

II IVI MOVI BIO				
Lab				
HW				

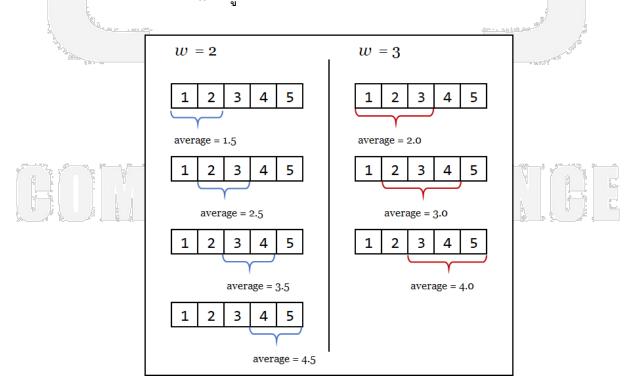
การบ้านปฏิบัติการ 8 1D Lists and Tuples Part I (20 คะแหน)

No.	Description of the second seco
ข้อกำหนด 🎢 💮 💮	200
i. การเรียกใช้ฟังก์ชันเพื่อการทดสอบ ต้องอยู่ภายใต่	ไเงื่อนไข ifname == 'main' : เพื่อความ
สะดวกในการ import จาก Script อื่น ๆ	Service Control of the Control of th
ii. <u>ไม่</u> อนุญาตให้ใช้ for, while (Iterations), หรือ	Data Type อื่น ๆ ที่ยังไม่สอนในบทเรียน เช่น dict หรือ
set ในการแก้ปัญหา	
iii. นักศึกษาสามารถสร้างฟังก์ชันย่อยต่าง ๆ เพิ่มเติม	มได้ตามความเหมาะสม
iv. ในข้อที่ระบุว่ามี [Attachments] ให้ Download ไ	
	Salar Sa
Hint: ควรใช้ Statement assert เพื่อทำการทดสอบฟังก์	ชับที่เทียงกับข้องการคลายรถาย พอโดยอัตโรงก็
HIII. YIJII Statement assert twen iii ijimaabanii	าหนเราอหนากอที่สนับผลอานุม เจ.) ก็ผเผรอผเหพื่
9 % 2	
1) 4 คะแหน (Lab08_1_6XXXXXXX.py) ให้เขียนฟัง	ก็ชับ pondect notate list(list a n) เพื่อท้าการ
หมุน (Rotate) list_a ไปทางขวา n ตำแหน่ง (หรือท	างซ้ายหาก n เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non-
	างซ้ายหาก n เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non-
หมุน (Rotate) list_a ไปทางขวา n ตำแหน่ง (หรือท	างซ้ายหาก n เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non-
หมุน (Rotate) $list_a$ ไปทางขวา n ตำแหน่ง (หรือท destructive (คืนค่า List ที่ Rotate แล้วโดย <u>ไม่เปลี่ย</u> น	างซ้ายหาก <i>n</i> เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non- แ <u>ปลง</u> List เดิม)
หมุน (Rotate) <i>list_a</i> ไปทางขวา <i>n</i> ตำแหน่ง (หรือท destructive (คืนค่า List ที่ Rotate แล้วโดย <u>ไม่เปลี่ยน</u> <u>Input</u> [1, 2, 3, 4]	างซ้ายหาก <i>n</i> เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non- <u>แปลง</u> List เดิม) <u>Output</u> [4, 1, 2, 3]
หมุน (Rotate) list_a ไปทางขวา n ตำแหน่ง (หรือท destructive (คืนค่า List ที่ Rotate แล้วโดย <u>ไม่เปลี่ยน Input</u> [1, 2, 3, 4] [1, 2, 3, 4]	างซ้ายหาก <i>n</i> เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non- แปลง List เดิม) <u>Output</u>
หมุน (Rotate) list_a ไปทางขวา n ตำแหน่ง (หรือท destructive (คืนค่า List ที่ Rotate แล้วโดย <u>ไม่เปลี่ยน Input</u> [1, 2, 3, 4] 1 [1, 2, 3, 4] 105	างซ้ายหาก <i>n</i> เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non-
หมุน (Rotate) list_a ไปทางขวา n ตำแหน่ง (หรือท destructive (คืนค่า List ที่ Rotate แล้วโดย <u>ไม่เปลี่ยน Input</u> [1, 2, 3, 4] [1, 2, 3, 4]	างซ้ายหาก <i>n</i> เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non- <u>แปลง</u> List เดิม) <u>Output</u> [4, 1, 2, 3]
หมุน (Rotate) list_a ไปทางขวา n ตำแหน่ง (หรือท destructive (คืนค่า List ที่ Rotate แล้วโดย <u>ไม่เปลี่ยน Input</u> [1, 2, 3, 4] [1, 2, 3, 4] [1, 2, 3, 4]	างซ้ายหาก <i>n</i> เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non-
หมุน (Rotate) list_a ไปทางขวา n ตำแหน่ง (หรือท destructive (คืนค่า List ที่ Rotate แล้วโดย <u>ไม่เปลี่ยน Input</u> [1, 2, 3, 4] [1, 2, 3, 4] [1, 2, 3, 4]	างซ้ายหาก <i>n</i> เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non-
หมุน (Rotate) list_a ไปทางขวา n ตำแหน่ง (หรือท destructive (คืนค่า List ที่ Rotate แล้วโดย <u>ไม่เปลี่ยน Input</u> [1, 2, 3, 4] 1 [1, 2, 3, 4] 105 [1, 2, 3, 4] -1	างซ้ายหาก n เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non- แปลง List เดิม) Output [4, 1, 2, 3] [2, 3, 4, 1]
หมุน (Rotate) list_a ไปทางขวา n ตำแหน่ง (หรือท destructive (คืนค่า List ที่ Rotate แล้วโดย <u>ไม่เปลี่ยน Input</u> [1, 2, 3, 4] 1 [1, 2, 3, 4] 105 [1, 2, 3, 4] -1	างซ้ายหาก <i>n</i> เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Non-

2) 4 คะแนน (Lab08_2_6xxxxxxxx.py) ให้เขียนฟังก์ชัน dest_rotate_list(list_a, n) เพื่อทำการหมุน (Rotate) list_a ไปทางขวา n ตำแหน่ง (หรือทางซ้ายหาก n เป็นลบ) โดยฟังก์ชันจะทำงานแบบ Destructive (ไม่ มีการคืนค่า และเปลี่ยนแปลง List เดิมให้เป็น List ที่ rotate แล้ว) ข้อกำหนด: <u>ไม่</u>อนุญาตให้เรียกใช้ฟังก์ชัน nondest_rotate_list() จาก Lab08_1 หรือสร้าง List ชั่วคราวที่มี ขนาดเกิน 2 element ขึ้นมาใช้พักข้อมูลในขณะดำเนินการ

Input .	<u>Output</u>	See All
[1, 2, 3, 4]	[4, 1, 2, 3]	a sality of a sality of the sa
[1, 2, 3, 4] 105	[4, 1, 2, 3]	3
[1, 2, 3, 4]	[2, 3, 4, 1]	
• การวิเคราะห์ปัญหา	A CONTRACTOR OF THE CONTRACTOR	a significant and a significan
• Input; จำ	นวนข้อมูลชนิดข้อมูล	
• Output: (แสดงค่า) จำ	นวนข้อมูลชนิดข้อมูล	
์ (คืนค่า) จำ	นวนข้อมูลชนิดข้อมูล	

3) **4 คะแนน** (HW08_1_6xxxxxxxx.py) ให้เขียนฟังก์ชัน moving_average ($list_a$, w) เพื่อคืนค่า List ของ ค่าเฉลี่ยเคลื่อนที่ (Moving Average) ของ element ใน $list_a$ ซึ่งเป็น List ของ<u>จำนวนจริง</u> โดยคำนวณตาม window size ที่กำหนดด้วยจำนวนเต็มบวก w ดังรูป



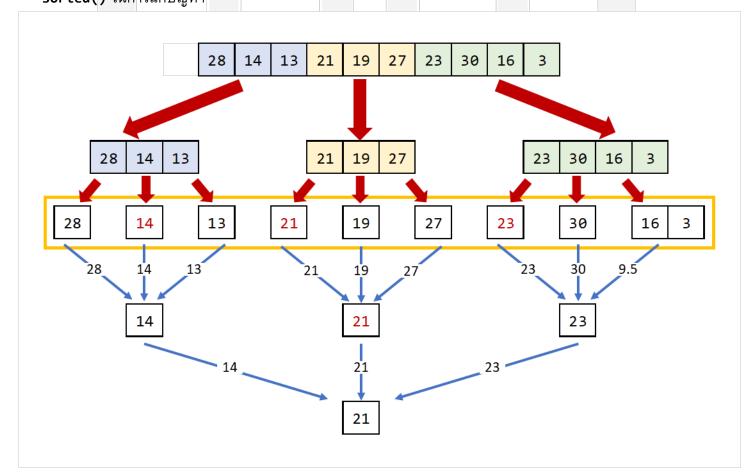
<u>Input</u>	<u>Output</u>
--------------	---------------

[1, 2, 3, 4, 5] 2	[1.5, 2.5, 3.5, 4.5]
[1, 2, 3, 4, 5] 3	[2.0, 3.0, 4.0]

• การวิเคราะห์ปัญหา

• Input:		จำนวนข้อมูล	ชนิดข้อมูล	
• Output:	์ (แสดงค่า)	จำนวนข้อมูล	ชนิดข้อมูล	2 0 De
	(คืนค่า)	จำนวนข้อมูล	ชนิดข้อมูล	and the state of t

4) 4 คะแนน (Hw08_2_6xxxxxxxx.py) ให้เขียนฟังก์ชัน median_of_median(list_a) เพื่อคืนค่ากลางของ element ใน list_a ซึ่งเป็น List ของ<u>จำนวนจริง</u> โดยคำนวณค่ากลางดังกล่าวด้วยวิธี Median of Median การคำนวณ Median of Median เป็นการคำนวณค่ากลางแบบ recursive โดยมีเงื่อนไขคือหาก list_a มีความ ยาว<u>อย่างน้อย</u> 3 จะต้องแบ่ง list_a เป็นกลุ่มย่อย 3 กลุ่มเท่าๆ กัน (นำเศษจากการแบ่งไปไว้ที่กลุ่มสุดท้าย) แล้ว คำนวณค่ามัธยฐานในแต่ละกลุ่มย่อยก่อน จากนั้นจึงนำค่าที่ได้มาหาค่ามัธยฐานอีกที โดยกำหนดให้ค่ามัธยฐานของ List ที่มีความยาว 2 element คือค่าเฉลี่ยของทั้งสอง element นั้น ๆ ดังรูปด้านล่าง ทั้งนี้<u>ไม่อนุญาต</u>ให้ใช้ฟังก์ชัน sorted() ในการแก้ปัญหา



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

[28,	14,	13,	21,	19,	27,	23,	30,	21.0	
16,	3]								

• การวิเคราะห์ปัญหา

• Input:		จำนวนข้อมูล	ชนิดข้อมูล	
• Output:	(แสดงค่า)	จำนวนข้อมูล	ชนิดข้อมูล	
L)	(คืนค่า)	จำนวนข้อมล	ชนิดข้อมล	O 102

5) 4 คะแนน (HW08_3_6XXXXXXX.py) ให้เขียนฟังก์ชัน medal_allocation(list_a) เพื่อคืนค่าการกำหนด เหรียญรางวัลในรูป Tuple of Lists ให้กับการแข่งขันเขียนโปรแกรม "โค้ดดิ้งโอลิมเป็ด" (Coding Olympaid) โดย พิจารณาจากคะแนนของผู้เข้าแข่งขันใน list_a ที่เป็น List ของ<u>จำนวนเต็มตั้งแต่ศูนย์ขึ้นไป</u> โดยจะแบ่งประเภทของ เหรียญเป็น Gold, Silver และ Bronze เหรียญละ 1 รางวัล ทั้งนี้ในแต่ละ List ที่คืนค่าจะเป็นคะแนนของผู้เข้าแข่งขัน ที่ได้เหรียญ เรียงตามลำดับจาก Gold, Silver และ Bronze และเป็น List ว่างหากไม่มีผู้ได้เหรียญในประเภทนั้น ๆ

กรณีมีผู้เข้าแข่งขันคะแนนเท่ากันและเข้าข่ายได้เหรียญรางวัล ก็จะได้รับเหรียญทุกคนและหักจากโควตาเหรียญ ประเภทที่ต่ำกวาแทน เช่น หากมีผู้ได้คะแนนสูงสุดเท่ากันสองคน ก็จะแจก 2 เหรียญทอง และไม่แจกเหรียญเงิน แต่ จะข้ามไปแจกเหรียญทองแดง หรือหากมีผู้เข้าแข่งขันได้คะแนนสูงสุด 5 คน ก็จะแจกเหรียญทอง 5 เหรียญ และงด ให้เหรียญประเภทอื่น ๆ ทั้งนี้ในการแข่งขันจะมีผู้เข้าแข่งขันไม่น้อยกว่า 3 คนเสมอ และจะ<u>ไม่พิจารณา</u>รางวัลให้ผู้เข้า แข่งขันที่ได้ 0 คะแนน

<u>Input</u>	<u>Output</u>
[9, 8, 7, 6, 5, 4, 3, 2]	([9], [8], [7])
[9, 8, 7, 7, 6, 5, 4, 3, 2]	([9], [8], [7, 7])
[9, 9, 8, 7, 6, 5, 4, 3, 2]	([9, 9], [], [8])
[9, 9, 9, 9, 8, 7, 6, 5, 4, 3, 2]	([9, 9, 9, 9], [], [])

• การวิเคราะห์ปัญหา

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

Chiang Mai University

การ<u>ส่งงาน</u>

- 1. ลักษณะ/ลำดับข้อความของการรับค่า/แสดงผล จะ**ต้องเป็นไปตามที่ร**ะบในตัวอย่างการ 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