**Python程式設計**

**期末考試題**

**銘傳大學電腦與通訊工程系**

|  |  |
| --- | --- |
| 班 級 | 電通四乙 |
| 姓 名 | 陳昱叡 |
| 學 號 | 04052474 |
| 成 績 | 試卷共 10 題，每題10分  共完成 10 題，應得 100 分 |
| 授課教師 | 陳慶逸 |

※ 將你的程式碼貼在指定的欄位裡，並且執行題目要求的輸入參數(為避免打錯，建議使用複製和貼上的功能來執行程式的參數)，再複製執行結果並貼在指定的欄位裡。

※ 需將執行結果擷圖貼上，以確保是程式執行結果，請確實依題目要求來完成。

※ 答完試題後，直接上傳word檔即可，不必轉成pdf。

※ 不得使用任何通訊軟體和手機，切記：千萬不要作弊! 千萬不要作弊! 千萬不要作弊! 否則不論首從均以零分計算。

1: 請以Python實現一個函式prod\_in\_list(list)，該函式能求得所傳入之串列(list)中，相鄰元素兩兩相加再除二(即相鄰兩點求平均值)所構成的串列，並且回傳。

例如:

|  |  |
| --- | --- |
| **輸入** | **執行結果** |
| **print(prod\_in\_list([27, 3, 4, 5, 7, 1]))** | **[15.0, 3.5, 4.5, 6.0, 4.0]** |
| **print(prod\_in\_list([2, 3, 14, 5, 7, 41]))** | **[2.5, 8.5, 9.5, 6.0, 24.0]** |

我的作答(包含程式碼、貼上結果的數值、將執行結果擷圖貼上)：

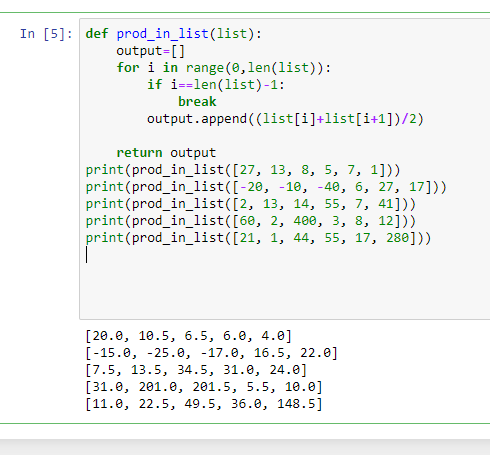
**請在下面欄位貼上程式碼：**

|  |
| --- |
| **def prod\_in\_list(list):**  **output=[]**  **for i in range(0,len(list)):**  **if i==len(list)-1:**  **break**  **output.append((list[i]+list[i+1])/2)**    **return output**  **print(prod\_in\_list([27, 13, 8, 5, 7, 1]))**  **print(prod\_in\_list([-20, -10, -40, 6, 27, 17]))**  **print(prod\_in\_list([2, 13, 14, 55, 7, 41]))**  **print(prod\_in\_list([60, 2, 400, 3, 8, 12]))**  **print(prod\_in\_list([21, 1, 44, 55, 17, 280]))** |

**請依下面要求輸入參數，並將執行結果的數值貼在指定欄位:**

|  |  |
| --- | --- |
| **print(prod\_in\_list([27, 13, 8, 5, 7, 1]))** | [20.0, 10.5, 6.5, 6.0, 4.0] |
| **print(prod\_in\_list([-20, -10, -40, 6, 27, 17]))** | [-15.0, -25.0, -17.0, 16.5, 22.0] |
| **print(prod\_in\_list([2, 13, 14, 55, 7, 41]))** | [7.5, 13.5, 34.5, 31.0, 24.0] |
| **print(prod\_in\_list([60, 2, 400, 3, 8, 12]))** | [31.0, 201.0, 201.5, 5.5, 10.0] |
| **print(prod\_in\_list([21, 1, 44, 55, 17, 280]))** | [11.0, 22.5, 49.5, 36.0, 148.5] |

**執行結果擷圖：**



2: 請以Python實現一個函式missing\_char(str, n)，函式輸入的參數str為非空的字串(non-empty string)，參數n為正整數。函式執行後會傳回一個由原字串str所構成的新字串，但新字串全部為大寫，而且其中索引n對應的字元會變成'$'。

注意：n的值得是原始字串str中字元的有效索引值（即n必須在0..len（str）-1的範圍之內）。

例如:

|  |  |
| --- | --- |
| **輸入** | **執行結果** |
| **print(missing\_char('kitten', 1))** | K$TTEN |
| **print(missing\_char('Kitten', 0))** | $ITTEN |
| **print(missing\_char('kitTen', 4))** | KITT$N |

我的作答：(包含程式碼、貼上結果的數值、將執行結果擷圖貼上)：

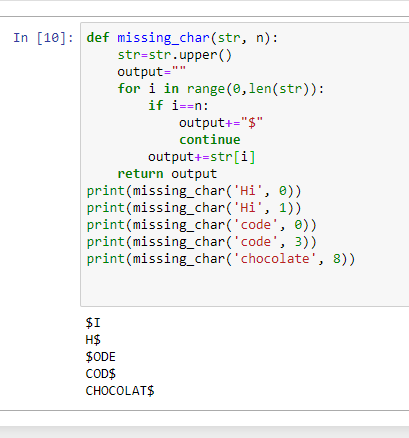
**請在下面欄位貼上程式碼：**

|  |
| --- |
| **def missing\_char(str, n):**  **str=str.upper()**  **output=""**  **for i in range(0,len(str)):**  **if i==n:**  **output+="$"**  **continue**  **output+=str[i]**  **return output**  **print(missing\_char('Hi', 0))**  **print(missing\_char('Hi', 1))**  **print(missing\_char('code', 0))**  **print(missing\_char('code', 3))**  **print(missing\_char('chocolate', 8))** |

**請依下面要求輸入參數，並將執行結果的數值貼在指定欄位:**

|  |  |
| --- | --- |
| **print(missing\_char('Hi', 0))** | $I |
| **print(missing\_char('Hi', 1))** | H$ |
| **print(missing\_char('code', 0))** | $ODE |
| **print(missing\_char('code', 3))** | COD$ |
| **print(missing\_char('chocolate', 8))** | CHOCOLAT$ |

**執行結果擷圖：**



3: 試實現一個python函式add\_string(str1)，當使用者輸入一個英文句子的字串，該程式能將句子的每一個單字以空格切開，轉成一個串列(list)。並依序在字串中的每個元素最末端加上'ed'；但是當元素的最末兩個字元已經是'ed'時，則在'ed'後面加上'ly'。

例如:

|  |  |
| --- | --- |
| **輸入** | **執行結果** |
| add\_string('try aborted') | ['tryed', 'abortedly'] |
| add\_string('listed around') | ['listedly', 'arounded'] |

我的作答：(包含程式碼、貼上結果的數值、將執行結果擷圖貼上)：

**請在下面欄位貼上程式碼：**

|  |
| --- |
| **def add\_string(str1):**  **output=[]**  **str1=str1.split(" ")**  **for string in str1:**  **if string[-2:]=="ed":**  **string+="ly"**  **output.append(string)**  **else:**  **string+="ed"**  **output.append(string)**  **return output**  **print(add\_string('string read'))**  **print(add\_string('abced goodly'))**  **print(add\_string('elected circuit'))**  **print(add\_string('bring book'))**  **print(add\_string('trying work'))** |

**請依下面要求輸入參數，並將執行結果的數值貼在指定欄位:**

|  |  |
| --- | --- |
| **print(add\_string('string read'))** | ['stringed', 'readed'] |
| **print(add\_string('abced goodly'))** | ['abcedly', 'goodlyed'] |
| **print(add\_string('elected circuit'))** | ['electedly', 'circuited'] |
| **print(add\_string('bring book'))** | ['bringed', 'booked'] |
| **print(add\_string('trying work'))** | ['tryinged', 'worked'] |

**執行結果擷圖：**



4: 試寫一個名稱為generateDict(n)的函式，它會根據我們所給予的*n*值，回傳一個內容為{1:cos(1/2), 2:cos(2/2), … , n:cos(*n/*2)}的字典(dictionary)。

例如:

輸入: generateDict(3)

輸出:

{1: 0.8775825618903728, 2: 0.5403023058681398, 3: 0.0707372016677029}

我的作答：(包含程式碼、貼上結果的數值、將執行結果擷圖貼上)：

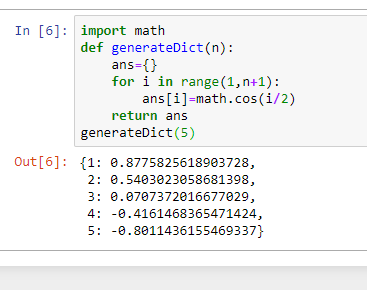
**請在下面欄位貼上程式碼：**

|  |
| --- |
| **import math**  **def generateDict(n):**  **ans={}**  **for i in range(1,n+1):**  **ans[i]=math.cos(i/2)**  **return ans**  **generateDict(5)** |

**請執行 generateDict(5)，並將執行結果的數值貼在指定欄位:**

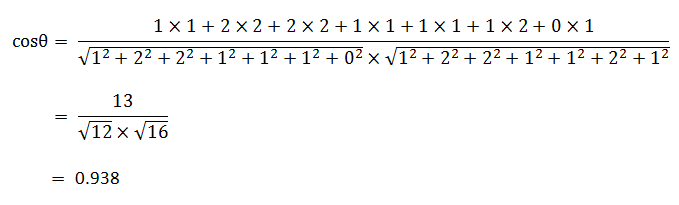
|  |  |
| --- | --- |
| 輸入: | 輸出: |
| generateDict(5) | {1: 0.8775825618903728,  2: 0.5403023058681398,  3: 0.0707372016677029,  4: -0.4161468365471424,  5: -0.8011436155469337} |

**執行結果擷圖：**



5: 假設A向量的值為[A1,A2,A3]，B向量是[B1,B2,B3]，則向量A與向量B的夾角餘弦值(餘弦相似度)為

舉例來說，A：[1, 2, 2, 1, 1, 1, 0], B：[1, 2, 2, 1, 1, 2, 1], 則



試寫一個函式CosSimilarity(a,b)，當我們輸入a,b兩筆相同長度的向量時，函式可輸出這兩筆向量的餘弦相似度。

例如:

|  |  |
| --- | --- |
| **輸入** | **執行結果** |
| **CosSimilarity([1, 2, 2, 1, 1, 1, 0],[1, 2, 2, 1, 1, 2, 1])** | 0.938 |
| **CosSimilarity([4,3,2],[1,6,2])** | 0.754 |

我的作答：(包含程式碼、貼上結果的數值、將執行結果擷圖貼上)：

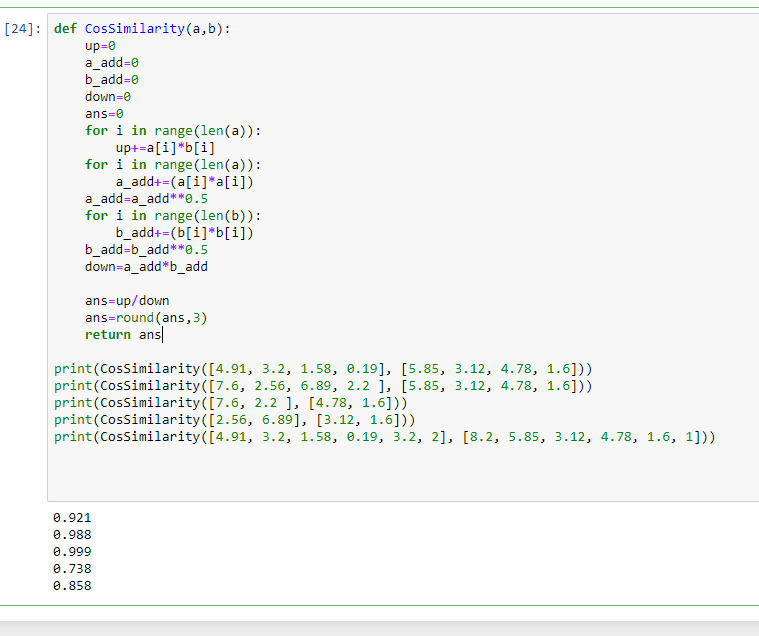
**請在下面欄位貼上程式碼：**

|  |
| --- |
| **def CosSimilarity(a,b):**  **up=0**  **a\_add=0**  **b\_add=0**  **down=0**  **ans=0**  **for i in range(len(a)):**  **up+=a[i]\*b[i]**  **for i in range(len(a)):**  **a\_add+=(a[i]\*a[i])**  **a\_add=a\_add\*\*0.5**  **for i in range(len(b)):**  **b\_add+=(b[i]\*b[i])**  **b\_add=b\_add\*\*0.5**  **down=a\_add\*b\_add**    **ans=up/down**  **ans=round(ans,3)**  **return ans**  **print(CosSimilarity([4.91, 3.2, 1.58, 0.19], [5.85, 3.12, 4.78, 1.6]))**  **print(CosSimilarity([7.6, 2.56, 6.89, 2.2 ], [5.85, 3.12, 4.78, 1.6]))**  **print(CosSimilarity([7.6, 2.2 ], [4.78, 1.6]))**  **print(CosSimilarity([2.56, 6.89], [3.12, 1.6]))**  **print(CosSimilarity([4.91, 3.2, 1.58, 0.19, 3.2, 2], [8.2, 5.85, 3.12, 4.78, 1.6, 1]))** |

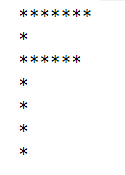
**請依下面要求輸入參數，並將執行結果的數值貼在指定欄位**

|  |  |
| --- | --- |
| **print(CosSimilarity([4.91, 3.2, 1.58, 0.19], [5.85, 3.12, 4.78, 1.6]))** | 0.921 |
| **print(CosSimilarity([7.6, 2.56, 6.89, 2.2 ], [5.85, 3.12, 4.78, 1.6]))** | 0.988 |
| **print(CosSimilarity([7.6, 2.2 ], [4.78, 1.6]))** | 0.999 |
| **print(CosSimilarity([2.56, 6.89], [3.12, 1.6]))** | 0.738 |
| **print(CosSimilarity([4.91, 3.2, 1.58, 0.19, 3.2, 2], [8.2, 5.85, 3.12, 4.78, 1.6, 1]))** | 0.858 |

**執行結果擷圖：**



6: 試撰寫一個Python程式來列印'F'這個字型的形狀如下：



我的作答：(包含程式碼、將執行結果擷圖貼上)：

**請在下面欄位貼上程式碼：**

|  |
| --- |
| **for row in range(0,7):**  **for col in range(0,7):**  **if row==0:**  **print('\*',end='')**  **if col==6:**  **print(end='\n')**  **elif row==2:**  **if col>5:**  **print(' ',end='')**  **if col<=5:**  **print('\*',end='')**  **if col==6:**  **print(end='\n')**  **else:**  **if col==0:**  **print('\*',end='')**  **elif col==6:**  **print(' ',end='')**  **print(end='\n')**  **else:**  **print(' ',end='')** |

**執行結果擷圖(本題請連同程式碼與執行結果一併擷圖)：**



7: 試寫一個Python函式remove\_5t(num)，它能移除輸入之串列(list)中所有為4的倍數或5的倍數的元素，並將結果放在一個tuple中回傳。

例如:

|  |  |
| --- | --- |
| **輸入** | **執行結果** |
| **print(remove\_5t([2,7, 11, 16, 20, 22, 25, 34, 55]))** | **(2, 7, 11, 22, 34)** |
| **print(remove\_5t([12,3,4,5,16,8,10,24,34,35,25,55]))** | **(3, 34)** |

我的作答：(包含程式碼、貼上結果的數值、將執行結果擷圖貼上)：

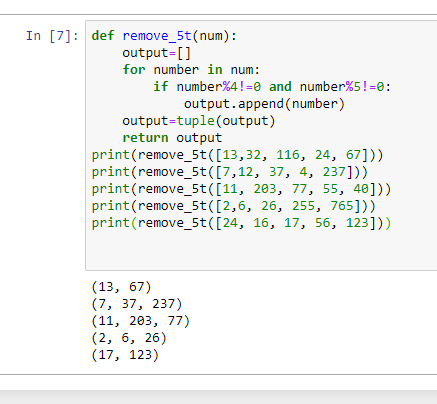
**請在下面欄位貼上程式碼：**

|  |
| --- |
| **def remove\_5t(num):**  **output=[]**  **for number in num:**  **if number%4!=0 and number%5!=0:**  **output.append(number)**  **output=tuple(output)**  **return output**  **print(remove\_5t([13,32, 116, 24, 67]))**  **print(remove\_5t([7,12, 37, 4, 237]))**  **print(remove\_5t([11, 203, 77, 55, 40]))**  **print(remove\_5t([2,6, 26, 255, 765]))**  **print(remove\_5t([24, 16, 17, 56, 123]))** |

**請依下面要求輸入參數，並將執行結果的數值貼在指定欄位:**

|  |  |
| --- | --- |
| **print(remove\_5t([13,32, 116, 24, 67]))** | (13, 67) |
| **print(remove\_5t([7,12, 37, 4, 237]))** | (7, 37, 237) |
| **print(remove\_5t([11, 203, 77, 55, 40]))** | (11, 203, 77) |
| **print(remove\_5t([2,6, 26, 255, 765]))** | (2, 6, 26) |
| **print(remove\_5t([24, 16, 17, 56, 123]))** | (17, 123) |

**執行結果擷圖：**



8: 試以NumPy程式產生一個由0所構成的8\*8矩陣(Fig.1(a))，

1. 將周邊一圈以外的0全部變為2，輸出型式如Fig.1(b)：
2. 將周邊第一圈和周圍第二圈以外的值全部變為4，輸出型式如Fig.1(c)：
3. 將周邊第一圈、第二圈和第三圈以外的值全部變為6，輸出型式如Fig.1(d)：

|  |  |
| --- | --- |
| (a) "Original array:" | (b) |
| (c) | (d) |

Fig.1

我的作答：(包含程式碼、將執行結果擷圖貼上)：

**請在下面欄位貼上程式碼：**

|  |
| --- |
| **import numpy as np**  **np1=np.zeros((8,8))**  **print(np1)**  **for row in range(0,len(np1)):**  **for col in range(0,len(np1)):**  **if row!=0 and row!=len(np1)-1 and col!=0 and col!=len(np1)-1:**  **np1[row][col]=2**  **print(np1)**  **for row in range(0,len(np1)):**  **for col in range(0,len(np1)):**  **if row!=0 and row!=len(np1)-1 and col!=0 and col!=len(np1)-1:**  **if row!=1 and row!=len(np1)-2 and col!=1 and col!=len(np1)-2:**  **np1[row][col]=4**  **print(np1)**  **for row in range(0,len(np1)):**  **for col in range(0,len(np1)):**  **if row!=0 and row!=len(np1)-1 and col!=0 and col!=len(np1)-1:**  **if row!=1 and row!=len(np1)-2 and col!=1 and col!=len(np1)-2:**  **if row!=2 and row!=len(np1)-3 and col!=2 and col!=len(np1)-3:**  **np1[row][col]=6**  **print(np1)** |

**執行結果擷圖(因結果較長，可連同程式碼和結果分兩次擷圖，再分別貼在欄位左邊和右邊位置)：**

|  |  |
| --- | --- |
|  |  |

9: 對於txt資料，在Numpy裡可以使用.loadtxt或是np.genfromtxt來讀取它。下面輸入的程式可以下載iris data的150資料(irisdata)，每筆資料有4個維度(花萼的長度、寬度和花瓣的長、寬)。試將irisdata第0行(花萼的長度)的150筆資料進行正規化，使其值的分布介於0到1之間。

輸入：

import numpy as np

url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data'

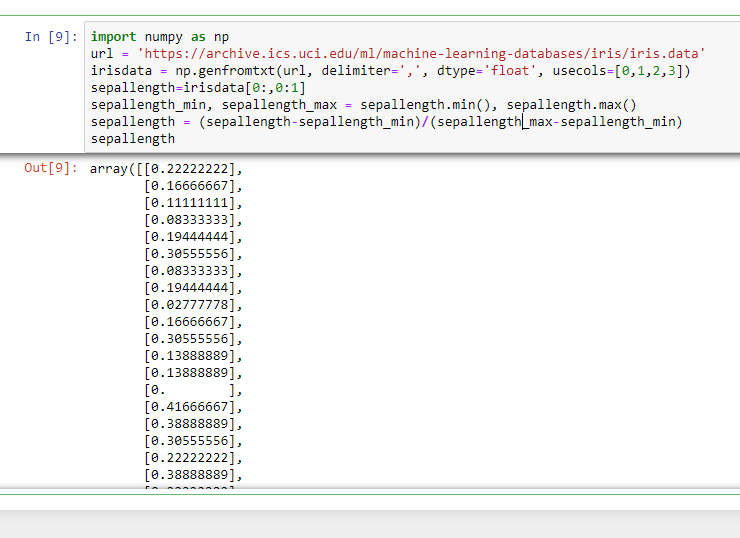
irisdata = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])

我的作答：(包含程式碼、將執行結果擷圖貼上)：

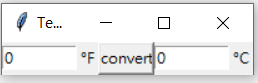
**請在下面欄位貼上程式碼：**

|  |
| --- |
| **import numpy as np**  **url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data'**  **irisdata = np.genfromtxt(url, delimiter=',', dtype='float', usecols=[0,1,2,3])**  **sepallength=irisdata[0:,0:1]**  **sepallength\_min, sepallength\_max = sepallength.min(), sepallength.max()**  **sepallength = (sepallength-sepallength\_min)/(sepallength\_max-sepallength\_min)**  **sepallength** |

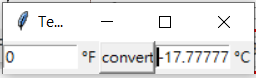
**執行結果擷圖：**

****

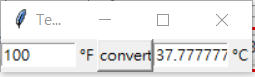
10: 試利用GUI的文字方塊功能實現一個華氏對攝氏的轉換程式(攝氏 = (華氏-32)\*5/9)，其介面外觀如下:



當我們在左側文字方塊輸入0°F時，按下convert按鈕，右邊的文字方塊會顯示 -17.777°C:



當我們在左側文字方塊輸入100°F時，按下convert按鈕，右邊的文字方塊會顯示 37.777°C:



我的作答：(包含程式碼、將執行結果擷圖貼上)：

**請在下面欄位貼上程式碼：**

|  |
| --- |
| **from tkinter import \***  **def add(): # 加法運算**  **n2.set((n1.get()-32)\*5/9)**  **window = Tk()**  **window.title("ex3") # 視窗標題**  **n1 = IntVar()**  **n2 = IntVar()**  **e1 = Entry(window,width=8,textvariable=n1) # 文字方塊1**  **label1 = Label(window,width=3,text='°F') # 加號**  **e2 = Entry(window,width=8,textvariable=n2) # 文字方塊2**  **btn = Button(window,width=5,text='convert',command=add) # =按鈕**  **label2= Label(window,width=3,text='°C') # 儲存結果文字方塊**  **e1.grid(row=0,column=0) # 定位文字方塊1**  **label1.grid(row=0,column=1) # 定位加號**  **btn.grid(row=0,column=2)**  **e2.grid(row=0,column=3)**  **label2.grid(row=0,column=4) # 定位儲存結果**  **window.mainloop()** |

**執行結果擷圖：**

例如:

|  |  |
| --- | --- |
| **左側文字方塊輸入值** | **執行結果擷圖 (請擷圖並貼在欄位上)** |
| 50°F | 10.0 |
| 80°F | 26.666666666666668 |

