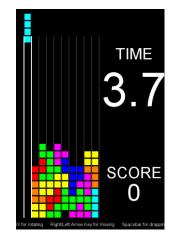
# Assignment 11: Tetris

เกมเตตริส (Tetris) เป็นเกมแนวแก้ปริศนาที่โด่งดังมาก ในเกม Tetris จะมีบล็อครูปทรงต่าง ๆ ที่ประกอบด้วยก้อนสี่เหลี่ยมจัตุรัสเล็ก ๆ เรียกว่า Tetrominoes รูปทรงเหล่านี้หล่นลงมาใน กระดานทีละชิ้น และเมื่อสามารถเติมเต็มทุกช่องได้ในหนึ่งแถว แถวนั้นจะหายไป โดยการ หายไปนี้สามารถหายไปพร้อมกันมากที่สุดได้ 4 แถวเรียกว่า Tetris

ในที่นี้เราจะมาลองเขียนเกม Tetris โดยใช้ library pygame กัน

เราแทนรูปทรงแต่ละรูปด้วย nested list ดังตัวอย่างข้างล่างนี้ (ค่าในลิสต์แทนสีของก้อน, 0 แทนไม่มีก้อน)

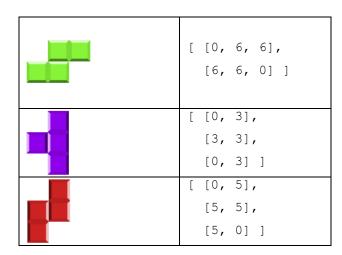


[ [1, 1, 1], [1, 0, 0] ]
[ [2, 2, 2], [0, 0, 2] ]
[ [3, 3, 3], [0, 3, 0] ]
[ [4, 4, 4, 4] ]

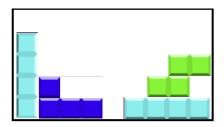
[ [5, 5, 0], [0, 5, 5] ]
[ [6, 0], [6, 6], [0, 6] ]
[ [7, 7], [7, 7] ]

ถ้ารูปทรงถูกหมุน nested list ที่แทนรูปทรงก็เปลี่ยนไปด้วย เช่น

[ [4], [4], [4], [4]]
[ [0, 2], [0, 2], [2, 2] ]
[ [2, 2], [2, 0], [2, 0] ]



และกระดานทั้งกระดานก็แทนด้วย nested list ในลักษณะเดียวกัน เช่น



```
[ [0, 0, 0, 0, 0, 0, 0, 0, 0],
        [4, 0, 0, 0, 0, 0, 0, 0],
        [4, 0, 0, 0, 0, 0, 0, 6, 6],
        [4, 2, 0, 0, 0, 0, 6, 6, 0],
        [4, 2, 2, 2, 0, 4, 4, 4, 4]]
```

#### งานของคุณ

จากโปรแกรมต้นฉบับที่มีให้ download จงเขียนชุดคำสั่งใน 4 ฟังก์ชันตามข้อกำหนดดังนี้

#### rotateR(shape)

- o รับ shape เป็น nested list (shape ก็คือรูปทรงต่าง ๆ ในเกม)
- o คืน nested list ที่เกิดจากการหมุนรูป shape ไปทางขวา 90 องศา

Input VOV function	Output VON function
shape = [[1,1,1], [0,1,0]]	[[0,1], [1,1], [0,1]]
shape = [[1,2,3,4]]	[[1], [2], [3], [4]]

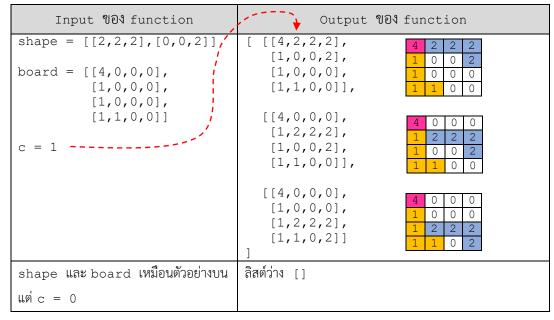
#### rotateL(shape)

- o รับ shape เป็น nested list (shape ก็คือรูปทรางต่าง ๆ ในเกม)
- o คืน nested list ที่เกิดจากการหมุนรูป shape ไปทางซ้าย 90 องศา

Input VON function	Output VON function
shape = [[1,1,1], [0,1,0]]	[[1,0], [1,1], [1,0]]
shape = [[1,2,3,4]]	[[4], [3], [2], [1]]

## animate\_drop(shape, board, c)

- o รับ shape และ board เป็น nested list, c เป็นจำนวนเต็ม
- o คืนลิสต์ของบอร์ด โดยบอร์ดแรกสุดเป็นบอร์ดเมื่อ shape ถูกวางไว้ที่บนสุดของ board และช่องซ้ายสุดของ shape อยู่ที่ช่องที่ c ของบอร์ด ส่วนบอร์ดถัด ๆ ไปจะเป็นบอร์ดเมื่อ shape ขยับลงมาทีละช่องจนไม่สามารถลง ต่อได้ แต่ถ้าไม่สามารถวาง shape ใน board ตอนเริ่มต้นได้ (เพราะมีอย่างอื่นขวางอยู่) ก็ต้องคืนลิสต์ว่าง



# animate\_clear(board)

- o รับ board เป็น nested list
- คืนลิสต์ของบอร์ด โดยบอร์ดแรกสุดเป็นบอร์ดที่เปลี่ยนทุกแถวที่ทุกค่าในแถวไม่เป็นศูนย์ ให้เป็นเป็นศูนย์ก่อน จากนั้น บอร์ดถัด ๆ ไปจะเป็นบอร์ดเมื่อขยับทุกแถวลงมาทีละช่องจนไม่สามารถขยับต่อได้ (แต่ถ้าไม่มีแถวที่ต้อง เปลี่ยนเป็น 0 หมด ก็คืน ลิสต์ว่าง)

เปลี่ยนเป็น 0 หมด กิคน สสตราง)	
Input VON function	Output VON function
board = [[3,3,0,0], [2,2,2,6], [0,5,5,4], [7,7,7,6], [0,0,0,0], [4,0,1,1]]	[ [[3,3,0,0],
	[[0,0,0,0], [3,3,0,0], [0,0,0,0], [0,5,5,4], [0,0,0,0], [4,0,1,1]],
	[[0,0,0,0], [0,0,0,0], [3,3,0,0], [0,0,0,0], [0,5,5,4], [4,0,1,1]],
	[[0,0,0,0], [0,0,0,0], [0,0,0], [3,3,0,0], [0,5,5,4], [4,0,1,1]]
board = [[3,0,0,0,2], [3,0,4,0,2], [4,4,4,0,2]]	[]
board = [[0,0,0,0,0],	[ [[0,0,0,0,0],
	[[0,0,0,0,0], [0,0,0,0,0], [2,0,0,0,0]]

## อย่าลืม

- ก่อนจะทำโจทย์ได้ ต้องติดตั้ง package ที่ชื่อ pygame
  (วิธีติดตั้ง pygame ใน Thonny เหมือนการบ้านที่ต้องติดตั้ง pandas แต่ในการบ้านนี้ชื่อ pygame)
- ห้ามเขียนคำสั่งในฟังก์ชันที่ไปใช้ตัวแปรที่อยู่นอกฟังก์ชัน (หรือที่เรียกว่าตัวแปร global)
- อนุญาตให้เพิ่มคำสั่งในบริเวณพื้นที่สีเขียวที่กำหนดเท่านั้น (สามารถเขียนฟังก์ชันเพิ่มเติมในบริเวณนี้ได้)
- ตั้งชื่อแฟ้ม ให้ถูกต้องตามที่เขียนใน CourseVille
- เพิ่มเติมข้อมูลใน comment ต้นโปรแกรมให้ตรงตามความจริง
- ก่อนส่งโปรแกรม ควรสั่งทำงานโปรแกรมที่ส่งอีกครั้ง ว่าทำงานได้ตามที่ต้องการ

# ์ข้อแนะนำ

• ถ้าต้องการ copy ลิสต์ซ้อนลิสต์ x ให้ใช้คำสั่ง copy.deepcopy(x) (ต้อง import copy ก่อน)

```
Prog-11: Tetris
# Fill in your ID & Name
                                                 def rotateR(shape):
                                                                                 ไม่ใช้ตัวแปรใด ๆ
# Declare that you do this by yourself
                                                     return ???
                                                                                  ที่อยู่นอกฟังก์ชัน
import pygame
                                                 def rotateL(shape):
import copy
                       ไม่เพิ่ม ลบ หรือ เปลี่ยนแปลง
import random
                        บริเวณคำสั่ง สีแดง เด็ดขาด
def make shape():
                                                     return ???
    # เขียนให้แล้ว
                                                 def animate drop(shape, board, c):
def top(board):
                                                     return ???
    # เขียนให้แล้ว
                                                 def animate clear (board):
def scoring(board):
    # เขียนให้แล้ว
                                                      return ???
def show text(x,y,size,text,screen):
    # เขียนให้แล้ว
                                                 pgame()
def pgame():
                                                              download code ฉบับเต็ม
    # เขียนให้แล้ว
```

# คำเตือน

• ข้อมูลของ shape ที่ใช้ทดสอบ ไม่จำเป็นต้องเหมือนกับ shape ในเกมที่ใช้กันทั่วไป เช่น

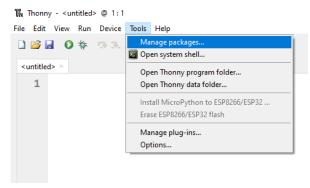




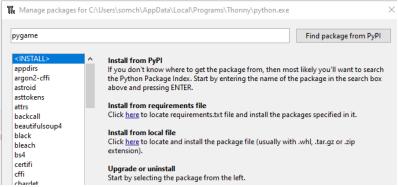


# วิธีติดตั้ง pygame

• ใน Thonny เลือกเมนู Tools -> Manage packages



• ใส่คำว่า pygame และกดปุ่ม Find package from PyPl



- แล้วก็กดปุ่ม install รอจนเสร็จ แล้วก็กดปุ่ม close
- ที่ Thonny เลือกเมนู Tools -> Open system shell... จะแสดงวินโดว์ดำ ๆ
  - o ถ้าใช้ Windows ก็ใส่คำสั่ง python -m pygame.examples.aliens แล้วกด enter
  - o ถ้าใช้ Mac OS ก็ใส่คำสั่ง python3 -m pygame.examples.aliens แล้วกด enter ถ้ามีเกมโผล่ขึ้นมาให้เล่น ก็แสดงว่าได้ติดตั้งสำเร็จ

