

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



กระบวนวิชา **229223**

Lab	
HW	
Until	

การบ้านปฏิบัติการ 4 Conditionals (20 คะแนน)

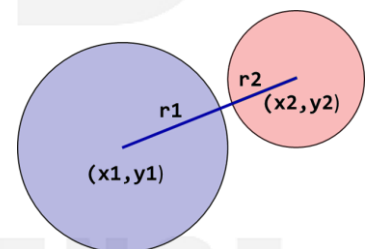
ข้อกำหนด

- การเรียกใช้ฟังก์ชันเพื่อการทดสอบ ต้องอยู่ภายใต้เงื่อนไข `if __name__ == '__main__':` เพื่อให้สามารถ import ไปเรียกใช้งานจาก Script อื่น ๆ ได้
- ไม่อนุญาตให้ใช้การทำซ้ำเช่น **for**, **while** (Iterations), Recursions, หรือ Data Type อื่น ๆ ที่ยังไม่สอนในบทเรียน เช่น **range**, **list** หรือ **map** ในการแก้ปัญหา
- นักศึกษาสามารถสร้างฟังก์ชันย่อยต่าง ๆ เพิ่มเติมได้ตามความเหมาะสม
- ในข้อที่ระบุว่ามี **[Attachments]** ให้ Download ไฟล์ Template จาก Grader ลงมา implement

Hint: ควรสร้างฟังก์ชันทดสอบ เพื่อทำการทดสอบฟังก์ชันที่เขียนกับข้อมูลทดสอบหลายๆ ชุดโดยอัตโนมัติ โดยใช้ Statement assert ตามที่ได้เรียนในบทเรียน

- 4 คะแนน (Lab04_1_6XXXXXXX.py) **[Attachments]** ให้เขียนฟังก์ชัน `circle_intersect(x1, y1, r1, x2, y2, r2, epsilon=10**-6)` เพื่อคำนวณว่าวงกลมสองวง ที่มีจุดศูนย์กลางที่ Coordinate (x_1, y_1) และ (x_2, y_2) และมีรัศมี r_1 และ r_2 ตามลำดับ สัมผัสกัน (Touching) ตัดกัน (Intersecting) หรือ ไม่ตัดกัน (Non-intersecting) โดยหาส่วนที่ใกล้ที่สุดของเส้นรอบวงของวงกลมทั้งสอง ห่างกันไม่เกินค่า `epsilon` ให้ถือว่าวงกลมทั้งสองสัมผัสกัน ทั้งนี้ระยะห่างระหว่างสองจุดใด ๆ (Distance) สามารถหาได้จากสูตร

$$distance = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$



โดยฟังก์ชันจะมีการคืนค่าดังนี้

- **1** กรณีวงกลมสองวงตัดกัน (intersecting)
- **0** กรณีวงกลมสองวงสัมผัสกัน (touching)
- **-1** กรณีวงกลมสองวงไม่ตัดและไม่สัมผัสกัน (non-intersecting)

Hint: พิจารณาศึกษาฟังก์ชัน `almost_equal()` หรือ `math.isclose()` จาก slide เรื่อง Conditionals Part I

Input	Output
2 3 5 5 7 1	1
0 0 2.5 3 4 2.5	0

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

Pythagorean triple คือสามเหลี่ยมมุมฉากที่มีความยาวด้านทั้ง 3 เป็นจำนวนเต็ม เช่น 3 4 5 หรือ 5 12 13

<u>Input</u>	<u>Output</u>
4 5 3	True
1 1 2	False

- | | | | |
|-----------|-----------|-------------------|------------------|
| • Input: | | จำนวนข้อมูล _____ | ชนิดข้อมูล _____ |
| • Output: | (คี่นค่า) | จำนวนข้อมูล _____ | ชนิดข้อมูล _____ |
| • Output: | (แสดงค่า) | จำนวนข้อมูล _____ | ชนิดข้อมูล _____ |

b. (HW04_1_6XXXXXXXXX.py) เขียนฟังก์ชันในภาษา python ตาม Algorithm ที่ออกแบบไว้

<u>Input</u>	<u>Output</u>
1	min = 1
2	mid = 2
3	max = 3

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

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

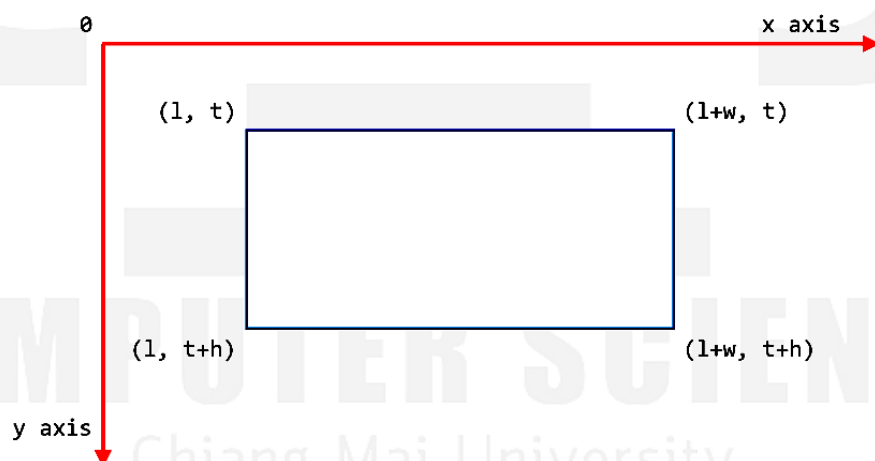
- 4) 4 คะแนน (HW04_2_6XXXXXXX.py) [Attachments] ให้เขียนฟังก์ชัน `min_diff(hour1, min1, period1, hour2, min2, period2)` เพื่อคืนค่าระยะห่างเป็นนาทีของเวลาที่ระบุด้วยจำนวนเต็ม `hour1, min1` และ `hour2, min2` ($1 \leq \text{hour}_x \leq 12$ และ $0 \leq \text{min}_x \leq 59$) โดย `hourx` และ `minx` จะแทนเวลาเป็นนาฬิกา และ นาทีตามลำดับ และตัวแปร `period1` และ `period2` เป็น string ระบุช่วงเวลาก่อนหรือหลังเที่ยงในรูป 'AM' และ 'PM' ทั้งนี้ให้ถือว่าเวลาที่ระบุเป็นเวลาที่อยู่ในวันเดียวกัน และไม่อนุญาตให้ใช้ module `datetime` ในการแก้ปัญหา
- Hint:** นักศึกษาสามารถศึกษาการระบุช่วงเวลากับ AM และ PM ได้จาก https://en.wikipedia.org/wiki/12-hour_clock

Function Call	Output
<code>min_diff(8, 23, 'AM', 8, 24, 'AM')</code>	1
<code>min_diff(8, 23, 'AM', 1, 24, 'PM')</code>	301
<code>min_diff(1, 24, 'PM', 8, 23, 'AM')</code>	301

• การวิเคราะห์ปัญหา

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

- 5) 4 คะแนน (HW04_3_6XXXXXXX.py) [Attachments] ให้เขียนฟังก์ชัน Boolean `is_overlapped(l1, t1, w1, h1, l2, t2, w2, h2)` เพื่อตรวจสอบว่าสี่เหลี่ยมมุมฉากสองรูปมีส่วนทับ (Overlap) กันหรือไม่ โดยที่เราสามารถนิยามสี่เหลี่ยมมุมฉากดังนี้



โดย t คือ top, l คือ left, w คือ width และ h คือ height ของรูปสี่เหลี่ยม

Hint:

- พิจารณาเงื่อนไขกรณีสี่เหลี่ยมที่ไม่ทับกันจะแก้ปัญหาได้ง่ายกว่า
- พิจารณาวาดรูปตามลักษณะการทับหรือไม่ทับดังแสดงด้านล่าง แล้วใส่ตัวเลข coordinate เพื่อทำความเข้าใจสิ่งที่ต้องการตรวจสอบ และสร้างชุดทดสอบ

The diagram is divided into two horizontal sections. The top section, labeled 'True' on the left, contains six pairs of overlapping blue and red rectangles. The bottom section, labeled 'False' on the left, contains three pairs of non-overlapping blue and red rectangles. This visualizes that a prediction is 'True' only when the predicted bounding box (red) overlaps with the ground truth bounding box (blue).

- การวิเคราะห์ปัญหา
- | | |
|---------------------------|---------------------------------|
| • Input: | จำนวนข้อมูล_____ชนิดข้อมูล_____ |
| • Output: (คืนค่า) | จำนวนข้อมูล_____ชนิดข้อมูล_____ |
| • Output: (แสดงค่า) | จำนวนข้อมูล_____ชนิดข้อมูล_____ |

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>