				. , .
ชื่อ-นามสกล	รหัสนักศึกษา	ตอนเรียน	ลำดับที่	
9			กำหนดส่ง	



Lab	
HW	
Until	

การบ้านปฏิบัติการ 11

Collections and I/O Redirections (20 คะแนน)

ข้อกำหนด

- i. การเรียกใช้ฟังก์ชันเพื่อการทดสอบ ต้องอยู่ภายใต้เงื่อนไข **if** __name__ == '__main__' : เพื่อให้สามารถ import ไปเรียกใช้งานจาก Script อื่น ๆ ได้
- ii. นักศึกษาสามารถสร้างฟังก์ชันย่อยต่าง ๆ เพิ่มเติมได้ตามความเหมาะสม และเลือกใช้การแก้ปัญหาได้ทั้งวิธี iteration หรือ recursion
- 1) **4 คะแนน** (Lab11_1_6XXXXXXX.py) ให้เขียนฟังก์ชัน word_count(*text*) เพื่อคืนค่า dictionary ผลลัพธ์จาก การนับจำนวนคำที่ปรากฏในสายอักขระ (String) *text* โดยฟังก์ชันจะ คืนค่าเป็น dict ที่มี *key* เป็นแต่ละคำที่ ปรากฏใน *text* และมี *value* เป็นความถี่ ทั้งนี้ตัวอักษรที่อยู่ใน *key* จะต้องเป็นตัวอักษรพิมพ์เล็กเท่านั้น ข้อกำหนด
 - การนับความถี่จะเป็นแบบ Case Insensitive ('ant' และ 'Ant' ถือเป็นคำเดียวกัน)
 - ข้อความในไฟล์จะเป็นภาษาอังกฤษมาตรฐานในรูปแบบที่ถูกต้อง (well-formed English)
 - ไม่พิจารณาเครื่องหมายวรรคตอนต่าง ๆ <u>เฉพาะที่ล้อมรอบคำ</u> เช่น !"#\$%&'()*+,./:;<=>?@[\]^_`{|}~
 - เนื่องจาก key มีคุณสมบัติเป็น set ลำดับในการแสดงผลใน output จึงไม่จำเป็นต้องเหมือนตัวอย่าง

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

"He doesn't want to pay \$40,000 for a new car, but his wife doesn't seem to care."	{'new': 1, 'but': 1, 'pay': 1, 'want': 1,
	'seem': 1, 'care': 1,
COMPUTER	'his': 1, '40,000': 1, 'wife': 1, 'a': 1,
Chiang Mai	'for': 1, 'car': 1, "doesn't": 2,
	'to': 2, 'he': 1}

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

• การวิเคราะห์ปัญหา

- 2) 4 คะแนน (Lab11_2_6xxxxxxxx.py) [Attachments] ลูฟี่ ตัวละครหลักจากอนิเมะเรื่อง One Piece ได้พบสมุด บันทึกเก่าแก่ซึ่งมีตำแหน่งของสมบัติทั้งหมดในโลกของ One Piece ในสมุดที่พบมีข้อมูลเกี่ยวกับตำแหน่งของสมบัติ (location) ประเภทของสมบัติ (treasure type) และมูลค่าของสมบัติ (value) ลูฟี่และสมาชิกกลุ่มหมวกฟางต้องการ ทราบมูลค่ารวมของสมบัติประเภทใดประเภทหนึ่งเช่น 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)],
    "Bronze": [("Enies Lobby", 43000)],
    "Silver": [("Elbaf", 9600)],
    "Gold": [("Dressrosa", 190), ("Punk Hazard", 900)]
}
```

b) ฟังก์ชัน total_value(treasure_type, treasures) เพื่อคืนค่ามูลค่ารวมที่คำนวณตามประเภทของสมบัติที่ ระบุด้วยตัวแปร treasure_type เมื่อ treasures คือ dict ที่ได้จาก output ของฟังก์ชัน read_input() และ ฟังก์ชันจะ<u>คืนค่า -1</u> หากไม่พบประเภทของสมบัติตามที่ระบุด้วยตัวแปร treasure_type

Function Call

Output:

total_value('Gold', treasures)	1090
total value('Ruby', treasures)	-1

				• ,
الم	م به به	a		
A			9 -	
ชอ-นามสกล	รหสนกศกษา.	ตอนเรยน	ถ้าดบท	
ш — № 10N 611161	8 7 1 6 1 7 1 1 1 1 1 1 1 1 .	VI U M6 8 U M		

3) 4 คะแนน (HW11_1_6XXXXXXX.py) ให้เขียนฟังก์ชัน display_calendar(month, year) เพื่อแสดงปฏิทิน ในเดือนและปีคริสต์ศักราชตามปฏิทินกริกอเรียน (Gregorian Calendar) ที่ระบุด้วยตัวแปร month และ year โดย ให้ศึกษา Algorithm การคำนวณวันในสัปดาห์ (อาทิตย์, จันทร์, อังคาร, ... เสาร์) เมื่อทราบวันเดือนปีจาก https://en.wikipedia.org/wiki/Zeller%27s_congruence#Formula ทั้งนี้ไม่อนุญาตให้ใช้ module datetime ในการ แก้ปัญหา

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

Function Call	<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	

4) 4 คะแนน (HW11_2_5XXXXXXX.py) ให้เขียน) ให้เขียนฟังก์ชัน runner_up() เพื่อรับค่าคะแนนของนักศึกษา ในห้อง ตามจำนวนนักศึกษาที่ระบุในบรรทัดแรก แล้ว<u>แสดงผล</u>คะแนนที่สูงเป็นอันดับที่ 1, อันดับที่ 2 และ ค่าเฉลี่ย คะแนน (ทศนิยม 2 ตำแหน่ง) โดยหากไม่มีตำแหน่งที่ 2 ให้ แสดงคำว่า None ข้อกำหนด: ไม่อนุญาตให้ใช้ตัวแปรประเภท iteralbles เช่น list หรือ tuple ในการเก็บข้อมูลที่ได้รับเพื่อ การคำนวณ

Hint: ศึกษา Slide Input/Output Redirection เพื่อความสะดวกในการทดสอบข้อมูลนำเข้าจำนวนมาก

ตัวอย่างการ run 1

ตัวอย่างการ run 2

Total students: 3	CI: M:	Total students: 2
Enter score:		Enter score:
13		61
12		61
13		
		Max score is: 61.00
Max score is: 13.00		Runner up is: None
Runner up is: 12.00		Average is: 61.00
Average is: 12.67		

<u>ตัวอย่างการ run 3</u>

<u>ตัวอย่างการ run 4</u>

Total students: 7	Total students: 1
Enter score:	Enter score:
61	13
72	
64	Max score is: 13.00
81	Runner up is: None
61	Average is: 13.00
79	
63	
Max score is: 81.00	
Runner up is: 79.00	
Average is: 68.71	

5) **4 คะแนน** (HW11_3_6XXXXXXX.py) ให้เขียนฟังก์ชัน words_to_num(*words*) เพื่อทำการคืนค่า<u>จำนวนเต็ม บวก</u>ที่คำนวณได้จากคำอ่านของจำนวนเต็ม *words* โดยผลลัพธ์จะมีความยาวไม่เกิน 12 หลัก

<u>Hint</u>: สามารถศึกษาการอ่านตัวเลขในภาษาอังกฤษได้จาก http://en.wikipedia.org/wiki/English_numerals

InputOutputfourteen14two hundred forty-eight248one hundred eleven111forty-two billion six hundred forty-one
million three hundred twenty-three42641323862

• การวิเคราะห์ปัญหา

thousand eight hundred sixty-two

• Input: จำนวนข้อมูล_____ชนิดข้อมูล______
 • Output: (แสดงค่า) จำนวนข้อมูล______ชนิดข้อมูล_______
 (คืนค่า) จำนวนข้อมูล_______ชนิดข้อมูล_______

การ<u>ส่งงาน</u>

- 1. ลักษณะ/ลำดับข้อความของการรับค่า/แสดงผล จะ<u>ต้องเป็นไปตามที่ระบ</u>ุในตัวอย่างการ 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