

Lab	
HW	
Until	

การบ้านปฏิบัติการ 11

Sets, Dictionaries and I/O Redirection (20 คะแนน)

QJ	0			
ขอ	กา	าห	ห	ิด

- การเรียกใช้ฟังก์ชันเพื่อการทดสอบ ต้องอยู่ภายใต้เงื่อนไข if __name__ == '__main__' : เพื่อให้สามารถ
 import ไปเรียกใช้งานจาก Script อื่น ๆ ได้อย่างเป็นมาตรฐาน
- ii. ทุกข้อต้องมีการสร้างฟังก์ชัน my_id() โดยให้คืนค่าสายอักขระแทนเลขประจำตัวนักศึกษา 9 หลัก
- 1) **4 คะแนน** (Lab11_1_6XXXXXXX.py) ให้เขียนฟังก์ชัน word_count(*text*) เพื่อคืนค่า dictionary ผลลัพธ์จาก การนับจำนวนคำที่ปรากฏในสายอักขระ (String) *text* โดยฟังก์ชันจะ คืนค่าเป็น dict ที่มี *key* เป็นแต่ละคำที่ ปรากฏใน *text* และมี *value* เป็นความถี่ ทั้งนี้ตัวอักษรที่อยู่ใน *key* จะต้องเป็นตัวอักษรพิมพ์เล็กเท่านั้น ข้อกำหนด
 - การนับความถี่จะเป็นแบบ Case Insensitive ('ant' และ 'Ant' ถือเป็นคำเดียวกัน)
 - ข้อความในไฟล์จะเป็นภาษาอังกฤษมาตรฐานในรูปแบบที่ถูกต้อง (well-formed English)
 - ไม่พิจารณาเครื่องหมายวรรคตอนต่าง ๆ <u>เฉพาะที่ล้อมรอบคำ</u> เช่น !"#\$%&'()*+,./:;<=>?@[\]^_`{|}~
 - เนื่องจาก key มีคุณสมบัติเป็น set ลำดับในการแสดงผลใน output จึงไม่จำเป็นต้องเหมือนตัวอย่าง

<u>Input</u> Output

"He doesn't want to pay \$40,000 for a	{'new': 1,
new car, but his wife doesn't seem to	'but': 1,
care."	'pay': 1,
	'want': 1,
	'seem': 1,
	'care': 1,
	'his': 1,
	'40,000': 1,
	'wife': 1,
	'a': 1,
	'for': 1,
Chiana Mai	'car': 1,
Chiang Mai	"doesn't": 2,
	'to': 2,
	'he': 1}

• การวิเคราะห์ปัญหา

• Input:		จำนวนข้อมูล	ชนิดข้อมูล	
• Output:	(แสดงค่า)	จำนวนข้อมูล	ชนิดข้อมูล	
	(คืนค่า)	จำนวนข้อมูล	ชนิดข้อมูล	

2) 4 คะแนน (Lab11_2_6XXXXXXX.py) ให้เขียนฟังก์ชัน matching_sum(t, target_value) เพื่อ<u>คืนค่า</u> list ของจำนวนเต็มสองจำนวนใน tuple t ที่มีผลรวมเท่ากับจำนวนเต็ม target_value ทั้งนี้สมาชิกใน tuple t จะ เป็นจำนวนเต็มเท่านั้นและจะมีสมาชิกอย่างน้อย 1 ตัวเสมอ โดยหากไม่สามารถหาจำนวนสองจำนวนดังกล่าวได้ ให้ คืนค่า list ว่าง และหากมีสมาชิกใน t มากกว่า 1 คู่ ที่สามารถเป็นคำตอบได้ ให้คืนค่า<u>เพียงคำตอบเดียว</u>เท่านั้น <u>Hint</u>: การจับคู่สมาชิกทั้งหมด แล้วนำมาตรวจสอบผลบวกทีละคู่ จะทำให้เวลาบน Grader เกิน ควรแก้ปัญหาโดยใช้ collections ประเภท set หรือ dict

Function Call	<u>Output</u>

matching_sum((1,), 1)	[]
matching_sum((5, 2), 7)	[5, 2]
	[2, 5]
matching_sum((10, -1, 1, -8, 3, 1), 2)	[10, -8]
	[-8, 10]
	[-1, 3]
	[1, 1]
matching_sum((10, -1, 1, -8, 3, 1), 10)	[]

• การวิเคราะห์ปัญหา

• Input:		จำนวนข้อมูล	_ชนิดข้อมูล
• Output:	(แสดงค่า)	จำนวนข้อมูล	_ชนิดข้อมูล
	(คืนค่า)	จำนวนข้อมูล	_ชนิดข้อมูล

3) **4 คะแนน** (HW11_1_6XXXXXXX.py) ให้เขียนฟังก์ชัน words_to_num(*words*) เพื่อทำการคืนค่า<u>จำนวนเต็ม</u> บวกที่คำนวณได้จากคำอ่านของจำนวนเต็ม *words* โดยผลลัพธ์จะมีความยาวไม่เกิน 12 หลัก
Hint:

- สามารถศึกษาการอ่านตัวเลขในภาษาอังกฤษได้จาก http://en.wikipedia.org/wiki/English_numerals
- •สามารถใช้ฟังก์ชัน num_to_word() ที่เคยเขียนไว้ในสัปดาห์ก่อนๆ ในการทดสอบเช่น assert word_to_num(num_to_word(5)) == 5

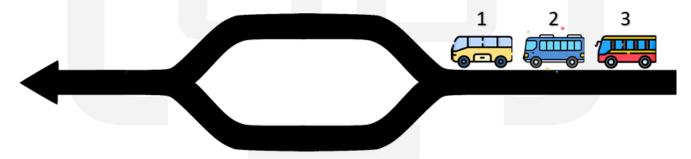
<u>Input</u> <u>Output</u>

fourteen	14
two hundred forty-eight	248
one hundred eleven	111
forty-two billion six hundred forty-one million three hundred twenty-three thousand eight hundred sixty-two	42641323862

• การวิเคราะห์ปัญหา

• Input:		จำนวนข้อมูล	ชนิดข้อมูล	
• Output:	(แสดงค่า)	จำนวนข้อมูล	ชนิดข้อมูล	
	(คืนค่า)	จำนวนข้อมูล	ชนิดข้อมูล	

4) 4 คะแนน (HW11_2_6XXXXXXX.py) ท่ามกลางพายุฝนที่โหมกระหน่ำ ขบวนรถบัสจากบริษัทสัมพุ่งมาถึงทาง แยก เพื่อกระจายความเสี่ยงที่จะพบปัญหาที่อาจเกิดขึ้นในเส้นทาง คนขับรถจึงตัดสินใจกระจายรถในขบวนให้วิ่งทั้ง สองทาง กล่าวคือในแต่ละเส้นทางจะต้องมีอย่างน้อย 1 คันเสมอ ในจากการสำรวจเส้นทางล่วงหน้าพบว่าทางทั้งสอง ไปบรรจบกันที่ระยะ 50 เมตรก่อนถึงโรงแรมที่หมาย เพื่อความปลอดภัยรถทุกคันจะไม่แซงกัน และจะไม่ขับถอยหลัง



หน้าที่ของคุณคือให้เขียนฟังก์ชัน split_and_merge(n) <u>เพื่อคืนค่า</u> list ของ String แทนลำดับที่เป็นไป ได้ทั้งหมดที่รถบัสของบริษัทสัมจะถึงโรงแรม โดย String ของลำดับที่เป็น Output จะอยู่ในรูป เลขประจำรถ 1 ถึง n คั่นด้วย '>' (เครื่องหมายมากกว่า) ทั้งนี้ String ใน list ที่คืนค่าจะอยู่ในลำดับใดก็ได้และจะต้องไม่มีลำดับการ มาถึงที่ซ้ำกัน และกำหนดให้ถนนทุกเส้น<u>เป็นถนนเลนเดียวและเดินรถได้ทางเดียว</u>

Hint: พิจารณานำฟังก์ชัน arrival_sequnces() จากการบ้าน w09 มาเรียกใช้

<u>Input</u> <u>Output</u>

3	['1>2>3', '1>3>2', '2>1>3', '2>3>1', '3>1>2']
4	['1>2>3>4', '1>2>4>3', '1>3>2>4', '1>3>4>2', '1>4>2', '2>1>3>4', '2>1>3>4', '2>1>4>3', '2>1>4>1', '2>1>4>3', '2>3>1>4', '2>1>4>1', '2>4>1>3', '3>1>2>4', '3>1>4>2', '3>4>1>2', '4>1

5) 4 คะแนน (HW11_3_5XXXXXXX.py) ให้เขียนฟังก์ชัน runner_up() เพื่อรับค่าคะแนนของนักศึกษาในห้อง ตามจำนวนนักศึกษาที่ระบุในบรรทัดแรก แล้ว<u>แสดงผล</u>คะแนนที่สูงเป็นอันดับที่ 1, อันดับที่ 2 และ ค่าเฉลี่ยคะแนน (ทศนิยม 2 ตำแหน่ง) โดยหากไม่มีตำแหน่งที่ 2 ให้ แสดงคำว่า None ข้อกำหนด: ไม่อนุญาตให้ใช้ตัวแปรประเภท iterables เช่น list หรือ tuple ในการเก็บข้อมูลที่ได้รับเพื่อการ คำนวณ

Hint: ศึกษา Slide Input/Output Redirection เพื่อความสะดวกในการทดสอบข้อมูลนำเข้าจำนวนมาก

<u>ตัวอย่างการ run 1</u>

ตัวอย่างการ run 2

Total students: 3	Total students: 2
Enter score:	Enter score:
13	61
12	61
13	
	Max score is: 61.00
Max score is: 13.00	Runner up is: None
Runner up is: 12.00	Average is: 61.00
Average is: 12.67	

<u>ตัวอย่างการ run 3</u>

ตัวอย่างการ run 4

Total students: 7 Total students: 1	
Enter score: Enter score:	
61	
72	
Max score is: 13.00	
Runner up is: None	
61 Average is: 13.00	
79	
63	
Max score is: 81.00	
Runner up is: 79.00	
Average is: 68.71	

• การวิเคราะห์ปัญหา

• Input: จำนวนข้อมูล_____ชนิดข้อมูล__

Chiang Mai University

การ<u>ส่งงาน</u>

- 1. ลักษณะ/ลำดับข้อความของการรับค่า/แสดงผล จะ<u>ต้องเป็นไปตามที่ระบ</u>ุในตัวอย่างการ run
- 2. ไฟล์งานที่ส่ง จะต้องมีการแทรก comment ที่ต้นไฟล์ตามข้อกำหนดใน canvas รายวิชา
- 3. ไฟล์งานโปรแกรมที่ส่ง จะต้องมีการแทรก pseudocode เป็น comment ในแต่ละขั้นตอน
- 4. Upload ไฟล์ source code ตามที่ระบุในแต่ละข้อ ไปยังระบบตรวจให้คะแนนอัตโนมัติ <u>https://cmu.to/gdr111</u>