

Assignment 10: Tic-Tac-Toe

ทุกคนน่าจะรู้จักเกม Tic-Tac-Toe หรือที่รู้จักกันว่า เกมเอ็กซ์โอ/โอเอ็กซ์ เป็นเกมที่เล่นบนตารางขนาด 3×3 ผู้เล่นที่เล่นก่อนจะเขียนเครื่องหมาย O ในช่อง (หรือ X ก็ได้แล้วแต่) จากนั้นอีกคนหนึ่งก็จะเขียนเครื่องหมายตรงข้ามกับผู้เล่นก่อนหน้านั้นทำไปเรื่อย ๆ โดยผู้เล่นห้ามเขียนเครื่องหมายของตัวเองซ้ำช่องที่มีคนเขียนก่อนหน้านั้น ทั้งสองฝ่ายก็จะผลัดกันเขียนเครื่องหมายของตัวเอง เกมนี้จะมีผู้ชนะก็ต่อเมื่อมีผู้เล่นคนหนึ่งเขียนเครื่องหมายของตัวเองเรียงเป็นแนวตรงหรือแนวทแยงต่อกัน 3 อัน หรือเล่นกันจนถึงจุดที่รู้ว่า เล่นต่อก็ไม่มีใครแพ้หรือชนะได้ ก็ถือว่าเสมอกัน (<https://en.wikipedia.org/wiki/Tic-tac-toe>)

การบ้านนี้ให้เขียนฟังก์ชันสำคัญในโปรแกรม Tic-Tac-Toe โดยใช้ความรู้เรื่อง nested list เป็นการเล่นระหว่างผู้เล่น vs คอมพิวเตอร์ โดยผู้เล่นเป็น X และคอมพิวเตอร์เป็น O (ผู้เล่นเริ่มก่อนเสมอ) แต่ละกระดานจะแทนด้วยลิสต์ที่เก็บลิสต์ของเครื่องหมายในแต่ละแถว ดังตัวอย่างข้างล่างนี้ (การบ้านนี้จะไม่จำกัดกระดานที่ขนาด 3×3 มีขนาด N×N โดยที่ $N > 2$)

```
board = [['-', '-', '-'],
          ['- ', '- ', '- '],
          ['- ', '- ', '- ']]
```

	0	1	2
0	-	-	-
1	-	-	-
2	-	-	-

```
board = [['-', '-', '-'],
          ['- ', '- ', '- '],
          ['- ', 'X', '- ']]
```

	0	1	2
0	-	-	-
1	-	-	-
2	-	X	-

```
board = [['X', '-', '-'],
          ['O', '-', 'O'],
          ['- ', 'X', '- ']]
```

	0	1	2
0	X	-	-
1	O	-	O
2	-	X	-

```
board = [['X', 'O', '-', '-', '-'],
          ['- ', 'X', '-', 'O', '-'],
          ['O', '-', 'X', '-', '-'],
          ['- ', '-', 'O', '-', 'X'],
          ['- ', '-', '-', 'X', 'O']]
```

	0	1	2	3	4
0	X	O	-	-	-
1	-	X	-	O	-
2	O	-	X	-	-
3	-	-	O	-	X
4	-	-	-	X	O

งานของคุณ

ให้เขียนฟังก์ชันข้างล่างนี้ ในบริเวณสี่เหลี่ยม ตามข้อกำหนด

- `fill(board, r, c)`
 - ถ้าช่องที่ตำแหน่งแถวที่ `r` คอลัมน์ที่ `c` บน board เป็นช่องว่าง ให้เติม X ลงในช่อง และคืนค่าจริง
 - ถ้าไม่สามารถเติมในช่องนั้นได้ ให้คืนค่าเท็จ
- `check_win(board)` ตรวจสอบว่า สามารถสรุปอะไรได้ จากข้อมูลใน board
 - ถ้าสรุปได้ว่า X ชนะ ให้คืนค่า 'X'
 - ถ้าสรุปได้ว่า O ชนะ ให้คืนค่า 'O'
 - ถ้าสรุปได้ว่า เล่นต่อไปยังไงก็เสมอ ให้คืนค่า 'D' (เช่น ตัวอย่างกระดานขนาด 5×5 ที่แสดงข้างบน)
 - ถ้ายังสรุปอะไรไม่ได้ ให้คืนค่า '-'

```
# Prog-10: Tic-Tac-Toe
# Fill in your ID & Name
# ...
# Declare that you do this by yourself
```

```
import random
```

```
def main():
    N = int(input('Board size = '))
    board = [["-"]*N for j in range(N)]
    end = False
    print_board(board)
    while(not end):
        print("===== Player Turn =====")
        player_input(board)
        print_board(board)
        check = check_win(board)
        if(check != "-"):
            if(check == "D"):
                print("##### DRAW #####")
            else:
                print("\(^o^)/   YOU WIN !!!   \(^o^)/")
                break
        print("===== Computer Turn =====")
        com_fill(board)
        print_board(board)
        check = check_win(board)
        if(check != "-"):
            if(check == "D"):
                print("##### DRAW #####")
            else:
                print("(;-;)   YOU LOSE!!!   (-;-)")
                break
        print("Game has ended, thanks for playing :D")
```

```
def com_fill(board):
    N = len(board)
    new_board = [[x for x in y] for y in board]
    for i in range(N):
        for j in range(N):
            if board[i][j] == "-":
                new_board[i][j] = "O"
                if (check_win(new_board) == "O"):
                    board[i][j] = "O"
                    return
            new_board[i][j] = "X"
            if (check_win(new_board) == "X"):
                board[i][j] = "O"
                return
            new_board[i][j] = "-"
    while True:
        i = random.randint(0, N-1)
        j = random.randint(0, N-1)
        if board[i][j] == "-":
            board[i][j] = "O"
            break
```

ไม่เพิ่ม ลบ หรือ เปลี่ยนแปลง
บริเวณคำสั่ง สีแดง เด็ดขาด

```
def print_board(board):
    N = len(board)
    print(" "*3, end = "")
    for i in range(N):
        print((str(i)+ " "*(3))[:3], end = "")
        print()
    for row in range(N):
        print((str(row)+ " "*(3))[:3], end = "")
        for col in range(N):
            print(board[row][col], end = " ")
        print()
```

```
def player_input(board):
    f = False
    while not f:
        try:
            row = int(input("row = "))
            col = int(input("col = "))
            f = fill(board, row, col)
            if (not f):
                print("!!! You can't fill that spot !!!")
                print("---try again---")
        except:
            print("!!! Invalid Input !!!")
            print("---try again---")
```

```
#-----
```

```
def fill(board, r, c):
```

ไม่ใช่ตัวแปรใด ๆ
ที่อยู่นอกฟังก์ชัน

```
def check_win(board):
```

```
#-----
```

```
main()
```

[download code นี้ได้](#)

ปฏิบัติตามอย่างเคร่งครัด

- ห้ามเขียนคำสั่งในฟังก์ชันที่ใช้ตัวแปรที่อยู่นอกฟังก์ชัน (หรือที่เรียกว่าตัวแปร global)
- อนุญาตให้เพิ่มคำสั่งในบริเวณพื้นที่สีขาวเท่านั้น จะเขียนฟังก์ชันเพิ่มเติมก็ได้
- ตั้งชื่อแฟ้ม ให้ถูกต้องตามที่เขียนใน CourseVille
- เพิ่มเติมข้อมูลใน comment ต้นโปรแกรมให้ตรงตามความจริง
- ก่อนส่งโปรแกรม ควรสั่งทำงานโปรแกรมที่ส่งอีกครั้ง ว่าทำงานได้ตามที่ต้องการ