| | | | | . , . |
|-------------|--------------|----------|----------|-------|
| ชื่อ-นามสกล | รหัสนักศึกษา | ตอนเรียน | ลำดับที่ | |
| q q | | | กำหนดส่ง | |



กระบวนวิชา <u>229223</u>

| Lab | |
|-------|--|
| HW | |
| Until | |

การบ้านปฏิบัติการ 13 *n*-Dimensional Lists (20 คะแนน)

ข้อกำหนด

- i. การเรียกใช้ฟังก์ชันเพื่อการทดสอบ ต้องอยู่ภายใต้เงื่อนไข **if** __name__ == '__main__' : เพื่อให้สามารถ import ไปเรียกใช้งานจาก Script อื่น ๆ ได้
- ii. นักศึกษาสามารถสร้างฟังก์ชันย่อยต่าง ๆ เพิ่มเติมได้ตามความเหมาะสม
- iii. ฟังก์ชัน main() ควรอ่าน Input จากไฟล์ด้วยวิธี Command Redirection เพื่อความสะดวกในการทดสอบและหา ข้อผิดพลาด
- 1) **4 คะแนน** (Lab13_1_5XXXXXXXX.py) [Attachment] ให้เขียนฟังก์ชัน matrix_mult(m1, m2) เพื่อทำการ หาผลคูณของเมทริกซ์ m1 และ เมทริกซ์ m2 (wikipedia: https://goo.gl/S0DDZv) โดยฟังก์ชันจะทำงานแบบ Non-destructive กล่าวคือจะ<u>คืนค่า</u>ผลคูณที่ได้โดยไม่เปลี่ยนแปลงเมทริกซ์ m1 และ m2 ที่อยู่ในในรูปแบบ List สองมิติ หากไม่สามารถหาผลคูณได้ให้คืนค่า None

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 7 & 8 \\ 9 & 10 \\ 11 & 12 \end{bmatrix} = \begin{bmatrix} 58 \\ \end{bmatrix}$$

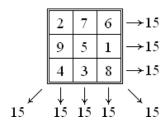
<u>Output</u>

| [[1, 2, 3], | [[58, 64], |
|--|--------------------------|
| [4, 5, 6]] | [139, 154]] |
| [[7, 8], [9, 10], [11,12]] | |
| [[1, 2, 3], | [[58, 64, 17, 29, 56], |
| [4, 5, 6]] | [139, 154, 41, 83, 131]] |
| [[7, 8, 5, 9, 3], [9, 10, -3, 7, 13], [11, 12, 6, 2, 9]] | University |

• การวิเคราะห์ปัญหา

• Input: จำนวนข้อมูล_____ชนิดข้อมูล______
 • Output: (แสดงค่า) จำนวนข้อมูล_____ชนิดข้อมูล______
 (คืนค่า) จำนวนข้อมูล______ชนิดข้อมูล______

2) **4 คะแนน** (Lab13_2_6XXXXXXXX.py) **[Attachment]** ให้เขียนฟังก์ชัน Boolean is_magic_square(board) เพื่อทดสอบว่า list 2 มิติ board เป็น จัตุรัสกล (Magic Square) หรือไม่



ในทางคณิตศาสตร์ จัตุรัสกล (magic square) ขนาด n คือการนำตัวเลขจำนวนเต็ม<u>ทุกตัว</u> ตั้งแต่ 1 ถึง n^2 (ไม่มี ตัวเลขซ้ำ) มาเรียงลงในตารางรูปสี่เหลี่ยมจัตุรัสขนาด $n \times n$ ซึ่งผลบวกของจำนวนในแต่ละแถว, แต่ละหลัก, แต่ละ แนวทแยงจะเท่ากันทั้งหมด ดังตัวอย่าง Magic Square ที่มี n=3 ในรูปด้านบน

| [[2, 7, 6], [9, 5, 1], [4, 3, 8]] | True |
|---|-------|
| [[5, 5, 5], [5, 5, 5], [5, 5, 5]] | False |

Output

True

• การวิเคราะห์ปัญหา

[[7, 12, 1, 14],

[2, 13, 8, 11], [16, 3, 10, 5], [9, 6, 15, 4]]

<u>Input</u>

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

3) 4 คะแนน (HW13_1_6XXXXXXX.py) [Attachment] ให้เขียนฟังก์ชัน square_matrix(list_x) เพื่อทำให้ List 2 มิติ list_x ที่มีสมาชิกเป็น<u>จำนวนเต็ม</u>กลายเป็น matrix จัตุรัสโดยเติม 0 เพื่อให้มีขนาด row และ column เท่ากัน โดยจะต้องคงทุก element ใน List เดิมไว้ และจำนวน 0 ที่เติมต้องเป็นจำนวนที่น้อยที่สุดที่เป็นไปได้ ทั้งนี้กำหนดให้ ฟังก์ชันทำงานแบบ Destructive และแต่ละ element จะต้องไม่เป็น alias ซึ่งกันและกัน

| <u>Input</u> |
|--------------|
|--------------|

| [[]] | FF2 2 43 |
|-------------|-------------|
| [[2, 3, 4], | [[2, 3, 4], |
| [1, 2, 3]] | [1, 2, 3], |
| | [0, 0, 0]] |

<u>Input</u>

Output

| [[1, 2], | [[1, 2, 0, 0, 0], |
|--------------------|--------------------------------------|
| [1, 2, 3], | [1, 2, 3, 0, 0], |
| [1, 2], [1, 2], | [1, 2, 0, 0, 0], |
| [1] | [1, 2, 0, 0, 0], [1, 0, 0, 0, 0]] |

• การวิเคราะห์ปัญหา

4) **5 คะแนน** (HW13_2_6XXXXXXX.py) [Attachment] ให้เขียนฟังก์ชัน <u>Destructive</u> simplified_m_sort(*list_x*, *show_step=False*) เพื่อทำการเรียงลำดับสมาชิกใน Integer List *list_x* ตามลำดับจากน้อยไปมากโดยใช้ Merge Sort Algorithm โดยมี Optional Parameter *show_step* เพื่อแสดง/ไม่ แสดงขั้นตอนในแต่ละ Iteration

Function Call

Outpu

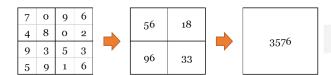
| <pre>list_x = [3, 7, 4, 9, 5, 2, 6] simplified_m_sort(list_x, True) print('') print(list_x)</pre> | [[3], [7], [4], [9], [5], [2], [6]] [[3, 7], [4, 9], [2, 5], [6]] [[3, 4, 7, 9], [2, 5, 6]] [[2, 3, 4, 5, 6, 7, 9]] [2, 3, 4, 5, 6, 7, 9] |
|---|---|
| <pre>list_x = [3, 7, 4, 9, 5, 2, 6, 1] simplified_m_sort(list_x) print('') print(list_x)</pre> | [1, 2, 3, 4, 5, 6, 7, 9] |

• การวิเคราะห์ปัญหา

• Input: จำนวนข้อมูล_____ชนิดข้อมูล_____

5) 4 คะแนน (HW13_3_6XXXXXXXX.py) [Attachment] ให้เขียนฟังก์ชัน sum_d_product(m) เพื่อ<u>คืนค่า</u> ผลบวกของผลคูณทแยงใน matrix m ที่มีขนาด $n \times n$ เมื่อ n สามารถเขียนในรูปของ 2^x (x เป็นจำนวนเต็มบวก)

ดังนั้น matrix 3 4 จะมี sum_d_product() = $(1 \times 4) + (3 \times 2) = 10$ กรณีต้องการหา sum_d_product() ของ matrix ขนาดใหญ่กว่า 2×2 ทำได้โดยการหา sum_d_product() ของ matrix ย่อย ขนาด 2×2 ก่อน แล้วหา sum_d_product() ของ matrix ผลลัพธ์อีกที เช่นกรณี matrix ขนาด 4×4 จะมีขั้นตอนดังนี้



หรือกรณี matrix 8 × 8

| 3 | 4 | 3 | 3 | 0 | 4 | 4 | 3 | 16 | 6 | 4 | 17 | | | |
|---|---|---|---|---|---|---|---|----|----|----|----|-----|-----|--------|
| 1 | 4 | 1 | 1 | 1 | 2 | 3 | 2 | 10 | U | 4 | 1/ | 324 | 500 | |
| 5 | 5 | 2 | 3 | 3 | 5 | 5 | 5 | 30 | 0 | 20 | 40 | 324 | 300 | |
| 1 | 5 | 1 | 3 | 1 | 5 | 3 | 5 | 30 | 9 | 20 | 40 | | | 408644 |
| 3 | 0 | 5 | 1 | 4 | 5 | 3 | 1 | 9 | 17 | 19 | 20 | | | 400044 |
| 5 | 3 | 2 | 3 | 3 | 1 | 5 | 5 | 9 | 1/ | 19 | | 376 | 681 | |
| 2 | 2 | 4 | 1 | 4 | 1 | 4 | 5 | 20 | 4 | 16 | 19 | 3/0 | 001 | |
| 5 | 5 | О | 1 | 0 | 4 | 3 | 1 | 20 | 4 | 10 | 19 | | | |

| <u>Input</u> | <u>Output</u> |
|---|---------------|
| [[3, 3, 3, 2], [2, 0, 3, 1], [2, 1, 2, 3], [1, 0, 2, -1] | 33 |
| [[1, 1, 5, -1], [12, 2, -2, 0], [4, 8, 8, 12], [4, 12, 12, 15]] | 3856 |
| [[0, -1, -1, 3, 2, 3, -1, 3], [3, -1, -1, 2, 0, -1, 2, 1], [3, 0, 1, 2, 3, 1, 3, 1], [2, 2, 1, -1, -1, 2, 0, 3], [1, 3, 2, 1, 3, 2, 2, 1], [1, 2, 2, 1, 3, 3, 1, 3], [2, 2, 2, 2, 2, 2, 3, 3], [1, 3, 2, 3, 1, 1, 2, 2]] | -6290 |

• การวิเคราะห์ปัญหา

| Input: | | จำนวนขอมูล | _ชนดขอมูล |
|-----------|-----------|------------------|------------------|
| • Output: | (แสดงค่า) | จำนวนข้อมูล | _ _ชนิดข้อมูล |
| | (คืนค่า) | • จำนวนข้อมูล | _ _ชนิดข้อมูล |