



กระบวนวิชา 204111

HW+Lab	
D+2	

## การบ้านปฏิบัติการ 12

*n*-Dimensional Lists (20 คะแนน)

## ข้อกำหนด

- การเรียกใช้ฟังก์ชันเพื่อการทดสอบ ต้องอยู่ภายใต้เงื่อนไข `if __name__ == '__main__':` เพื่อความสะดวกในการ import จาก Script อื่น ๆ
- ไม่อนุญาตให้ใช้ Data Type อื่น ๆ ที่ยังไม่สอนในบทเรียน เช่น `dict` หรือ `set` ในการแก้ปัญหา
- สามารถใช้ Iteration หรือ Recursion แก้ปัญหาได้

**Hint:** ควรใช้ Statement `assert` เพื่อทำการทดสอบฟังก์ชันที่เขียนกับข้อมูลทดสอบหลายๆ ชุดโดยอัตโนมัติ

- 4 คะแนน (Lab12\_1\_6XXXXXXX.py) ให้เขียนฟังก์ชัน `calculate_score(id_list, ex1, ex2, ex3)` เพื่อประมวลผลคะแนนสอบนักศึกษาภาควิชาการคอมพิวเตอร์ชั้นปีที่หนึ่ง ในมหาวิทยาลัยแห่งหนึ่งทางภาคเหนือ โดยจะมีการสอบทั้งหมด 3 ครั้ง แต่ละครั้งมีคะแนนเต็ม 100 คะแนน เพื่อที่จะเลือกคะแนนสูงสุดสองครั้ง แล้วเก็บคะแนนรวม 10 % ของคะแนนเก็บทั้งภาคการศึกษา (ครั้งละ 5%)

โดย `id_list` จะเป็น list ของสายอักขระเก็บรหัสนักศึกษาทั้งหมดและ `ex1`, `ex2` และ `ex3` จะเป็น list คะแนนสอบ ของการสอบครั้งที่ 1 2 และ 3 ตามลำดับ ทั้งนี้ ในการสอบแต่ละครั้ง หากมีนักศึกษาขาดสอบ ช่องคะแนนของนักศึกษาดังกล่าวจะมีค่าเป็น None

ฟังก์ชันจะคืนค่าเป็น list ของ tuple โดยแต่ละ element จะเป็น tuple ของรหัสนักศึกษาและคะแนนเก็บ 10% ดังแสดงด้านล่าง

	Input	Output
<code>id_list</code>	<code>['6XXX10111', '6XXX10112', '6XXX10113']</code>	<code>[('6XXX10111', 5.75),</code>
<code>ex1</code>	<code>[53.57, 64.04, 88.53]</code>	<code>('6XXX10112', 6.34),</code>
<code>ex2</code>	<code>[59.69, None, 91.22]</code>	<code>('6XXX10113', 8.99)]</code>
<code>ex3</code>	<code>[55.36, 62.84, 84.45]</code>	

ทั้งนี้ ให้รับ input ผ่านฟังก์ชัน `main()` ด้วยวิธี command redirection โดยไฟล์ input จะมีรูปแบบเป็นไฟล์ *n* บรรทัด และแต่ละบรรทัด จะอยู่ในรูปแบบ รหัสนักศึกษา และ คะแนนสอบทั้ง 3 ครั้งของนักศึกษาคนนั้นดังแสดงด้านล่าง

6XXX10111	53.57	59.69	55.36
6XXX10112	64.04	None	62.84
6XXX10113	88.53	91.22	84.45
6XXX10114	89.46	None	None
6XXX10115	6.72	80.57	20.36
6XXX10116	12.74	41.2	22.23

- Input:                      จำนวนข้อมูล \_\_\_\_\_ ชนิดข้อมูล \_\_\_\_\_
- Output:                  (แสดงค่า)            จำนวนข้อมูล \_\_\_\_\_ ชนิดข้อมูล \_\_\_\_\_  
                                (คืนค่า)                จำนวนข้อมูล \_\_\_\_\_ ชนิดข้อมูล \_\_\_\_\_

### Output

[[2, 3, 4], [1, 2, 3]]	[[2, 3, 4], [1, 2, 3], [0, 0, 0]]
[[1, 2], [1, 2, 3], [1, 2], [1, 2], [1]]	[[1, 2, 0, 0, 0], [1, 2, 3, 0, 0], [1, 2, 0, 0, 0], [1, 2, 0, 0, 0], [1, 0, 0, 0, 0]]

- Input:                      จำนวนข้อมูล \_\_\_\_\_ ชนิดข้อมูล \_\_\_\_\_
- Output:        (แสดงค่า)     จำนวนข้อมูล \_\_\_\_\_ ชนิดข้อมูล \_\_\_\_\_  
                                 (คืนค่า)        จำนวนข้อมูล \_\_\_\_\_ ชนิดข้อมูล \_\_\_\_\_

- 3) 4 คะแนน (HW12\_1\_6XXXXXXX.py) ให้เขียนฟังก์ชัน `multiply_polynomials(p1, p2)` เพื่อคืนค่า list แทนผลคูณของ  $p1$  และ  $p2$  ซึ่งเป็น list แทน Polynomial (พหุนาม) ตัวอย่างเช่น  $[2, 3, 0, 4]$  คือ  $2x^3 + 3x^2 + 4$  ดังนั้น `multiply_polynomials([2, 0, 3], [4, 5])` จะแทนผลคูณ  $(2x^2 + 3)(4x + 5) = 8x^3 + 10x^2 + 12x + 15$  และคืนค่า  $[8, 10, 12, 15]$  ทั้งนี้ฟังก์ชันจะต้องไม่เปลี่ยนแปลง list  $p1$  และ  $p2$

**Input****Output**

$[2, 0, 3]$ $[4, 5]$	$[8, 10, 12, 15]$
-------------------------	-------------------

- การวิเคราะห์ปัญหา

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

- 4) 4 คะแนน (HW12\_2\_6XXXXXXX.py) ให้เขียนฟังก์ชัน `spiral(n)` เพื่อคืนค่า list สองมิติขนาด  $n \times n$  ในรูปแบบดังแสดง โดย list ที่ได้จะมีการเรียงตัวเลขตั้งแต่ 1 ถึง  $n^2$  ลงในตารางเริ่มจาก 1 ที่ช่องซ้ายบนและเรียงค่าเวียนแบบก้นหอยตามเข็มนาฬิกาจากนอกไปในจนสิ้นสุดที่  $n^2$

1	2	3	4	5	6
20	21	22	23	24	7
19	32	33	34	25	8
18	31	36	35	26	9
17	30	29	28	27	10
16	15	14	13	12	11

1	→	→	→	→	↓
20→	→	→	→	↓	↓
↑	32→	→	↓	↓	↓
↑	↑	36←	←	↓	↓
↑	↑	←	←	←	↓
↑	←	←	←	←	←

**Input****Output**

5	$[[1, 2, 3, 4, 5],$ $[16, 17, 18, 19, 6],$ $[15, 24, 25, 20, 7],$ $[14, 23, 22, 21, 8],$ $[13, 12, 11, 10, 9]]$
---	---

5) 4 คะแนน (HW12\_3\_6XXXXXXX.py) ให้เขียนฟังก์ชันแบบ `sum_nested_list(list_a)` เพื่อคืนค่าผลรวมของจำนวนเต็มทั้งหมดใน `list_a` โดยแต่ละสมาชิกของ `list_a` มีชนิดข้อมูลที่เป็นไปได้ 2 ประเภท คือ เป็นจำนวนเต็ม (int) หรือเป็น list โดย list ที่เป็นสมาชิกดังกล่าว ก็สามารถมีสมาชิกเป็นจำนวนเต็มและ list ได้เช่นกัน ตัวอย่างเช่น

```
[1, 2, [[3, 0], 4], 8]
```

จากตัวอย่างเป็น list ที่มี 4 สมาชิก โดยสมาชิก ที่ 0, และ 1 และ 3 ของ list มีชนิดเป็นจำนวนเต็ม ในขณะที่ สมาชิกที่ 2 มีชนิดเป็น list: `[[3, 0], 4]` และผลรวมของจำนวนเต็มทั้งหมดจะมีค่า  $1 + 2 + 3 + 0 + 4 + 8 = 18$

**Hint:**

- สามารถเรียกใช้ฟังก์ชัน `isinstance(object, classinfo)` เพื่อตรวจสอบชนิดของสมาชิก เช่น `isinstance([3], list)` จะคืนค่าเป็น `True`
- สามารถแก้ปัญหาได้ด้วย iteration, recursion หรือวิธีอื่น ๆ ตามอัธยาศัย

<u>Input</u>	<u>Output</u>
[1, 2, [[2, [[145], 34]], [48, 22]]]	254
[58, [31, [90]]]	179
[61, [[2, [75]], 8000, [39]], [58, [46]]]	8281
[81, [[31, [159]], 9577, [22, [181, [41]]]]]	10092

- 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>