

商管程式設計 109-1

TA Lab 5

2020/10/12-16
影片助教 彭晨

Agenda

- Debug 技巧
- 期中考複習
- 期中考提醒
- 考古題

Debug 技巧

如何Debug?

- 學會看錯誤訊息
 - 通常最下面那行會告訴你錯誤原因，如果不懂.....
 - File 開頭的那一行會告訴你是哪一行錯了

```
scary_news = "Midterm is coming!"  
num = int(scary_news)
```

```
pengchen@pengchendeMacBook-Air week5 % python3 wrong.py  
Traceback (most recent call last):  
  File "wrong.py", line 2, in <module>  
    num = int(scary_news)  
→   ↑  
    ValueError: invalid literal for int() with base 10: 'Midterm is coming!'
```

如何Debug?

- 如何避免邏輯錯誤：
 - 將程式拆成不同邏輯區塊
→ 善用空行
 - 1. 接收輸入
 - 2. 用一變數紀錄最大值
 - 3. 比大小
- 寫完一塊邏輯馬上測試是否正確
→ 善用 `print()` 測試

```
f1 = float(input())
f2 = float(input())
f3 = float(input())

max_f = f1

if max_f < f2:
    max_f = f2
if max_f < f3:
    max_f = f3

print(max_f)
```

考試 Debug 小技巧

- 注意輸出格式是否符合題目規定
- `print()` 出變數來檢查是否出錯
- 考慮**極端值**，自己輸入極端值進行測試

期中考複習

Review

期中考重點複習

- Type conversion
- Conditionals
- Nested conditionals
- **for** loop / **while** loop
- Nested loop
- **break** / **continue**
- String concatenation & **split()**
- **list** initialization
- **list** operation
- **list** iteration

p.s. 此處重點複習為根據實體助教課之間卷調查，超過四分之一以上同學認為的「學習上覺得困難之主題」

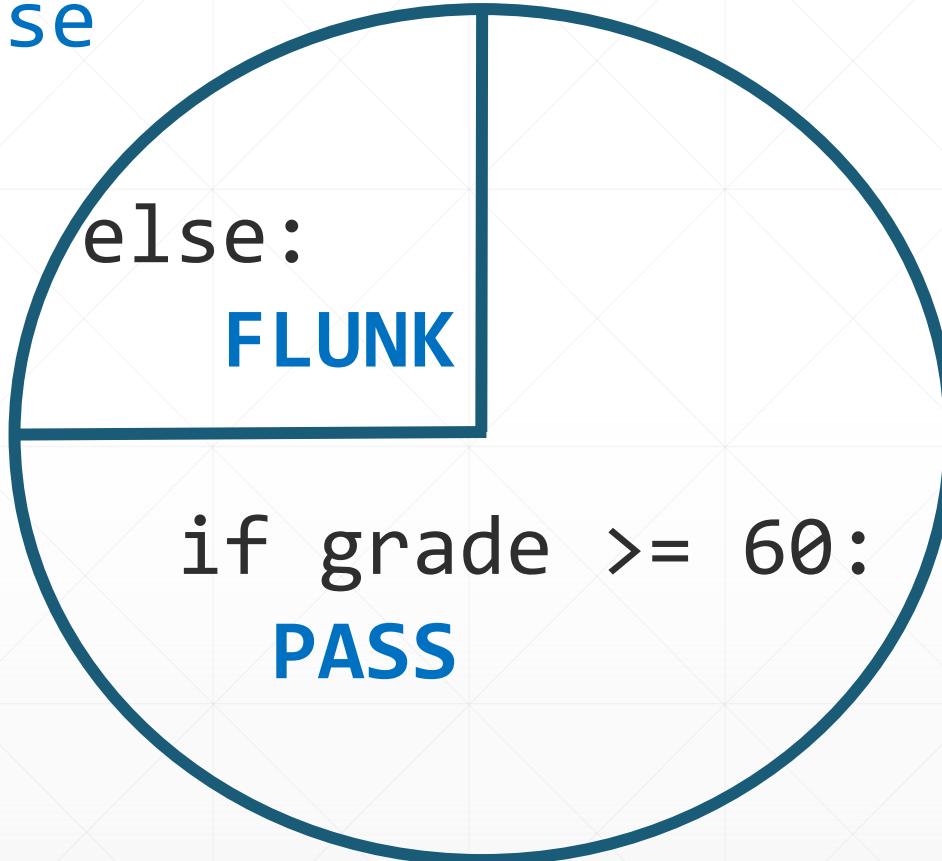
Type conversion

- `input()` 讀進來的一定是 string
 - 如果想得到整數？小數？ → 型別轉換
 - `num = int(input())`
 - `num = float(input())`
- 型別轉換函式
 - `int()` : 將其他型態的參數轉成整數型態
 - `str()` : 將其他型態的參數轉成字串型態
 - `float()` : 將其他型態的參數轉成浮點數型態
 - 注意: 不是所有型態轉換都可行！要看變數的值

```
a = "52"
b = int(a)
print(a)
print(b)
print(type(a))
print(type(b))
```

Conditionals

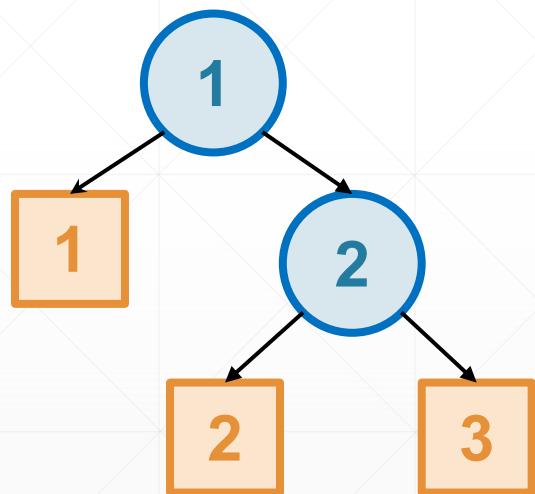
- if-else



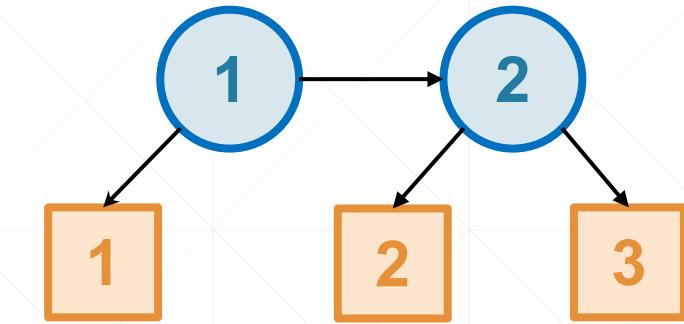
```
grade = int(input())  
if grade >= 60:  
    print("PASS")  
else:  
    print("FLUNK")
```

Nested conditionals

- 巢狀 if 判斷式
- 樹狀圖：每組 if-else 為一個節點



```
if condition1:  
    statement1  
else:  
    if condition2:  
        statement2  
    else:  
        statement3
```



- elif

```
if condition1:  
    statement1  
elif condition2:  
    statement2  
else:  
    statement3
```

for loop

- 從**序列**中獲取元素，將元素指派給前面的迭代變數
- 將常與 **range()** 配合使用
 - range(start, stop, step)**

```
for i in range(5):  
    print(i)
```

0
1

```
for i in range(0, 5):  
    print(i)
```

2
3

```
for i in range(0, 5, 1):  
    print(i)
```

4

```
for i in range(20, 2, -3):  
    print(i, end='/' )
```

20/17/14/11/8/5/

```
for i in range(3):  
    print(i, end=',')  
i = 7  
print(i)
```

0,7
1,7
2,7

while loop

- 在符合某種**條件**下，循環執行某段 statement

```
a = 0
while a <= 5:
    a += 1
    print(a)
```

1
2
3
4
5
6

Nested loop

- 當內圈執行完一輪，外圈才會進行下一項

```
for i in range(2):
    for j in range(3):
        print(i, j)
```

i = 0, j = 0	0 0
i = 0, j = 1	0 1
i = 0, j = 2	0 2
i = 1, j = 0	1 0
i = 1, j = 1	1 1
i = 1, j = 2	1 2

0 0
0 1
0 2
1 0
1 1
1 2

break

強制結束

```
for i in range(10):
    if i == 5:
        break
    else:
        print(i)
```

```
while condition:
    # 或者 for i in range:
        statement_block_1
    break
    statement_block_2
statement_block_3
```

0
1
2
3
4

continue

強制跳到下一次

```
for i in range(10):
    if i == 5:
        continue
    else:
        print(i)
```

```
while condition:
    # 或者 for i in range:
        statement_block_1
    continue
    statement_block_2
statement_block_3
```

0
1
2
3
4
6
7
8
9

String concatenation & split()

- 串聯字串 : +

```
q = "Quiz:"  
score = 100  
print(q + str(score))  
# 'Quiz:100'  
print(q, score)  
# 'Quiz: 100'
```

- str.split('分割符號')
- 預設用**空格**切割字串
- 可以更改成其他符號做切割

```
a = "apple banana orange"  
a = a.split()  
print(a)  
# ['apple', 'banana', 'orange']  
  
a = "apple,banana,orange"  
a = a.split(',')  
print(a)  
# ['apple', 'banana', 'orange']  
  
print(type(a))  
# <class 'list'>
```

list initialization

- 初始化 empty list

```
empty_list_1 = list()  
empty_list_2 = []  
  
multiple_list = [0]*3  
print(multiple_list)  
# [0, 0, 0]
```

list operations

- list 的 index 從 0 開始
- 最後一個 item 的 index 為 -1

方法	
.append(x)	將 x 加到 list 最後面
.sort()	將 list 由小到大排列
.reverse()	將 item 從最後一個排到第一個
.index(x)	將 x 第一次出現的 index 回傳
.insert(i,x)	將 x 加到位置 i
.count(x)	算 x 出現幾次
.remove(x)	將第一次出現的 x 刪掉
.pop(i)	將第 i 個 item 刪掉

list iteration

```
answer = [11, 22, 33, 44, 55]
# 從序列 ( answer ) 中獲取元素
# 將元素指派給前面的迭代變數 ( ans )
for ans in answer:
    print(ans, end=',')
```

11,22,33,44,55,

```
answer = [11, 22, 33, 44, 55]
# 使用 list 的 index 列印 list 的任務
for i in range(len(answer)):
    if i < len(answer) - 1: # i < 4
        print(answer[i], end=',')
    else: # i == 4
        print(answer[i])
```

11,22,33,44,55

期中考複習順序

- 完成作業基本題 (must-do)
- 完成 Coursera
- 寫考古題 (optional)
 - PDOGS 有歷年的課程作業和考試

基礎



進階

⚠ 小提醒：實體助教課會簡單帶考古題，請大家在家裡先行練習考古題，以達到最好的練習效果！

期中考提醒

Reminder

期中考提醒

- 注意考試規定與考場
 - 電腦充飽
 - 題目以 NTU COOL 所公布的 pdf 檔為準
 - **Open book**，不能使用通訊軟體
 - 考試當下看不到得幾分，只能看到 Sample 測資有沒有過
 - 時間以 PDOGS 系統時間為準避免在最後關頭才上傳答案，以免因為網路不穩影響你的繳交時間。
 - 分數以最後一次上傳為主
 - 只看 PDOGS 有沒有過：不看課內課外、不看 coding style
 - 考場之後會公告，考試位置當天會在教室內公布
 - 大家加油！

考古題

PBC 108-2 Midterm 1 (2)

<https://pdogs.ntu.im/judge/problem/view/2005010702>

- 紿定 n 個整數 x_1, x_2 直到 x_n ，請計算總共有多少個數字介於 t_1 和 t_2 之間（含 t_1 和 t_2 ）。
- 系統會提供一共 15 組測試資料，每組測試資料裝在一個檔案裡。

在每個檔案中有兩行，第一行是三個整數 n 、 t_1 和 t_2 ，依序代表共給定幾個數字、範圍下限和上限。

第二行是一串數字 x_1 、 x_2 直到 x_n 。

其中 $1 \leq n \leq 100$ 、 $1 \leq x_i \leq 1000$ 、 $1 \leq t_1 \leq 1000$ 、 $1 \leq t_2 \leq 1000$ ，

任兩個數字之間被一個逗號隔開。

Sample 1

Input:

8,3,9

Output:

5

7,4,1,8,3,11,6,2

Sample 2

Input:

10,1,4

21,3,5,8,2,1,6,9,13,16

Output:

3

PBC 108-1 Problem 3.

<https://pdogs.ntu.im/judge/problem/view/1902020703>

- 我們知道棒球比賽中，一位打者的打擊可能有許多種可能。為了簡化題目，讓我們在本題中只考慮「安打」、「出局」與「保送」三種可能，則這位打者的「打席」為其站上打擊區的總次數（亦即安打數加出局數加保送數等於打席），「打數」則為打席扣掉保送數，最後「打擊率」則是「安打數」除以「打數」。一位球員的打擊記錄為一串由 o、x 和 - 組成的列表，其中 o 代表安打，x 代表出局，- 代表因保送而不計打數。本題我們將給定兩位球員的打擊紀錄，請判斷何者的打擊率較高，並計算該球員的安打數。

注意：如果比較兩個分數 a/b 和 c/d ，不無可能遇到浮點數誤差的問題。更保險的方式，應該是比較 ad 和 bc 誰大。

PBC 108-1 Problem 3.

- **Input:**

系統會提供一共 10 組測試資料，每組測試資料裝在一個檔案裡。在每個檔案中有兩行，兩行皆為 o、x 和 - 組成的列表，第一行代表球員 1，第二行代表球員 2。

任兩個符號之間被一個逗號隔開，其中符號的數量小於等於 1000。

- **Output:**

讀入這些資料之後，請輸出打擊率較高的球員代碼（1 或 2），接著輸出該球員的安打數，兩個數字以逗號隔開。如果打擊率相同，則取安打較多那位球員。如果都相同，則取球員 1。

- **Hint :**
 1. 記下兩個打者的安打數跟打席數
 2. 比較打擊率

Sample 1

Input:

o,o,-,x,-
x,-,x,o,x

Output:

1, 2

Sample 2

Input:

o,x,-,-,x,o,x,x
o,-,x,o,x,o,x

Output:

2, 3

PBC 108-2 Midterm 1 (4)

<https://pdogs.ntu.im/judge/problem/view/2005010704>

- 假設現在有 n 位商人和 m 樣商品，在市場上共進行過 t 次販賣交易，每次交易中進行銷售的商人編號、賣出的商品編號以及顧客給予的評分皆有被記錄下來。根據給定的交易資料，請針對指定要尋找的商品編號 p ，找出該項商品之銷售評分最高的商人，並輸出該評分和商人編號，兩者以一個冒號隔開；若有多位商人評分同為最高，則依商人編號由小到大依序印出，兩兩以一個逗號隔開。舉例來說，若 $n = 2$ 、 $m = 2$ 、 $t = 3$ 、 $p = 1$ ，而三筆交易資料為

商人編號	商品編號	評分
1	1	5
1	2	4
2	1	2

則輸出 5:1。一位商人最多僅會販賣某商品一次，且指定的商品 p 一定至少有被至少一位商人販賣過。

PBC 108-2 Midterm 1 (4)

<https://pdogs.ntu.im/judge/problem/view/2005010704>

▪ Input

系統會提供一共 10 組測試資料，每組測試資料裝在一個檔案裡。

在每個檔案中有 $t + 1$ 列，第一列是四個正整數 n 、 m 、 t 和 p ，其中 n 代表商人數量， m 代表商品數量， t 代表交易資料筆數， p 代表指定尋找的商品。已知 $1 \leq n \leq 100$ 、 $1 \leq m \leq 100$ 、 $1 \leq t \leq 1000$ 。在第 2 到 $t + 1$ 列中，每列有 3 個整數，依序代表該筆交易的商人編號、商品編號，以及消費者評分。已知商人編號與商品編號都必為 1 到 100 之間的正整數、 p 必為 t 次交易牽涉到的商品編號中出現過至少一次的數字，且每筆交易的評分必介於 1 到 5。

同一列中的任兩個數字之間被一個逗號隔開。

▪ Output

讀入這些資料之後，請輸出指定商品最高評分和商人編號，兩者以一個冒號隔開；若賣過商品 p 的商人中有多人評分同為最高，請由小到大輸出商人編號，兩兩以一個逗號隔開。

PBC 108-2 Midterm 1 (4)

<https://pdogs.ntu.im/judge/problem/view/2005010704>

Sample 1

Input:	Output:
2,2,3,1	5:1
1,1,5	
1,2,4	
2,1,2	

Sample 2

Input:	Output:
3,3,6,2	3:1,2
1,1,5	
1,2,3	
2,3,1	
2,2,3	
1,3,3	
3,1,4	