

ชื่อ-นามสกุล.....รหัสนักศึกษา.....ตอนเรียน.....ลำดับที่.....
กำหนดส่ง



กระบวนวิชา **229223**

Lab	
HW	
Until	

การบ้านปฏิบัติการ 7

1D Lists and Tuples Part II (20 คะแนน)

ข้อกำหนด

- การเรียกใช้ฟังก์ชันเพื่อการทดสอบ ต้องอยู่ภายใต้เงื่อนไข `if __name__ == '__main__':` เพื่อให้สามารถ import ไปเรียกใช้งานจาก Script อื่น ๆ ได้
 - ไม่อนุญาตให้ใช้การทำซ้ำ `for`, `while` (Iterations), Recursions, หรือ Data Type อื่น ๆ ที่ยังไม่สอนในบทเรียน เช่น `set` หรือ `dict` ในการแก้ปัญหา
 - นักศึกษาสามารถสร้างฟังก์ชันย่อยต่าง ๆ เพิ่มเติมได้ตามความเหมาะสม
 - ให้ศึกษาการสร้างฟังก์ชันทดสอบ และการเขียน `main()` จาก template ในสัปดาห์ก่อนๆ และหัดเขียนฟังก์ชันดังกล่าวเองในในสัปดาห์นี้และสัปดาห์ถัดๆ ไป
- 1) 4 คะแนน (Lab07_1_6XXXXXXX.py) ให้เขียนฟังก์ชัน `corner_frame(n)` ($n \geq 2$) เพื่อคืนค่า String แทนรูปสี่เหลี่ยมผืนผ้าแสดงในรูปด้านล่าง โดยระหว่างตัวเลขจะต้องคั่นด้วย อักขระ space (' ') หนึ่งตัวเสมอและจะต้องจบแต่ละบรรทัด (รวมถึงบรรทัดสุดท้าย) ด้วย newline character ('\n')

Input

Output

4	<pre> 1 2 3 4 2 2 3 4 3 3 3 4 4 4 4 4 </pre>
7	<pre> 1 2 3 4 5 6 7 2 2 3 4 5 6 7 3 3 3 4 5 6 7 4 4 4 4 5 6 7 5 5 5 5 5 6 7 6 6 6 6 6 6 7 7 7 7 7 7 7 7 </pre>

การวิเคราะห์ปัญหา

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

ชื่อ-นามสกุล.....รหัสนักศึกษา.....ตอนเรียน.....ลำดับที่.....

- 2) 4 คะแนน (Lab07_2_6XXXXXXX.py) ให้เขียนฟังก์ชัน `uniform(line)` เพื่อคืนค่า ผลลัพธ์จากการแปลง String `line` ให้อยู่ในรูปตัวพิมพ์ใหญ่หรือพิมพ์เล็กทั้งหมด ขึ้นอยู่กับว่าชนิดตัวอักษรแบบใดมีมากกว่า หากมีจำนวนเท่ากัน ให้ยึดชนิดของตัวอักษรตัวแรกในคำเป็นเกณฑ์ แล้วแปลงอักษรทั้งหมดให้เป็นชนิดนั้น โดยพิจารณาเฉพาะ อักษรที่เป็นตัวอักษรในภาษาอังกฤษ (a-z และ A-Z)

Input	Output
HaPpY	HAPPY
cOdING	coding
coMP sci!!!	comp sci!!!

• การวิเคราะห์ปัญหา

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

- 3) 4 คะแนน (HW07_1_6XXXXXXX.py) ให้เขียนฟังก์ชัน `square_frame(n, sep=' ')` ($n \geq 3$) เพื่อแสดงผล กรอบสี่เหลี่ยมจัตุรัสตั้งแสดงด้านล่าง โดยจะต้องมีการ zero-padding ให้ตัวเลขมีความยาวหลักเท่ากันเสมอและ user สามารถระบุอักขระที่ใช้คั่น (`sep`) ระหว่างตัวเลขได้

Function Call	Output
<code>square_frame(3)</code>	1 2 3 8 4 7 6 5
<code>square_frame(4, ' .')</code>	01.02.03.04 12.....05 11.....06 10.09.08.07

• การวิเคราะห์ปัญหา

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

ชื่อ-นามสกุล.....รหัสนักศึกษา.....ตอนเรียน.....ลำดับที่.....

- 4) 4 คะแนน (HW07_2_6XXXXXXX.py) ให้เขียนฟังก์ชัน `medal_allocation(list_a)` เพื่อคืนค่าการกำหนดเหรียญรางวัลในรูป Tuple of Lists ให้กับการแข่งขันเขียนโปรแกรม “โค้ดดิ้งโอลิมเปียด” (Coding Olympiad) โดยพิจารณาจากคะแนนของผู้เข้าแข่งขันใน `list_a` ที่เป็น List ของจำนวนเต็มตั้งแต่ศูนย์ขึ้นไป โดยจะแบ่งประเภทของเหรียญเป็น Gold, Silver และ Bronze เหรียญละ 1 รางวัล ทั้งนี้ในแต่ละ List ที่คืนค่าจะเป็นคะแนนของผู้เข้าแข่งขันที่ได้เหรียญ เรียงตามลำดับจาก Gold, Silver และ Bronze และเป็น List ว่างหากไม่มีผู้ได้เหรียญในประเภทนั้น ๆ
- กรณีมีผู้เข้าแข่งขันคะแนนเท่ากันและเข้าข่ายได้เหรียญรางวัล ก็จะได้รับเหรียญทุกคนและหักจากโควตาเหรียญประเภทที่ต่ำกว่าแทน เช่น หากมีผู้ได้คะแนนสูงสุดเท่ากันสองคน ก็จะแจก 2 เหรียญทอง และไม่แจกเหรียญเงิน แต่จะข้ามไปแจกเหรียญทองแดง หรือหากมีผู้เข้าแข่งขันได้คะแนนสูงสุด 5 คน ก็จะแจกเหรียญทอง 5 เหรียญ และงดให้เหรียญประเภทอื่น ๆ ทั้งนี้ในการแข่งขันจะมีผู้เข้าแข่งขันไม่น้อยกว่า 3 คนเสมอ และจะไม่พิจารณารางวัลให้ผู้เข้าแข่งขันที่ได้ 0 คะแนน

Input	Output
[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
	(คืนค่า)	ชนิดข้อมูล

- 5) 4 คะแนน (HW07_3_6XXXXXXX.py) ให้เขียนฟังก์ชันเพื่อคืนค่าคำอ่านในภาษาอังกฤษของจำนวนเต็มตามข้อกำหนดดังนี้

- a. 2 คะแนน ให้เขียนฟังก์ชัน `three_digits_to_word(n)` เพื่อทำการคืนค่าคำอ่านในภาษาอังกฤษของจำนวนเต็ม n ($0 < n \leq 999$)
- b. 2 คะแนน ให้เขียนฟังก์ชัน `num_to_word(num)` เพื่อทำการคืนค่าคำอ่านของจำนวนเต็ม num ($0 \leq num$) ความยาวไม่เกิน 12 หลัก โดยจะต้องเรียกใช้ฟังก์ชัน `three_digits_to_word()` จากข้อ a.

Hint:

- สามารถศึกษาการอ่านตัวเลขในภาษาอังกฤษได้จาก http://en.wikipedia.org/wiki/English_numerals
- พิจารณาเรียกใช้ฟังก์ชัน `divmod()` $\xrightarrow{\text{input } x=8, y=3} (2, 2) \xrightarrow{\text{output}} x=3, y=2 \xrightarrow{\text{output}} (1, 1)$
- พิจารณาการใช้ `list` ในการแปลงตัวเลขให้เป็นคำอ่าน ดังแสดงด้านล่าง

```
>>> unit_list = ["", "one", "two", "three", "four", "five",
                "six", "seven", "eight", "nine", "ten",
                "eleven", "twelve", "thirteen", "fourteen", "fifteen",
                "sixteen", "seventeen", "eighteen", "nineteen"]
>>> unit_list[17]
'seventeen'
```

ชื่อ-นามสกุล.....รหัสนักศึกษา.....ตอนเรียน.....ลำดับที่.....

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

- การวิเคราะห์ปัญหา

- Input:

- Output:

(แสดงค่า)

(คืนค่า)

จำนวนข้อมูล

ชนิดข้อมูล

จำนวนข้อมูล

ชนิดข้อมูล

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/gdr223>

```
def three_digits_unit_list(n): # เก็บ list
    if divmod(n, 10)
```

COMPUTER SCIENCE
Chiang Mai University