式，那就可快速解決問題，還可幫我驗算，檢驗一長串的計算是否正確。

專題原理

一、空間中兩條直線的交點演算法如下：

1. 空間中兩條直線的方程式如下：

$$L1=\frac{x+p1}{a1}=\frac{y+q1}{b1}=\frac{z+r1}{c1}$$

$$L2=\frac{x+p2}{a2}=\frac{y+q2}{b2}=\frac{z+r2}{c2}$$

令 $$L1=\frac{x+p1}{a1}=\frac{y+q1}{b1}=\frac{z+r1}{c1}=s$$

所以

$$x1=a1s-p1$$

$$y1=b1s-q1$$

$$z1=c1s-r1$$

令 $$L2=\frac{x+p2}{a2}=\frac{y+q2}{b2}=\frac{z+r2}{c2}=t$$

所以

$x2=a2t-p2$

$y2=b2t-q2$

$z2=c2t-r2$

2. 先假設能相交，$x1=x2, y1=y2$, 解出 s,t

$a1s-a2t=p1-p2$

$b1s-b2t=q1-q2$

爲了簡化計算，先令

$p=p1-p2$

$q=q1-q2$

3. 以上二元一次方程式，以公式法，解得 s,t 如下：

$$s=\frac{\begin{vmatrix} p & -a2 \\ q & -b2 \end{vmatrix}}{\begin{vmatrix} a1 & -a2 \\ b1 & -b2 \end{vmatrix}}=\frac{(-pb2+a2q)}{(-a1b2+a2b1)}$$

$$t=\frac{\begin{vmatrix} a1 & p \\ b1 & q \end{vmatrix}}{\begin{vmatrix} a1 & -a2 \\ b1 & -b2 \end{vmatrix}}=\frac{(a1q-b1p)}{(-a1b2+a2b1)}$$

4. 將上面的 s,t 代入 $z1,z2$

$z1=c1s-r1$

$z2=c2t-r2$

5. if ($z1==z2$):

   相交，輸出 ($x1,y1,z1$) 即爲交點。

   else:

   歪斜，計算最短距離。

二、空間中兩條直線歪斜時計算最短距離演算法如下：

1. 設 P 點在 $L1$，則 P 點座標（$a1s-p1,b1s-q1,c1s-r1$）
2. 設 Q 點在 $L2$，則 Q 點座標（$a2t-p2,b2t-q2,c2t-r2$）
3. 由 1 與 2 的假設，向量

   $PQ=(a2t-p2-a1s+p1,b2t-q2-b1s+q1,c2t-r2-c1s+r1)$

   整理一下

   $=(a2t-a1s+p1-p2,b2t-b1s+q1-q2,c2t-c1s+r1-r2)$

4. 向量 PQ 垂直 $L1$。$L1$ 的方向向量是（$a1,b1,c1$）所以 PQ*L1 ＝ 0（* 是內積）

   $a1a2t-a1^2s+a1(p1-p2)+b1b2t-b1^2s+b1(q1-q2)+c1c2t-c1^2s+c1(r1-r2)=0$

   整理

   $(a1a2+b1b2+c1c2)t-(a1^2+b1^2+c1^2)s=a1(p2-p1)+b1(q2-q1)+c1(r2-r1)$　　(1)

5. 向量 PQ 垂直 L2。L2 的方向向量是（$a2,b2,c2$）所以 $PQ*L2$ ＝ 0（* 是內積）

   $a2^2t-a2a1s+a2(p1-p2)+b2^2t-b2b1s+b2(q1-q2)+c2^2t-c2c1s+c2(r1-r2)=0$

整理

$(a2^2+b2^2+c2^2)t-(a2a1+b2b1+c2c1)s=a2(p2-p1)+b2(q2-q1)+c2(r2-r1)$ (2)

6. 解以上二元一次方程式。先令

$a3=a1a2+b1b2+c1c2$

$b3=-(a1^2+b1^2+c1^2)$

$c3=a1(p2-p1)+b1(q2-q1)+c1(r2-r1)$

$a4=a2^2+b2^2+c2^2$

$b4=-a3$

$c4=a2(p2-p1)+b2(q2-q1)+c2(r2-r1)$

7. 以上方程式簡化如下：

$a3t+b3s=c3$ (3)

$a4t+b4s=c4$ (4)

8. 由 (3)(4) 解二元一次方程式，可得 t,s 如下：

$$t=\frac{\begin{vmatrix} c3 & b3 \\ c4 & b4 \end{vmatrix}}{\begin{vmatrix} a3 & b3 \\ a4 & b4 \end{vmatrix}}=\frac{(c3b4-c4b3)}{(a3b4-a4b3)}$$

$$s=\frac{\begin{vmatrix} a3 & c3 \\ a4 & c4 \end{vmatrix}}{\begin{vmatrix} a3 & b3 \\ a4 & b4 \end{vmatrix}}=\frac{(a3c4-a4c3)}{(a3b4-a4b3)}$$

9. 由以上 s,t 求出 P,Q 點座標

P 點座標 $(xp,yp,zp)=(a1s-p1,b1s-q1,c1s-r1)$

Q 點座標 $(xq,yq,zq)=(a2t-p2,b2t-q2,c2t-r2)$

10. 公垂線段 $\overline{PQ}$ 長度即爲 L1,L2 最短距離。

$d=$ 線段 $\overline{PQ}=\sqrt{(xp-xq)^2+(yp-yq)^2+(zp-zq)^2}$

三、Python語法

1. 加法運算子如下：

   小型計算機有加、減、乘、除等鍵，這樣就可計算加、減、乘、除的結果，程式語言要能計算加減乘除，也要有這些算術運算子。下表是 Python 的算術運算子列表：

運算子	定義	優先順序	結合律
**	次方	1	由左至右
+/-	正負號	2	由右至左
*	乘法運算	3	由左至右
/	除法，得實數商	3	由左至右
//	除法，得整數商	3	由左至右
%	求餘數 (Modulus)	3	由左至右
+/-	加法 / 減法運算	4	由左至右
=	指派	14	由右至左

2. Python 決策敘述語法如下：

```
if運算式:
    敘述區塊1；
[else :
    敘述區塊2；]
```

四、根據以上說明，本例Python程式如下：

```
#歪斜線測資,d=3
p1=-3;q1=-1;r1=-2
a1=2;b1=2;c1=1
p2=-4;q2=4;r2=-1
a2=-2;b2=1;c2=2
#相交測資,交點是-2,4,-3
p1=-4;q1=-2;r1=-1
a1=3;b1=-1;c1=2
p2=-5;q2=-3;r2=2
```

```
a2=-7;b2=1;c2=-1
p=p1-p2
q=q1-q2
s=(-p*b2+a2*q)/(-a1*b2+a2*b1)
t=(a1*q-b1*p)/(-a1*b2+a2*b1)
x1=a1*s-p1
y1=b1*s-q1
z1=c1*s-r1
x2=a2*t-p2
y2=b2*t-q2
z2=c2*t-r2
print('%d,%d,%d'%(x1,y1,z1))
print('%d,%d,%d'%(x2,y2,z2))
if z1==z2 :
    print('焦點座標是%d,%d,%d'%(x1,y1,z1))
else:
    print('歪斜線')
    a3=a1*a2+b1*b2+c1*c2
    b3=-(a1*a1+b1*b1+c1*c1)
    c3=a1*(p2-p1)+b1*(q2-q1)+c1*(r2-r1)
    a4=a2*a2+b2*b2+c2*c2
    b4=-a3
    c4=a2*(p2-p1)+b2*(q2-q1)+c2*(r2-r1)
    t=(c3*b4-c4*b3)/(a3*b4-a4*b3)
    s=(a3*c4-a4*c3) /(a3*b4-a4*b3)
    xp=a1*s-p1
    yp=b1*s-q1
    zp=c1*s-r1
    xq=a2*t-p2
    yq=b2*t-q2
    zq=c2*t-r2
    d=((xp-xq)**2+(yp-yq)**2+(zp-zq)**2)**(1/2)
    print(d)
```

心得

程式設計眞的是一個學習數學的好工具，自己把所有推導，再用程式寫一次，眞的可以更加瞭解數學，感謝洪老師的指導，讓我數學與程式設計有很好的跨領域學習，更加強我往資訊科學研究的動力。其次，本例直線方程式係數暫用指派，待學習視窗程式設計，還可改善此輸出入模式。

✍自我練習

1. 平面中，產生直線的方法有兩點式、點斜式與斜截式，請寫一程式，可以讓使用者輸入已知條件，且產生一條直線。
2. 空間中，任意不共線的相異三點可決定一個平面，請寫一程式，可以給定不共線的相異三點，自動產生平面方程式。
3. 空間中，任意相異兩點可決定一直線，請寫一程式，給定相異兩點，即可產生此直線。

泉勝相關程式設計書

全民自造與程式設計

本書目錄如下，以下內容所用的電路都可用以上實驗板快速完成：（章節部分內容請看 www.goodbooks.com.tw）

第 1 章　導論
第 2 章　基本輸出入
第 3 章　變數與常數
第 4 章　運算子與運算式
第 5 章　決策流程指令
第 6 章　迴圈指令
第 7 章　陣列
第 8 章　模組化程式設計
　8-1　概念
　8-2　外部中斷
　8-3　字串處理（動態火車時刻表）
第 9 章　掃描輸出入元件
　9-1　四位數七段顯示器
　9-2　8*8 點陣 LED
　9-3　4*4 鍵盤（製作計算器）
第 10 章　進階輸出入裝置
　10-1　蜂鳴器
　10-2　超音波距離感測器
　10-3　LCD
　10-4　紅外線遙控器
　10-5　溫度感測器
第 11 章　綜合實例探討
第 12 章　資訊科技應用專題
　12-1　電子時鐘的製作
　12-2　字幕機的製作

全民程式設計 - 使用 Arduino（泉勝 PA001，有印刷空板）

本書目錄與內容同上，但書末有夾印刷空板。

中小學自造與程式設計-使用Arduino第一冊

本書目錄如下，以下內容所用的電路都可用以上實驗板快速完成（章節部分內容請看 www.goodbooks.com.tw）

第一單元　導論與程式的編譯與執行
第二單元　輸出入與序列埠監控視窗
第三單元　I/O 腳位探索
第四單元　資料的數位化
第五單元　霹靂燈
第六單元　霹靂燈與陣列
第七單元　七段顯示器
第八單元　擲骰子
第九單元　for 迴圈
第十單元　算術運算子與算術運算
第十一單元　決策指令與比較運算子
第十二單元　指撥開關與表決器
第十三單元　位元運算子與霹靂燈
第十四單元　按壓開關與計數器
第十五單元　搶答器與 while 迴圈
第十六單元　電子時鐘