



HW+Lab	
D+2	

การบ้านปฏิบัติการ 13

Lists, Tuples, Sets and Dictionaries (20 คะแนน)

ข้อกำหนด

- การเรียกใช้ฟังก์ชันเพื่อการทดสอบ ต้องอยู่ภายใต้เงื่อนไข `if __name__ == '__main__':` เพื่อความสะดวกในการ import จาก Script อื่น ๆ
- สามารถใช้ Iteration หรือ Recursion แก้ปัญหาได้

- 1) 4 คะแนน (Lab13_1_6XXXXXXX.py) ให้เขียนฟังก์ชัน `matching_sum(t, target_value)` เพื่อคืนค่า list ของจำนวนเต็มสองจำนวนใน tuple `t` ที่มีผลรวมเท่ากับจำนวนเต็ม `target_value` ทั้งนี้สมาชิกใน tuple `t` จะเป็นจำนวนเต็มเท่านั้นและจะมีสมาชิกอย่างน้อย 1 ตัวเสมอ โดยหากไม่สามารถหาจำนวนสองจำนวนดังกล่าวได้ ให้คืนค่า list ว่าง และหากมีสมาชิกใน `t` มากกว่า 1 คู่ ที่สามารถเป็นคำตอบได้ ให้คืนค่าเพียงคำตอบเดียวเท่านั้น
- Hint:** การจับคู่สมาชิกทั้งหมด แล้วนำมาตรวจสอบผลบวกทีละคู่ จะทำให้เวลาบน Grader เกิน ควรแก้ปัญหาโดยใช้ **collections** ที่เรียนในสัปดาห์นี้

Function CallOutput

<code>matching_sum((1,), 1)</code>	<code>[]</code>
<code>matching_sum((5, 2), 7)</code>	<code>[5, 2]</code>
	<code>[2, 5]</code>
<code>matching_sum((10, -1, 1, -8, 3, 1), 2)</code>	<code>[10, -8]</code>
	<code>[-8, 10]</code>
	<code>[-1, 3]</code>
	<code>[1, 1]</code>
<code>matching_sum((10, -1, 1, -8, 3, 1), 10)</code>	<code>[]</code>

• การวิเคราะห์ปัญหา

- Input: จำนวนข้อมูล _____ ชนิดข้อมูล _____
- Output: (แสดงค่า) จำนวนข้อมูล _____ ชนิดข้อมูล _____
(คืนค่า) จำนวนข้อมูล _____ ชนิดข้อมูล _____

- Hint:** สามารถศึกษาการอ่านตัวเลขในภาษาอังกฤษได้จาก http://en.wikipedia.org/wiki/English_numerals

<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

- | | | | |
|-----------|-----------|-------------------|------------------|
| • Input: | | จำนวนข้อมูล _____ | ชนิดข้อมูล _____ |
| • Output: | (แสดงค่า) | จำนวนข้อมูล _____ | ชนิดข้อมูล _____ |
| | (คืนค่า) | จำนวนข้อมูล _____ | ชนิดข้อมูล _____ |

- 3) **4 คะแนน** (HW13_1_6XXXXXXX.py) ให้เขียนฟังก์ชัน `scramble(word)` เพื่อคืนค่า list ของคำที่เป็นไปได้ทั้งหมดจากการสลับอักษรใน `str word` (ไม่มีอักขระอื่นนอกจากตัวอักษรภาษาอังกฤษ) โดย list ที่ได้จะต้องไม่มีสมาชิกซ้ำกัน ทั้งนี้ไม่อนุญาตให้ `import` โมดูลเพิ่มเติมในการแก้ปัญหา

<u>Input</u>	<u>Output</u>
'Cat'	['Cat', 'Cta', 'aCt', 'atC', 'tCa', 'taC']
'bee'	['bee', 'ebe', 'eeb']
'bEe'	['Ebe', 'Eeb', 'bEe', 'beE', 'eEb', 'ebE']

- การวิเคราะห์ปัญหา

- Input: จำนวนข้อมูล _____ ชนิดข้อมูล _____
- Output: (แสดงค่า) จำนวนข้อมูล _____ ชนิดข้อมูล _____
- (คืนค่า) จำนวนข้อมูล _____ ชนิดข้อมูล _____

- 4) 4 คะแนน (HW13_2_6XXXXXXX.py) ให้เขียนฟังก์ชัน `subset_sum(set_a)` เพื่อคืนค่า list ของ ผลบวก สมาชิกของแต่ละเซตย่อยทั้งหมดของ `set_a` ซึ่งเซตย่อยดังกล่าวนั้นรวมถึงเซตว่าง (\emptyset) และ `set_a` เองด้วย โดยกำหนดให้ผลบวกของเซตว่าง คือ 0 ทั้งนี้ไม่อนุญาตให้ `import` โมดูลเพิ่มเติมในการแก้ปัญหา
- ในทางคณิตศาสตร์ถ้า S เป็นเซต $\{1, 2, 3\}$ แล้วเซตย่อยของ S ได้แก่:

- ☐ $\{\}$ (อาจเขียนแทนด้วย \emptyset ได้แก่เซตว่าง)
- ☐ $\{1\}$
- ☐ $\{2\}$
- ☐ $\{3\}$
- ☐ $\{1, 2\}$
- ☐ $\{1, 3\}$
- ☐ $\{2, 3\}$
- ☐ $\{1, 2, 3\}$

Input	Output
{1, 2, 3}	[0, 1, 2, 3, 3, 4, 5, 6]

• การวิเคราะห์ปัญหา

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

- 5) 4 คะแนน (HW13_3_6XXXXXXX.py) ให้เขียนฟังก์ชัน `sum_d_product(m)` เพื่อคืนค่าผลบวกของผลคูณทแยงใน matrix m ที่มีขนาด $n \times n$ เมื่อ n สามารถเขียนในรูปของ 2^x (x เป็นจำนวนเต็มบวก)

a	b
c	d

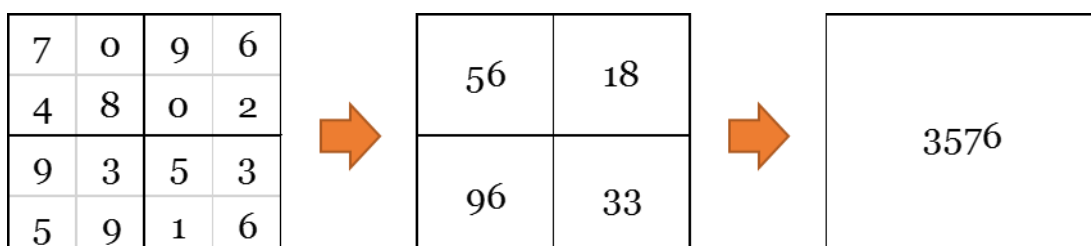
โดยกรณี matrix m ขนาด 2×2 เช่น

1	2
3	4

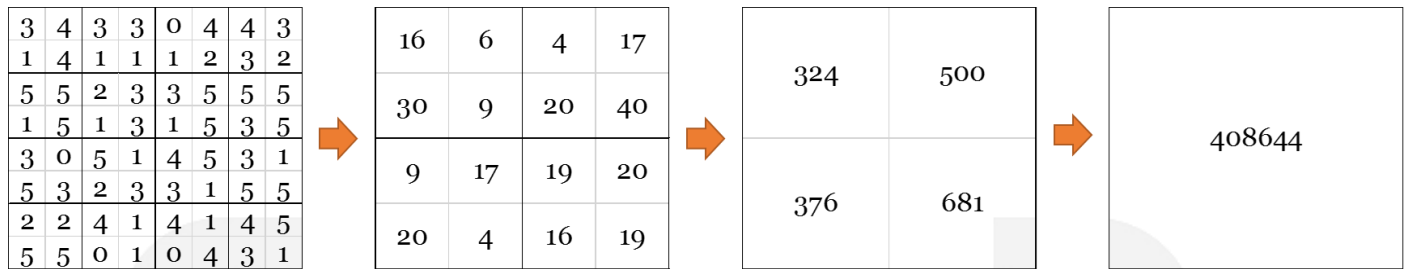
ดังนั้น matrix จะมี `sum_d_product()` = $(1 \times 4) + (3 \times 2) = 10$

กรณีต้องการหา `sum_d_product()` ของ matrix ขนาดใหญ่กว่า 2×2 ทำได้โดยการหา `sum_d_product()` ของ matrix ย่อย ขนาด 2×2 ก่อน แล้วหา `sum_d_product()` ของ matrix ผลลัพธ์อีกที

เช่นกรณี matrix ขนาด 4×4 จะมีขั้นตอนดังนี้



หรือกรณี matrix 8×8



Hint: ฟังก์ชัน `main()` ควรอ่าน Input จากไฟล์ด้วยวิธี Command Redirection เพื่อความสะดวกในการทดสอบและหาข้อผิดพลาด

Input	Output
[[3, 3, 3, 2], [2, 0, 3, 1], [2, 1, 2, 3], [1, 0, 2, -1]]	33
[[1, 1, 5, -1], [12, 2, -2, 0], [4, 8, 8, 12], [4, 12, 12, 15]]	3856
[[0, -1, -1, 3, 2, 3, -1, 3], [3, -1, -1, 2, 0, -1, 2, 1], [3, 0, 1, 2, 3, 1, 3, 1], [2, 2, 1, -1, -1, 2, 0, 3], [1, 3, 2, 1, 3, 2, 2, 1], [1, 2, 2, 1, 3, 3, 1, 3], [2, 2, 2, 2, 2, 2, 3, 3], [1, 3, 2, 3, 1, 1, 2, 2]]	-6290

- การวิเคราะห์ปัญหา

- Input: จำนวนข้อมูล _____ ชนิดข้อมูล _____
- Output: (แสดงค่า) จำนวนข้อมูล _____ ชนิดข้อมูล _____
(คืนค่า) จำนวนข้อมูล _____ ชนิดข้อมูล _____

การส่งงาน

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