2559_2_NumPy_V4

ฟังก์ชัน **read_square_matrix()** ข้างล่างนี้ อ่านข้อมูลของ **square matrix** มาสร้าง **numpy array** แบบสองมิติ ดังตัวอย่างในตารางข้างล่างนี้

input	ผลที่ได้จาก **read_square_matrix()**
1 1 1 1 2 2 2 2 4 4 3 3 0 0 5 5	array([[1, 1, 1, 1], [2, 2, 2, 2], [4, 4, 3, 3], [0, 0, 5, 5]])

จงเขียนฟังก์ชันต่าง ๆ ที่เว้นว่างในโปรแกรมข้างล่างนี้ ที่มีข้อกำหนดของพารามิเตอร์ และผลลัพธ์ที่ได้ตามตารางนี้

function	input parameter	return value
min_in_each_row(m)	**m** คือ **square matrix**	**array** หนึ่งมิติเก็บค่าน้อยสุดของแต่ละแถวแนวนอน เช่น จากตัวอย่าง **input** ที่แสดงข้างบนนี้จะได้ผลคือ **array([1, 2, 3, 0])**
max_in_each_column(m)	**m** คือ **square matrix**	**array** หนึ่งมิติเก็บค่ามากสุดของแต่ละคอลัมน์ เช่น จากตัวอย่าง **input** ที่แสดงข้างบนนี้จะได้ผลคือ **array([4, 4, 5, 5])**
diff_of_sums_of_two_diags(m)	**m** คือ **square matrix**	ผลต่างระหว่างผลรวมของค่าในแนวทแยงมุมสองแนวของ **m** เช่น จากตัวอย่าง **input** ที่แสดงข้างบนนี้จะได้ผลคือ **4**
halve(m)	**m** คือ **square matrix** จำนวนแถวเป็นจำนวนคู่ **> 0**	ถ้า **m** มีขนาด **2n × 2n** จะได้ผลลัพธ์เป็นเมทริกซ์ขนาด **n × n** แต่ละช่อง **[i,j]**ของผลลัพธ์มาจากผลรวมในช่อง **[2i,2j],[2i+1,2j],[2i,2j+1],[2i+1,2j+1]** ของ **m** เช่น จากตัวอย่าง **input** ที่แสดงข้างบนนี้จะได้ผลคือ **array([[6, 6],** **[8, 16]])**

```
import numpy as np

def read_square_matrix():
    d = [int(e) for e in input().split()]
    m = [d]
    for k in range(len(d)-1):
        m.append([int(e) for e in input().split()])
    return np.array(m)

def min_in_each_row(m):
    _____________________________________  # หาวิธีเขียนแค่คำสั่งเดียว

def max_in_each_column(m):
    _____________________________________  # หาวิธีเขียนแค่คำสั่งเดียว

def diff_of_sums_of_two_diags(m):
    _____________________________________  # หาวิธีเขียนอย่างมากสองคำสั่ง
    _____________________________________

def halve(m):
    _____________________________________  # หาวิธีเขียนอย่างมากสองคำสั่ง
    _____________________________________

exec(input().strip())  # do not remove this line
```

ข้อมูลนำเข้า

คำสั่งในการทดสอบฟังก์ชันที่เขียน

ข้อมูลส่งออก

ผลที่ได้จากคำสั่งที่ป้อนเป็นข้อมูลนำเข้า

ตัวอย่าง

input	output (ทางจอภาพ)
`print(min_in_each_row(np.array([[1,2],[3,0]])))`	`[1 0]`
`print(max_in_each_column(np.array([[1,2],[3,0]])))`	`[3 2]`
`print(diff_of_sums_of_two_diags(np.array([[1,2],[3,0]])))`	`4`
`print(halve(np.array([[1,2],[3,0]])))`	`[[6]]`