



Lab	
HW	
Until	

การบ้านปฏิบัติการ 12

Iteration Part II (20 คะแนน)

ข้อกำหนด

- i. การเรียกใช้ฟังก์ชันเพื่อการทดสอบ ต้องอยู่ภายใต้เงื่อนไข `if __name__ == '__main__':` เพื่อให้สามารถ `import` ไปเรียกใช้งานจาก Script อื่น ๆ ได้อย่างเป็นมาตรฐาน
 - ii. ทุกข้อต้องมีการสร้างฟังก์ชัน `my_id()` โดยให้คืนค่าสายอักขระแทนเลขประจำตัวนักศึกษา 9 หลัก
- 1) 4 คะแนน (Lab12_1_6XXXXXXXX.py) ให้เขียนฟังก์ชัน `display_calendar(month, year)` เพื่อแสดงปฏิทินในเดือนและปีคริสต์ศักราชตามปฏิทินกริกอเรียน (Gregorian Calendar) ที่ระบุด้วยตัวแปร `month` และ `year` โดยให้ศึกษา Algorithm การคำนวณวันในสัปดาห์ (อาทิตย์, จันทร์, อังคาร, ... เสาร์) เมื่อทราบวันเดือนปีจาก https://en.wikipedia.org/wiki/Zeller%27s_congruence#Formula ทั้งนี้ไม่อนุญาตให้ใช้ module `datetime` หรือโมดูลอื่น ๆ ที่ไม่ได้สอนในบทเรียนในการแก้ปัญหา

ข้อกำหนด: หลังแสดงผลวันสุดท้ายของเดือนฟังก์ชันจะต้องแสดงอักขระ new line ('\n') ทันที

<u>Function Call</u>	<u>Output</u>
display_calendar(2, 2023)	Su Mo Tu We Th Fr Sa -- -- -- 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Hint: สามารถใช้ คำสั่ง `cal` ใน bash shell (ตัวอย่างเช่น <https://replit.com/languages/bash>) เพื่อตรวจสอบ output ที่ถูกต้อง เช่น `cal 2 2023`

- การวิเคราะห์ปัญหา

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

- 2) 4 คะแนน (Lab12_2_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

- 3) 4 คะแนน (HW12_1_6XXXXXXX.py) ให้เขียนฟังก์ชัน `scramble(word)` เพื่อคืนค่า list ของคำที่เป็นไปได้ทั้งหมดจากการสลับอักษรใน `str word` (ไม่มีอักขระอื่นนอกจากตัวอักษรภาษาอังกฤษ) โดย list ที่ได้จะต้องไม่มีสมาชิกซ้ำกัน ทั้งนี้ไม่อนุญาตให้ `import` โมดูลเพิ่มเติมในการแก้ปัญหา

Input**Output**

'Cat'	$['Cat', 'Cta', 'aCt', 'atC', 'tCa', 'taC']$
'bee'	$['bee', 'ebe', 'eeb']$
'bEe'	$['Ebe', 'Eeb', 'bEe', 'beE', 'eEb', 'ebE']$

หมายเหตุ: ลำดับของผลลัพธ์ภายใน list ที่คืนค่าไม่จำเป็นต้องเหมือนตัวอย่าง

- การวิเคราะห์ปัญหา

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

Chiang Mai University

- 4) 4 คะแนน (HW12_2_6XXXXXXXXX.py) [Attachments] ลูฟี่ และกลุ่มหมวกฟางได้พบสมุดบันทึกเก่าแก่ซึ่งมีตำแหน่งของสมบัติทั้งหมดในโลก ในสมุดที่พบมีข้อมูลเกี่ยวกับตำแหน่งของสมบัติ (location) ประเภทของสมบัติ (treasure type) และมูลค่าของสมบัติ (value) ลูฟี่และสมาชิกกลุ่มหมวกฟางต้องการทราบมูลค่ารวมของสมบัติประเภทใดประเภทหนึ่งเช่น มูลค่ารวมของทองจากทุกจุดในแผนที่
- หน้าที่ของคุณคือต้อง เขียนฟังก์ชันดังต่อไปนี้:

- a) ฟังก์ชัน `read_input()` เพื่ออ่านข้อมูลสมบัติจากสมุดบันทึกด้วยวิธี command redirection และคืนค่า **dict** ที่มี **key** เป็นประเภทของสมบัติ โดยไฟล์หนึ่งๆ จะแทน 1 หน้าของสมุดบันทึกซึ่งมีรูปแบบดังนี้
- บรรทัดแรกสุดจะขึ้นต้นด้วยอักขระ '#' เสมอและแสดงข้อความแทนเลขหน้าของสมุดบันทึก และข้อมูลอื่น ๆ ซึ่งฟังก์ชันจะต้องไม่นำบรรทัดนี้มาใช้คำนวณ
 - บรรทัดถัดๆ ไปมีรูปแบบดังนี้:
`<location>, <treasure_type>, <value>`
 - จะไม่มีบรรทัดไหนที่เหมือนกันทั้งตำแหน่ง (location) และ ประเภทของสมบัติ (treasure type)

โดย **dict** ที่คืนค่าจะมีรูปแบบคือ มี **key** เป็น **treasure_type** (ประเภทของสมบัติที่พบ) และ **value** เป็น **list** ของ **tuple** ในรูปของ (location, value) หรือ (ตำแหน่ง, มูลค่า)

Input: 'page28.txt'

Output:

# page 28 - year 1400 Marineford, Platinum, 42000 Marineford, Diamond, 300 Enies Lobby, Bronze, 43000 Elbaf, Silver, 9600 Dressrosa, Gold, 190 Punk Hazard, Gold, 900	{ "Platinum": [("Marineford", 42000)], "Diamond": [("Marineford", 300)], "Bronze": [("Enies Lobby", 43000)], "Silver": [("Elbaf", 9600)], "Gold": [("Dressrosa", 190), ("Punk Hazard", 900)] }
---	--

- b) ฟังก์ชัน `total_value(treasure_type, treasures)` เพื่อคืนค่ามูลค่ารวมที่คำนวณตามประเภทของสมบัติที่ระบุด้วยตัวแปร **treasure_type** เมื่อ **treasures** คือ **dict** ที่ได้จาก output ของฟังก์ชัน `read_input()` และฟังก์ชันจะคืนค่า -1 หากไม่พบประเภทของสมบัติตามที่ระบุด้วยตัวแปร **treasure_type**

Function Call

Output:

<code>total_value('Gold', treasures)</code>	1090
<code>total_value('Ruby', treasures)</code>	-1

- การวิเคราะห์ปัญหา

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



- 5) 4 คะแนน (HW12_3_6XXXXXXX.py) กติกาเบื้องต้นของเกม Minesweeper คือผู้เล่นจะได้รับ Board ที่ประกอบด้วยกระเบื้องที่และตัวเลขที่ถูกซ่อนไว้ ผู้เล่นใช้ตัวเลขดังกล่าวเพื่อเดาตำแหน่งของกระเบื้อง แต่ละช่องบนกระดานสามารถเป็นช่องว่าง หรือมีตัวเลขแสดงว่ามีกระเบื้องที่ถูกล้อมรอบ ๆ ช่องนั้น (0 - 8) หรือมีกระเบื้องซ่อนอยู่ข้างใต้ เป้าหมายคือการล้างกระดานให้ได้ โดยเปิดช่องที่ไม่มีระเบิดทั้งหมดโดยไม่ทำให้เกิดการระเบิด (จากการคลิกโดนกับระเบิด)

คุณเป็นนักพัฒนาซอฟต์แวร์ตัวตั้งในบริษัท "Minesweeper Unlimited" ในตำนาน ซึ่งเป็นที่รู้จักจากการสร้างเกม Minesweeper วันหนึ่ง ทีมของคุณได้รับคำขอพิเศษจากผู้ที่ชื่นชอบการเล่น Minesweeper มาก เขาต้องการสร้าง board แบบกำหนดเองสร้างจาก List ของ Tuple แสดง พิกัดระเบิด โดยขอให้คุณคำนวณตัวเลขที่แสดงระเบิดล้อมรอบช่องใด ๆ และคืนค่าเป็น Python Dictionary ที่มี **key** เป็นพิกัดช่องที่มีระเบิดล้อม และ **value** เป็นจำนวนระเบิด เช่น ในตารางขนาด 3x3 ถ้า input คือ [(0, 1), (1, 2), (2, 0), (2, 1), (2, 2)] เมื่อ (i, j) หมายถึง row ที่ i และ column ที่ j Board ที่ได้จะมีรูปร่างดังนี้

	[0]	[1]	[2]
[0]		💣	
[1]			💣
[2]	💣	💣	💣

ตัวเลขแสดงจำนวนระเบิดที่ถูกต้องจึงเป็น

	[0]	[1]	[2]
[0]	1	💣	2
[1]	3	5	💣
[2]	💣	💣	💣

หรือในรูป Python Dictionary {(0, 0): 1, (0, 2): 2, (1, 0): 3, (1, 1): 5}

หน้าที่ของคุณคือให้เขียนฟังก์ชัน `ms_mine_hint(m, n, bomb_list)` เพื่อคืนค่า Dictionary แสดงพิกัดของตัวเลขแสดงจำนวนระเบิดดังที่บรรยายมา โดยให้ `m` และ `n` คือจำนวน row และจำนวน column ของ Board และ `bomb_list` คือ List ของ Tuple แสดงตำแหน่งระเบิด โดยให้คืนค่าพิกัดของช่องที่มีจำนวนระเบิดล้อมรอบมากกว่า 0 เท่านั้น ทั้งนี้สามารถศึกษากติกาได้จากการลองเล่นเกม Minesweeper Online คลายเครือข่ายที่

<https://minesweeper.online/>

Function Call

Output:

<code>ms_mine_hint(3, 3, [(0, 1), (1, 2), (2, 0), (2, 1), (2, 2)])</code>	<code>{(0, 0): 1, (0, 2): 2, (1, 0): 3, (1, 1): 5}</code>
---	---

การวิเคราะห์ปัญหา

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

