102306061 資管三甲 何秉哲 HW\_2

以下為python程式檔的內容程式碼：

‘’’

file = open('hw2\_data.doc', mode = 'r')

result1 = open('hw2\_result.doc', mode = 'w')

frstn = []

lstn=[]

stdid=[]

avgs=[]

def count\_averges(avgs):

file = open('hw2\_data.txt', mode = 'r')

for line in file.readlines():

s = 0

w = 0

avg = 0

for i in line.rstrip().split(' ')[3:18:2]:

i = int(i)

s = s + i

for x , y in zip(map(int , line.rstrip().split(' ')[3:18:2]) , map(int ,line.rstrip().split(' ')[4:19:2])):

w = w + x\*y

avg = w / s

avgs.append(avg)

file.close()

return avgs

def make\_lastname\_list(lstn):

file = open('hw2\_data.txt', mode = 'r')

for line in file.readlines():

lstn.append(line.rstrip().split(' ')[0])

file.close()

return lstn

def make\_firstname\_list(frstn):

file = open('hw2\_data.txt', mode = 'r')

for line in file.readlines():

frstn.append(line.rstrip().split(' ')[1])

file.close()

return frstn

def make\_student\_id(stdid):

file = open('hw2\_data.txt', mode = 'r')

for line in file.readlines():

stdid.append(line.rstrip().split(' ')[2])

file.close()

return stdid

def sort\_with\_score(frstn , lstn , stdid, avgs):

l = []

a = zip(frstn , lstn , stdid, avgs)

for frstn , lstn , stdid, avgs in a:

l.append((avgs,lstn , frstn , stdid))

l.sort(reverse = True)

result1.write("以學期平均成績排序，由大到小排列。排序為；學期平均成績，姓名，學號 ： " + '\n')

for a in range(len(l)):

result1.write("\n")

for b in range(len(l[a])):

result1.write(str(l[a][b]) + " ")

result1.write('\n' + '\n')

def sort\_with\_id(frstn, lstn, stdid,avgs):

l = []

a = zip(frstn , lstn , stdid, avgs)

for frstn , lstn , stdid, avgs in a:

l.append((stdid,lstn , frstn , avgs))

l.sort()

result1.write("以學號排序，由大到小排列。排序為；學號、姓名、學期平均成績 ： " + '\n')

for a in range(len(l)):

result1.write("\n")

for b in range(len(l[a])):

result1.write(str(l[a][b]) + " ")

result1.write('\n' + '\n')

def sort\_with\_name(frstn, lstn, stdid, avgs):

l = []

a = zip(frstn , lstn , stdid, avgs)

for frstn , lstn , stdid, avgs in a:

l.append((lstn , frstn , stdid, avgs))

l.sort()

result1.write("以姓名排序，由大到小排列。排序為；姓名、學號、學期平均成績 ： " + '\n')

for a in range(len(l)):

result1.write("\n")

for b in range(len(l[a])):

result1.write(str(l[a][b]) + " ")

result1.write('\n')

make\_student\_id(stdid)

make\_firstname\_list(frstn)

make\_lastname\_list(lstn)

count\_averges(avgs)

result1.write("102306061 資管三甲 何秉哲 HW\_2" + '\n' + '\n' + '\n')

sort\_with\_score(frstn , lstn , stdid, avgs)

sort\_with\_id(frstn, lstn, stdid,avgs)

sort\_with\_name(frstn, lstn, stdid, avgs)

file.close()

result1.close()

‘’’

輸出結果(分別寫入(write)至‘hw2\_result.doc’檔及‘hw2\_result.txt’檔)：

102306061 資管三甲 何秉哲 HW\_2

以學期平均成績排序，由大到小排列。排序為；學期平均成績，姓名，學號 ：

92.07692307692308 Lastname02 Firstname02 104207414

85.31818181818181 Lastname30 Firstname30 104207428

84.82142857142857 Lastname16 Firstname16 104207436

84.57142857142857 Lastname37 Firstname37 104207423

84.38461538461539 Lastname19 Firstname19 104207418

83.27777777777777 Lastname09 Firstname09 104207434

81.29411764705883 Lastname12 Firstname12 104207426

81.1304347826087 Lastname17 Firstname17 104207430

81.04761904761905 Lastname23 Firstname23 104207403

81.0 Lastname14 Firstname14 104207412

80.66666666666667 Lastname13 Firstname13 104207410

79.84615384615384 Lastname15 Firstname15 104207425

79.7 Lastname03 Firstname03 104207413

79.6 Lastname01 Firstname01 104207416

79.57894736842105 Lastname31 Firstname31 104207407

79.57142857142857 Lastname28 Firstname28 104207402

78.69565217391305 Lastname29 Firstname29 104207411

78.55 Lastname34 Firstname34 104207437

77.47058823529412 Lastname36 Firstname36 104207415

77.38461538461539 Lastname10 Firstname10 104207432

77.36363636363636 Lastname18 Firstname18 104207422

77.3529411764706 Lastname27 Firstname27 104207433

77.28 Lastname05 Firstname05 104207439

76.72727272727273 Lastname07 Firstname07 104207424

76.45454545454545 Lastname26 Firstname26 104207431

76.2 Lastname24 Firstname24 104207405

76.0 Lastname20 Firstname20 104207420

75.6 Lastname40 Firstname40 104207404

75.54545454545455 Lastname32 Firstname32 104207435

75.125 Lastname33 Firstname33 104207417

74.91304347826087 Lastname08 Firstname08 104207438

74.6470588235294 Lastname04 Firstname04 104207406

74.21052631578948 Lastname21 Firstname21 104207429

73.66666666666667 Lastname22 Firstname22 104207409

73.57142857142857 Lastname25 Firstname25 104207427

73.0 Lastname38 Firstname38 104207408

72.8 Lastname06 Firstname06 104207421

72.69565217391305 Lastname35 Firstname35 104207440

72.28571428571429 Lastname39 Firstname39 104207419

71.61111111111111 Lastname11 Firstname11 104207401

以學號排序，由大到小排列。排序為；學號、姓名、學期平均成績 ：

104207401 Lastname11 Firstname11 71.61111111111111

104207402 Lastname28 Firstname28 79.57142857142857

104207403 Lastname23 Firstname23 81.04761904761905

104207404 Lastname40 Firstname40 75.6

104207405 Lastname24 Firstname24 76.2

104207406 Lastname04 Firstname04 74.6470588235294

104207407 Lastname31 Firstname31 79.57894736842105

104207408 Lastname38 Firstname38 73.0

104207409 Lastname22 Firstname22 73.66666666666667

104207410 Lastname13 Firstname13 80.66666666666667

104207411 Lastname29 Firstname29 78.69565217391305

104207412 Lastname14 Firstname14 81.0

104207413 Lastname03 Firstname03 79.7

104207414 Lastname02 Firstname02 92.07692307692308

104207415 Lastname36 Firstname36 77.47058823529412

104207416 Lastname01 Firstname01 79.6

104207417 Lastname33 Firstname33 75.125

104207418 Lastname19 Firstname19 84.38461538461539

104207419 Lastname39 Firstname39 72.28571428571429

104207420 Lastname20 Firstname20 76.0

104207421 Lastname06 Firstname06 72.8

104207422 Lastname18 Firstname18 77.36363636363636

104207423 Lastname37 Firstname37 84.57142857142857

104207424 Lastname07 Firstname07 76.72727272727273

104207425 Lastname15 Firstname15 79.84615384615384

104207426 Lastname12 Firstname12 81.29411764705883

104207427 Lastname25 Firstname25 73.57142857142857

104207428 Lastname30 Firstname30 85.31818181818181

104207429 Lastname21 Firstname21 74.21052631578948

104207430 Lastname17 Firstname17 81.1304347826087

104207431 Lastname26 Firstname26 76.45454545454545

104207432 Lastname10 Firstname10 77.38461538461539

104207433 Lastname27 Firstname27 77.3529411764706

104207434 Lastname09 Firstname09 83.27777777777777

104207435 Lastname32 Firstname32 75.54545454545455

104207436 Lastname16 Firstname16 84.82142857142857

104207437 Lastname34 Firstname34 78.55

104207438 Lastname08 Firstname08 74.91304347826087

104207439 Lastname05 Firstname05 77.28

104207440 Lastname35 Firstname35 72.69565217391305

以姓名排序，由大到小排列。排序為；姓名、學號、學期平均成績 ：

Lastname01 Firstname01 104207416 79.6

Lastname02 Firstname02 104207414 92.07692307692308

Lastname03 Firstname03 104207413 79.7

Lastname04 Firstname04 104207406 74.6470588235294

Lastname05 Firstname05 104207439 77.28

Lastname06 Firstname06 104207421 72.8

Lastname07 Firstname07 104207424 76.72727272727273

Lastname08 Firstname08 104207438 74.91304347826087

Lastname09 Firstname09 104207434 83.27777777777777

Lastname10 Firstname10 104207432 77.38461538461539

Lastname11 Firstname11 104207401 71.61111111111111

Lastname12 Firstname12 104207426 81.29411764705883

Lastname13 Firstname13 104207410 80.66666666666667

Lastname14 Firstname14 104207412 81.0

Lastname15 Firstname15 104207425 79.84615384615384

Lastname16 Firstname16 104207436 84.82142857142857

Lastname17 Firstname17 104207430 81.1304347826087

Lastname18 Firstname18 104207422 77.36363636363636

Lastname19 Firstname19 104207418 84.38461538461539

Lastname20 Firstname20 104207420 76.0

Lastname21 Firstname21 104207429 74.21052631578948

Lastname22 Firstname22 104207409 73.66666666666667

Lastname23 Firstname23 104207403 81.04761904761905

Lastname24 Firstname24 104207405 76.2

Lastname25 Firstname25 104207427 73.57142857142857

Lastname26 Firstname26 104207431 76.45454545454545

Lastname27 Firstname27 104207433 77.3529411764706

Lastname28 Firstname28 104207402 79.57142857142857

Lastname29 Firstname29 104207411 78.69565217391305

Lastname30 Firstname30 104207428 85.31818181818181

Lastname31 Firstname31 104207407 79.57894736842105

Lastname32 Firstname32 104207435 75.54545454545455

Lastname33 Firstname33 104207417 75.125

Lastname34 Firstname34 104207437 78.55

Lastname35 Firstname35 104207440 72.69565217391305

Lastname36 Firstname36 104207415 77.47058823529412

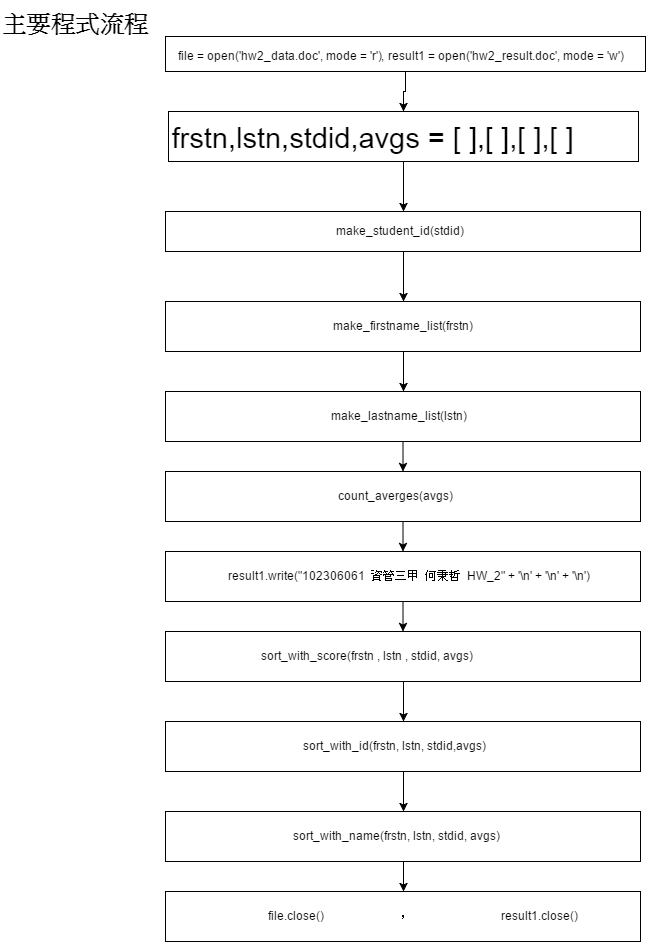
Lastname37 Firstname37 104207423 84.57142857142857

Lastname38 Firstname38 104207408 73.0

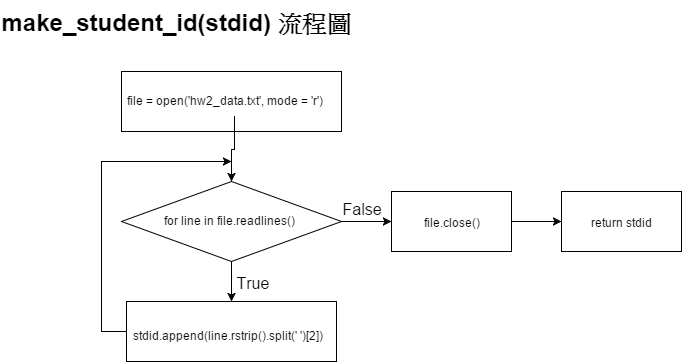
Lastname39 Firstname39 104207419 72.28571428571429

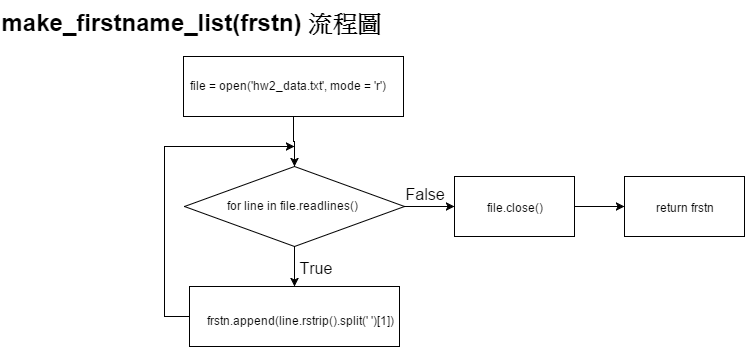
Lastname40 Firstname40 104207404 75.6

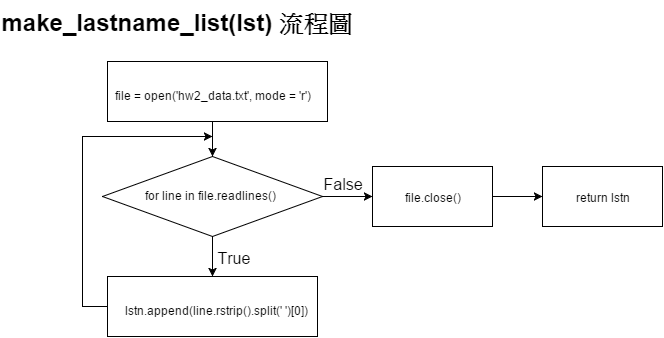
(以下流程圖透過draw.io完成)



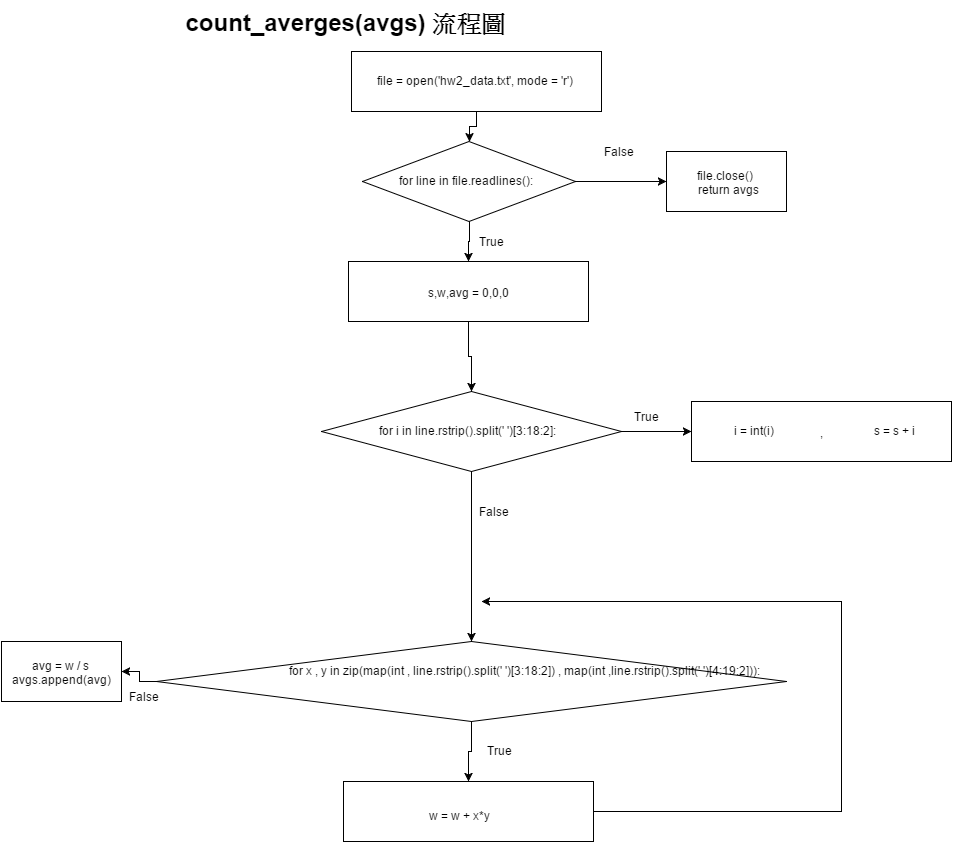
先分別以file = open('hw2\_data.doc', mode = 'r')，result1 = open('hw2\_result.doc', mode = 'w')讀取'hw2\_data.doc'檔(mode = 'r')並同時打開'hw2\_result.doc'檔以準備將檔案輸出結果寫入(mode = 'w')。宣布四個global variable : frstn, lstn, stdid, avgs 使他們成為empty lists，用以之後將hw2\_data.doc的內容字串的 名稱(firstname(frstn)和 lastname(lstn))，學號(stdid)，學期平均(avgs) 分別透過呼叫make\_firstname\_list(frstn)、make\_lastname\_list(lstn)、make\_student\_id(stdid)、count\_averges(avgs)運算、分出之後，將他們append進對應的list當中。之後寫入(.write())相關個人資訊後，透過呼叫sort\_with\_score(frstn , lstn , stdid, avgs)、sort\_with\_id(frstn, lstn, stdid,avgs)、sort\_with\_name(frstn, lstn, stdid, avgs) 將四個global list zip() 起來後， 將其以學期平均、學號、姓名進行排序後，將內容寫入'hw2\_result.doc'檔。最後再將兩個檔案關閉(.close())。



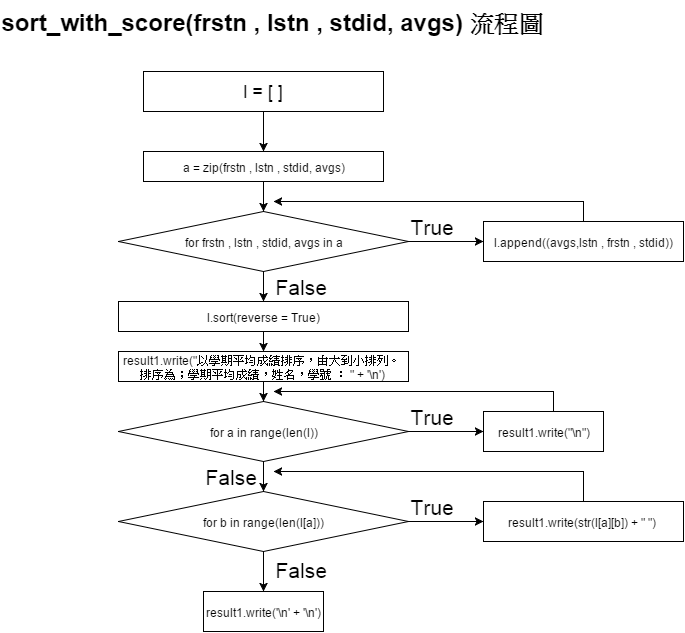


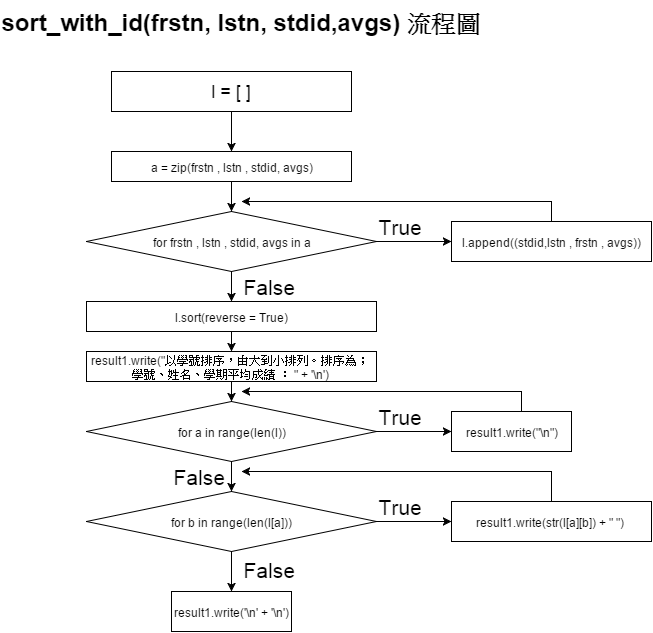


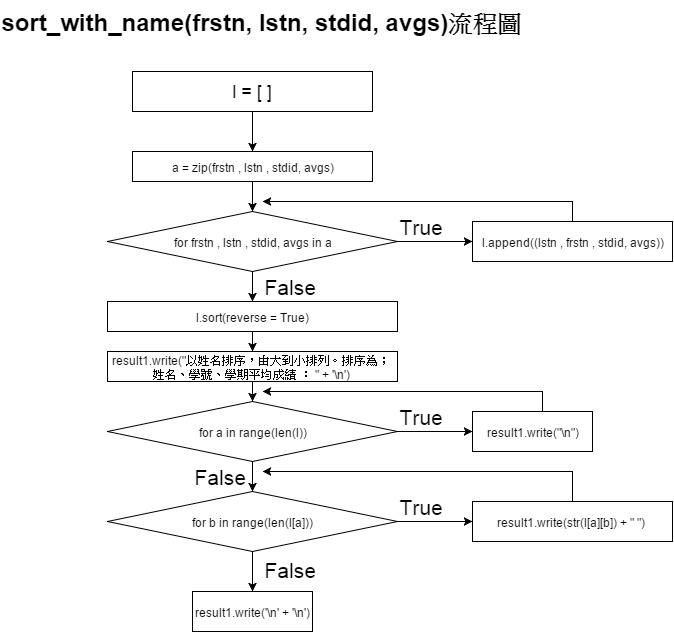
make\_lastname\_list(lstn)，make\_firstname\_list(frstn)，make\_student\_id(stdid)三個function架構一樣，分別將'hw2\_data.doc'開啟後，將內容一行一行的讀出來並將每一行字串最右邊的分行符號”\n”以rstrip()去除，並以split(“ ”)將字串以空格處為單位，並以索引直選取所要的物件(firstname, lastname, id, score)分別append 至相對應的list (frstn , lstn, stdid )中以準備在之後將學期平均算出來，存放進avgs的list後，把這四組list以zip()將對應的物件與值放入同一個tuple內，就可以直接對這些tuple進行排序。



將'hw2\_data.doc'開啟後，執行for loop將內容一行一行的讀出來，並在每次執行時將三個local variable: s,w,avg 的值assign成0，並再進入另一個for loop( for i in line.rstrip().split(' ')[3:18:2] )將右邊空格刪去並以空格為分割點，以索引值3到18每隔1個數字的內容取出(此為每項成績的學分)，以int()將他們從str轉換成為int型態再以 s = s + i(他們自己) 每個人的總學分數。之後再進入至另一個for loop (for x , y in zip(map(int , line.rstrip().split(' ')[3:18:2]) , map(int ,line.rstrip().split(' ')[4:19:2])))，這是分別以map()函式將line.rstrip().split(' ')[3:18:2] (學分數) 與 line.rstrip().split(' ')[4:19:2] (成績) 從str型態轉換成int，並且透過zip()將他們對應彼此(一個成績對應至他的學分，一起放進一個list 中的tuple)，並用 w = w + x\*y 將他們的相互乘積加總(加權)，之後再回到第一個for loop內將剛剛算出的w除以s assign給avg，並append給空白avgs的list，使avgs存放著每個學生的成績加權平均數(學期平均)。之後當把所有人的學期平均算出並append進avgs後，將'hw2\_data.doc'關閉，並將avgs這個list的值回傳。







sort\_with\_score(frstn , lstn , stdid, avgs) 、 sort\_with\_id(frstn, lstn, stdid,avgs)、sort\_with\_name(frstn, lstn, stdid, avgs) 架構一樣，先內建一個local variable : l 並將他的直設為空字串，再將之前的四個list (frstn , lstn , stdid , avgs) 以zip() 將每個四個對應之值放入list中的tuple並assign給variable a。並且進入for loop (for frstn , lstn , stdid, avgs in a) 將a list中每個tuple內的四個值以tuple型態append給空list: l，當執行完畢，將l list 以sort()進行排序( sort\_with\_score()中的sort()參數reverse = True 用意是將成績由大到小排序)，排序後再以write()將所要輸出的結果解釋先寫入檔案中，接著以for loop ( for a in range(len(l)) ) 再每當要寫入一個排序結果前先在檔案中換行 (‘\n’) ，在進入下一個for loop (for b in range(len(l[a])) ) 在每個list 中的tuple內之值取出並寫入檔案(result1.write(str(l[a][b]) + " ") (+ " " 的用意是將每個值之間以一個空格隔開) )，在跳脫出for loop後再用result1.write('\n' + '\n')寫入檔案中，用以換兩行，使每種排序結果之間空兩行，方便觀看。