

## แบบฝึกหัดปฏิบัติการคาบที่ 12: Problem Solving

ชื่อ-นามสกุล.....รหัสประจำตัวนักศึกษา.....

วันที่.....เดือน.....พ.ศ. 2564

Section.....

1. [4G] จากการประมวลผลระบบ 4G ที่ดูเด็ด บริษัทให้บริการโทรศัพท์มือถือมีแนวโน้มที่จะออกแพ็คเกจบริการที่ซับซ้อน โดยมีการระบุว่าถ้าใช้แบบเติมเงินแล้วโทรตอนกลางวันจะคิดนาทีละ 0.75 บาท แต่ถ้าโทรตอนกลางคืนจะคิด 1.25 บาท ส่วนแบบจ่ายรายเดือนมีให้เลือก 2 ทางเลือกคือแบบ 300 บาทต่อเดือนและแบบ 600 บาทต่อเดือน โดยแบบ 300 บาทต่อเดือนจะโทรได้ 500 นาทีถ้าเกินนั้นจะคิดค่าโทรนาทีละ 1.50 บาท ส่วนแบบ 600 บาทต่อเดือนจะโทรได้ 1200 นาทีถ้าเกินนั้นจะคิดค่าโทรนาทีละ 1.25 บาท

นักศึกษาต้องการประหยัดค่าใช้จ่ายมากที่สุดจึงได้ทำการบันทึกว่าในแต่ละสัปดาห์ตนเองโทรตอนกลางวันกี่นาทีและตอนกลางคืนกี่นาที โดยจดบันทึกข้อมูลการใช้โทรศัพท์นี้เป็นเวลา 4 สัปดาห์

จึงเขียนโปรแกรมรับค่าตัวเลขจำนวนการโทรตอนกลางวัน และตอนกลางคืน จากนั้นโปรแกรมจะพิมพ์เลข 1 ถ้าแบบเติมเงินมีค่าใช้จ่ายน้อยที่สุด พิมพ์เลข 2 ถ้าแบบ 300 บาทต่อเดือนมีค่าใช้จ่ายน้อยที่สุด และพิมพ์เลข ~~3~~ ถ้าแบบ 600 บาทต่อเดือนมีค่าใช้จ่ายน้อยที่สุด

**ข้อมูลนำเข้า** บรรทัดที่ 1 - 4 จำนวนการโทรตอนกลางวัน และตอนกลางคืน

**ข้อมูลส่งออก** พิมพ์เลขทางเลือกแพ็คเกจที่มีค่าใช้จ่ายน้อยที่สุด

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก	ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
100 100 100 100 100 100 100 100	3	50 20 60 70 40 30 50 50	2

~~2. [Distance] กำหนดจุดในระนาบสามมิติมีตัวอย่างการเก็บในรูปแบบต่อไปนี้~~

~~float points[][ ] = {{ 1, 0, 3}, { 1, 1, 1}, {4, 1, 1}, {2, 0.5, 9}, {3.5, 2, 1},  
{3, 1.5, 3}, { 1.5, 4, 2}, {5.5, 4, 0.5}};~~

~~จึงเขียนโปรแกรมเพื่อคำนวณระยะทางระหว่างจุดสองจุดในระนาบสามมิติที่มีระยะทางระหว่างจุดมากที่สุด 3 อันดับแรก โดยระยะทางระหว่างสองจุด  $(x_1, y_1, z_1)$  และ  $(x_2, y_2, z_2)$  คำนวณได้จาก~~

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

**ข้อมูลนำเข้า**

~~บรรทัดแรก ระบุนับจำนวนตัวเลข n~~

~~บรรทัดที่ 2 ถึง n+1 ระบุนับจุดในระนาบสามมิติ~~

**ข้อมูลส่งออก**

~~บรรทัดที่ 1 - 3 แสดงระยะทางระหว่างจุดมากที่สุด 3 อันดับแรก~~

<del>ตัวอย่างข้อมูลนำเข้า</del>	<del>ตัวอย่างข้อมูลส่งออก</del>
<del>8 -1 0 3 -1 -1 -1 4 1 1 2 0.5 9 3.5 2 1 3 1.5 3 -1.5 4 2 5.5 4 0.5</del>	<del>10.71 10.55 10.22</del>

3. [Visible Trees] มีต้นไม้ ความสูงต่าง ๆ กัน เรียงเป็นแนวเส้นตรง เมื่ออยู่ดูจาก เติบโตผ่านต้นไม้แต่ละต้น ใบไม้ที่กั้น ความสูงของแต่ละต้นเอาไว้ ตามลำดับ จากนั้นเมื่อมองย้อนกลับไป จะมีต้นไม้จำนวนหนึ่งเท่านั้น ที่สามารถมองเห็นได้ใน แนวเส้นตรงเดียวกัน เพราะต้นไม้ที่มีความสูงเท่ากันหรือต่ำกว่า จะถูกบดบัง จงหาว่า มีต้นไม้กี่ต้น ที่อยู่ดูจะสามารถ มองเห็นได้

ข้อมูลนำเข้า

บรรทัดแรกคือค่า  $n$  ( $1 \leq n \leq 10$ ) จำนวน test case

และในอีก  $n$  บรรทัดต่อมา แต่ละบรรทัดคือ หนึ่ง test case ซึ่งประกอบด้วย  $T$  ( $1 \leq T \leq 80$ ) ระบุจำนวนต้นไม้ และมีจำนวนเต็มบวกอีก  $T$  ค่า เป็นความสูงของต้นไม้แต่ละต้นที่บันทึกไว้ตามลำดับ

ข้อมูลส่งออก

แต่ละ test case ให้แสดง จำนวนต้นไม้ที่สามารถมองเห็นได้

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
3	1
12 1 2 3 4 5 6 7 8 9 10 11 12	3
8 2 13 6 1 7 2 1 3	4
5 15 10 10 9 8	

4. หน่วยสืบคดีพิเศษของประเทศแห่งหนึ่งต้องการค้นหาแหล่งกบดานของนักบวชรูปหนึ่ง โดยแหล่งที่พักของนักบวชรูปนี้มีความลับซับซ้อนเป็นพิเศษ บุคคลภายนอกไม่สามารถเข้าถึงได้โดยตรง ด้วยเหตุนี้หน่วยสืบคดีพิเศษจึงจำเป็นต้องอาศัยอากาศยานไร้คนขับ หรือโดรน ทำการถ่ายภาพบริเวณที่สนใจ โดยภายในภาพถ่ายจะปรากฏจำนวนคน ณ บริเวณที่กำหนด โดยเป็นรูปขนาด  $H \times W$  ช่อง ซึ่งหน่วยสืบคดีพิเศษต้องการหานักบวชจากรูปภาพนี้

ตัวอย่างของรูปขนาด  $4 \times 5$  แสดงเป็นตารางด้านล่างกำหนดตารางชื่อ A ตัวเลขในแต่ละช่องแสดงจำนวนคนที่อยู่ในช่องนั้น

5	1	2	10	4
4	30	3	0	100
3	25	10	4	10
3	20	4	8	5

ในการหาตำแหน่งของนักบวชเนื่องจากเป็นนักบวชที่มีความสำคัญจึงจำเป็นต้องมีคนที่อยู่รอบข้าง ดังนั้นจึงมีเงื่อนไข 3 ข้อดังนี้

- นักบวชจะปรากฏในบริเวณที่เป็น 2 ช่องติดกันพอดี
- สองช่องที่เป็นบริเวณที่มีนักบวชควรมีจำนวนคน ณ บริเวณนั้นต่างกันไม่เกิน 10
- เนื่องจากเป็นนักบวชที่มีความสำคัญจึงจำเป็นต้องมีคนที่อยู่รอบข้าง ตำแหน่งที่นักบวชอาศัยอยู่จึงน่าจะเป็นตำแหน่งที่มีจำนวนคน ณ บริเวณนั้นอยู่เป็นจำนวนมาก ก็คือต้องเป็นสองช่องที่มีผลรวมของจำนวนคน ณ บริเวณนั้นอยู่เป็นจำนวนมาก

จากตารางตำแหน่งที่ตรงตามเงื่อนไขคือ  $A[2][2]$  และ  $A[3][2]$

จึงเขียนโปรแกรมที่รับตารางแสดงตำแหน่งของนักบวช จากนั้นให้หาตำแหน่งมุมบนซ้ายของช่องที่น่าจะปรากฏนักบวชมากที่สุด โดยระบุแถวและคอลัมน์ช่องนั้น

ชื่อ-นามสกุล.....รหัสประจำตัวนักศึกษา.....

วันที่.....เดือน.....พ.ศ. 2564

ตอนเรียน Lab ที่.....

### ข้อมูลนำเข้า

~~บรรทัดแรก ระบุขนาดตาราง HxW~~

~~บรรทัดที่ 2 ถึง H+1 แสดงจำนวนคนในแถวที่ i โดยระบุเป็นจำนวนเต็มจำนวน W ตัว จำนวนที่ j จะเป็นจำนวนคนในช่องที่อยู่ในคอลัมน์ j~~

### ข้อมูลส่งออก

~~มีบรรทัดเดียว คือ มุมบนซ้ายของช่องที่น่าจะปรากฏนักบวชมากที่สุดโดยระบุแถวและคอลัมน์ของช่องนี้~~

<del>ตัวอย่างข้อมูลนำเข้า</del>	<del>ตัวอย่างข้อมูลส่งออก</del>	<del>ตัวอย่างข้อมูลนำเข้า</del>	<del>ตัวอย่างข้อมูลส่งออก</del>
<del>4 5</del>	<del>2 2</del>	<del>4 4</del>	<del>3 2</del>
<del>5 1 2 10 4</del>		<del>0 0 0 0</del>	
<del>4 30 3 0 100</del>		<del>0 0 0 0</del>	
<del>3 25 10 4 10</del>		<del>0 1 1 1</del>	
<del>3 20 4 8 5</del>		<del>1 1 0 0</del>	

~~5. [Line] เส้นตรงคือการนำจุดสองจุดใดมาเชื่อมต่อกันโดยเส้นตรงจะประกอบด้วยสมาชิกที่เป็นจุดจำนวน 2 จุด คือจุดที่เป็นจุดเริ่มต้นของเส้นตรง (begin) และจุดที่เป็นจุดสุดท้ายของเส้นตรง (end) โดยมีโครงสร้างดังนี้~~

```
typedef struct {  
    POINT begin;  
    POINT end;  
} LINE;
```

```
typedef struct {  
    int x;  
    int y;  
} POINT;
```

~~จงเขียนโปรแกรมโดยใช้ฟังก์ชันที่รับพารามิเตอร์ 2 ตัวที่มีชนิดข้อมูลเป็น POINT จากนั้นให้นำโครงสร้างดังกล่าวไปสร้างเป็นเส้นตรง (LINE) และคืนเป็นเส้นตรงออกมา หลังจากนั้นให้เขียนฟังก์ชันที่รับตัวแปรที่เป็น LINE เข้ามาในฟังก์ชันแล้วคืนเลข 1 2 หรือ 3 โดยที่~~

~~1 คือเส้นตรงที่มีลักษณะเป็นแนวตั้ง (Vertical)~~

~~2 คือเส้นตรงที่มีลักษณะเป็นแนวนอน (Horizontal)~~

~~3 คือเส้นตรงที่ไม่ถูกนิยาม (oblique)~~

~~โดย Vertical line คือ เส้นตรงที่มีจุด begin กับจุด end มีพิกัด x อยู่ตำแหน่งเดียวกัน~~

~~Horizontal line คือ เส้นตรงที่มีจุด begin กับจุด end มีพิกัด y อยู่ตำแหน่งเดียวกัน~~

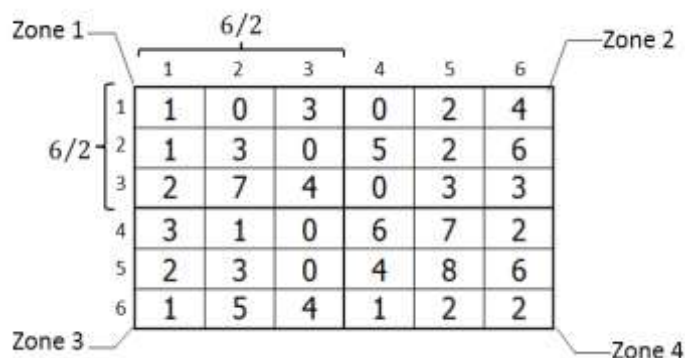
~~Oblique line คือ เส้นตรงที่ไม่เป็นทั้ง vertical line หรือ horizontal line~~

ชื่อ-นามสกุล.....รหัสประจำตัวนักศึกษา.....

วันที่.....เดือน.....พ.ศ. 2564

ตอนเรียน Lab ที่.....

6. [พื้นที่ในอาร์เรย์สองมิติ] อาร์เรย์ของเลขจำนวนเต็ม 2 มิติ ประกอบไปด้วย  $R$  แถว และ  $C$  คอลัมน์ โดยที่  $R$  และ  $C$  เป็นเลขคู่จำนวนเต็มบวก. ถ้าต้องการแบ่งพื้นที่ในอาร์เรย์นี้ออกเป็น 4 ส่วน ได้แก่ zone 1, 2, 3 และ 4 โดยที่แต่ละโซนจะประกอบไปด้วยพื้นที่สี่เหลี่ยมจัตุรัสขนาด  $R/2 \times C/2$  ของในอาร์เรย์ ตัวอย่างเช่น การแบ่งพื้นที่ของอาร์เรย์ขนาด 6 คูณ 6 แสดงได้ดังรูปด้านล่าง



จงเขียนโปรแกรมเพื่อแสดงผลบวกที่มากที่สุดของสมาชิกในแต่ละโซน (maximum total sum) ตัวอย่างเช่น ผลบวกของโซน 1 ในอาร์เรย์ด้านบน คือ  $1+0+3+1+3+0+2+7+4 = 21$  ขณะที่ผลบวกของอาร์เรย์ในโซน 4 คือ 36

ข้อมูลเข้า

บรรทัดแรกเป็นจำนวนเต็มบวก  $R$  และ  $C$

$R$  บรรทัดต่อมาเป็นตัวเลขในอาร์เรย์แต่ละแถว โดยแต่ละแถวมี  $C$  คอลัมน์

ข้อมูลส่งออก

ผลบวกที่มากที่สุดของสมาชิกในโซน

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลออก
<del>2 4</del> <del>1 2 3 4</del> <del>5 6 7 8</del>	15
<del>4 2</del> <del>1 2</del> <del>3 4</del> <del>5 6</del> <del>7 8</del>	14