
	<b>ICPC Pre-Contest 2017</b> <b>First Round</b>	
---	--	---

D	Hole	
	Time Limit	1 second
	Memory Limit	128 MB

คุณอาจจะเคยเห็นปัญหาเขาวนที่มีลักษณะแบบนี้

1234 -> 1	0671 -> 2	8888 -> 8
5678 -> 3	9966 -> 4	8496 -> ?

โจทย์ข้อนี้จะให้คุณเขียนโปรแกรมเพื่อตอบปัญหาเขาวนนี้ (หากยังคิดไม่ออก ให้ลองดูชื่อโจทย์ ถ้ายังคิดไม่ออกอีก เรามีเฉลยปัญหาเขาวนอยู่ในหน้าถัดไป)

#### ข้อมูลเข้า



มีบรรทัดเดียว เป็นจำนวนเต็ม  $N$  ประกอบด้วยตัวเลข 0 – 9 ไม่เกิน 50 หลัก

#### ข้อมูลออก

ให้ตอบปัญหาเขาวนที่ว่า  $N \rightarrow ?$

#### ตัวอย่าง

ข้อมูลเข้า	ข้อมูลออก
1234	1
5678	3
0671	2
9966	4
8888	8
1234567890	6

	<h1>ICPC Pre-Contest 2017</h1> <h2>First Round</h2>	
---	---	---

### เฉลยปัญหาเขาวน

ให้นับจำนวนรู (บริเวณปิด) ที่เกิดจากการเขียนตัวเลขนั้น ซึ่งก็คือ

‘1’, ‘2’, ‘3’, ‘5’, ‘7’ จะไม่มีรู

‘0’, ‘4’, ‘6’, ‘9’ มีตัวเลขละ 1 รู

และ ‘8’ มี 2 รู

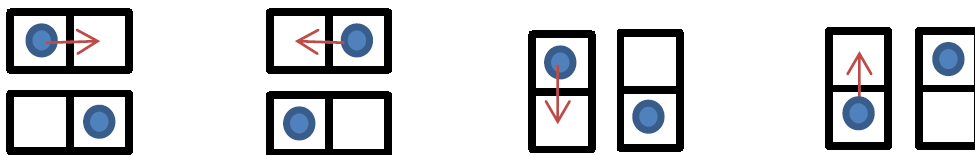
และนำจำนวนรูทั้งหมดมารวมกัน เช่น 0671 -> 2 เพราะมี 2 รูจากตัวเลข ‘0’ และ ‘6’

## มหาศึกย้ายลูกบอล (GameOfBalls)

สมมติให้เกมหนึ่งเล่นบนตารางขนาด  $M \times N$  ซึ่งมีเพียง  $S$  ช่องที่เป็นช่องว่าง และแต่ละช่องที่เหลือในตารางจะมีลูกบอลอยู่ช่องละหนึ่งลูก กำหนดให้เรียกพิกัดตำแหน่งของแต่ละช่องว่า  $(m,n)$  โดยที่  $1 \leq m \leq M$  และ  $1 \leq n \leq N$  ช่องในแถวบนสุดและล่างสุดมีค่า  $m$  เท่ากับ 1 และ  $M$  ตามลำดับ ช่องในหลักซ้ายสุดและขวาสุดมีค่า  $n$  เท่ากับ 1 และ  $N$  ตามลำดับ

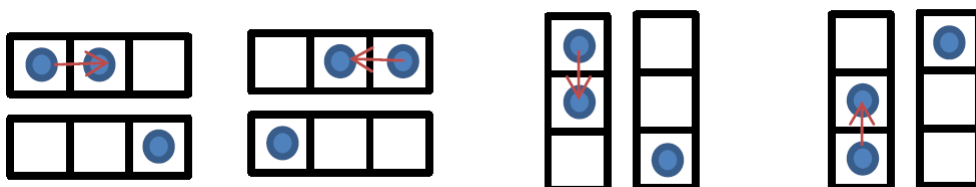
การเล่นแต่ละตาในเกมนี้จะทำได้เพียงสองวิธีเท่านั้น คือ

- 1) ระหว่างช่องสองช่องที่ติดกัน ช่องต้นทางต้องมีลูกบอลอยู่ไปยังช่องปลายทางซึ่งเป็นช่องว่างที่อยู่ติดกัน ทั้งนี้ช่องว่างอาจจะอยู่ด้านบน ด้านล่าง ด้านซ้าย หรือด้านขวาของช่องที่มีลูกบอลก็ได้ การเล่นเกมวิธีนี้จะย้ายลูกบอลจากช่องต้นทางไปยังช่องปลายทาง มีรูปแบบที่เป็นไปได้ 4 รูปแบบ ดังรูปที่ 1



รูปที่ 1 รูปแบบการย้ายลูกบอลระหว่างช่องสองช่องที่ติดกัน โดยแสดงให้เห็นก่อนและหลังการย้ายตามทิศของลูกศรสีแดง

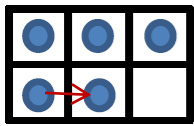
- 2) ระหว่างสามช่องที่อยู่ติดกันในแนวนอน หรือแนวตั้ง ช่องต้นทางและช่องตรงกลางจะต้องมีลูกบอลเสมอ ช่องปลายทางจะเป็นช่องว่าง โดยการเกมวิธีนี้จะย้ายลูกบอลข้ามจากช่องต้นทางที่อยู่ริมด้านหนึ่งไปยังช่องปลายทางที่อยู่ริมอีกด้านหนึ่ง และทำลายลูกบอลในช่องตรงกลางให้หายไป มีรูปแบบที่เป็นไปได้ 4 รูปแบบ ดังรูปที่ 2



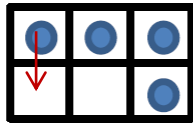
รูปที่ 2 รูปแบบการย้ายลูกบอลระหว่างสามช่องที่ติดกัน โดยแสดงให้เห็นก่อนและหลังการย้ายตามทิศของลูกศรสีแดง

ให้นักศึกษาเขียนโปรแกรมเพื่อรับข้อมูลการวางลูกบอลในตาราง และลำดับการย้ายลูกบอล  $K$  ตา แล้วคำนวณหาว่าเมื่อเล่นตาสุดท้ายเสร็จสิ้น จะเหลือจำนวนลูกบอลทั้งหมดกี่ลูก หากการเล่นในตาใดไม่เป็นไปตามวิธีข้างบน ให้ข้ามตาเล่นนั้นไปได้เลย

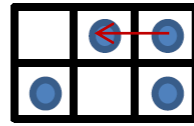
ยกตัวอย่างเช่น เกมที่มีตารางขนาด  $2 \times 3$  หลังการเดินครบทั้ง 8 ตา ดังแสดงในรูปที่ 3 จะเหลือลูกบอลเพียงลูกเดียว



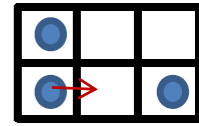
ตาที่ 1 (2,1) -> (2,3)



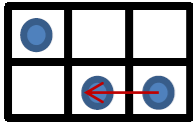
ตาที่ 2 (1,1) -> (2,1)



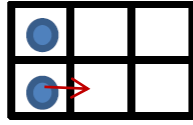
ตาที่ 3 (1,3) -> (1,1)



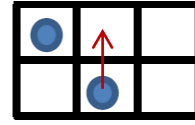
ตาที่ 4 (2,1) -> (2,2)



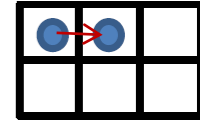
ตาที่ 5 (2,3) -> (2,1)



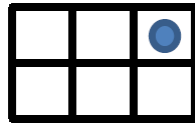
ตาที่ 6 (2,1) -> (2,2)



ตาที่ 7 (2,2) -> (1,2)



ตาที่ 8 (1,1) -> (1,3)



หลังเล่นตาสุดท้าย

รูปที่ 3 ตัวอย่างการเล่นเกมที่มีตารางขนาด  $2 \times 3$  ( $M = 2, N = 3, S = 1$ ) จากเริ่มต้น (มุมบนซ้าย)  
แสดงให้เห็นตารางหลังการเดินแต่ละตาตามลำดับ ทั้งหมด  $K = 8$  ตา จนเหลือลูกบอลเพียงลูกเดียว

### ข้อมูลเข้า

ข้อมูลเข้ามี  $S + K + 1$  บรรทัด ดังนี้

บรรทัดที่หนึ่ง ประกอบด้วยเลขจำนวนเต็มสามตัว ระบุค่า  $M, N, S$  และ  $K$  ตามลำดับ คั่นแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง  
กำหนดให้  $1 \leq M, N \leq 50, 1 \leq S \leq 2000$  และ  $1 \leq K \leq 100$

บรรทัดที่ 2 ถึง  $S + 1$  แต่ละบรรทัดประกอบด้วยเลขจำนวนเต็มสองตัว ระบุค่าพิกัดตำแหน่งของช่องว่างแต่ละช่อง คั่นแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง

บรรทัดที่  $S + 2$  ถึง  $S + K + 1$  แต่ละบรรทัดประกอบด้วยเลขจำนวนเต็มสี่ตัว ระบุค่าพิกัดตำแหน่งของช่องต้นทาง และค่าพิกัดตำแหน่งของช่องปลายทางในการย้ายลูกบอลแต่ละตา คั่นแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง

### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และ เซตของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

ข้อมูลส่งออกมีหนึ่งบรรทัด แสดงผลลัพธ์เป็นเลขจำนวนเต็มหนึ่งตัว ระบุจำนวนลูกบอลที่เหลือหลังจากการเล่นตาสุดท้าย

### ตัวอย่างที่ 1 (จากรูปที่ 3)

ข้อมูลเข้า	ข้อมูลส่งออก
2 3 1 8	1
2 3	
2 1 2 3	
1 1 2 1	
1 3 1 1	
2 1 2 2	
2 3 2 1	
2 1 2 2	
2 2 1 2	
1 1 1 3	

ตัวอย่างที่ 2 (บางส่วนจากรูปที่ 3)

ข้อมูลเข้า	ข้อมูลส่งออก
2 3 2 5	2
1 1	
2 2	
1 3 1 1	
2 1 2 2	
1 3 2 2	
1 1 1 3	
2 3 2 1	

หมายเหตุ ตารางที่ 3 และ 4 ในตัวอย่างนี้เป็นการเล่นที่ไม่เป็นไปตามวิธีที่ให้ในโจทย์ ดังนั้นสามารถข้ามไปได้เลย

ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรมดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

### สวัสดี (HelloWorld)

การทักทายอย่างเหมาะสมและสุภาพถือเป็นมารยาททางสังคมอันดี

นักศึกษาต้องเขียนโปรแกรมเพื่อรับข้อมูลนำเข้าเป็นชื่อต้น (first name) และนามสกุล (last name) ของผู้ใช้ ซึ่งเป็นข้อความจำนวน 1 ข้อความมีความยาวไม่เกิน 100 ตัวอักษร และข้อมูลตัวย่อของเพศ (gender) ของผู้ใช้ ซึ่งเป็นตัวอักษร 1 ตัว จากนั้นให้แสดงข้อความทักทายผู้ใช้อย่างเหมาะสม

#### ข้อมูลเข้า

ข้อมูลเข้ามีสองบรรทัด

บรรทัดแรกเป็นข้อความจำนวน 1 ข้อความ ความยาวไม่เกิน 100 ตัวอักษร ข้อความประกอบด้วยข้อมูล 2 ส่วน ส่วนแรกเป็นชื่อต้น (first name) และส่วนที่สองเป็นนามสกุล (last name) ของผู้ใช้ ทั้งสองส่วนคั่นด้วยตัวอักษรช่องว่าง (blank) อย่างน้อย 1 ตัว

บรรทัดที่สองเป็นตัวอักษรจำนวน 1 ตัว ซึ่งมีค่าที่เป็นได้ 2 ค่า ได้แก่ F และ M

**หมายเหตุ** กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามขอบเขตและเช็ตของค่าที่เป็นไปได้เสมอ

#### ข้อมูลส่งออก

ข้อมูลส่งออกมีหนึ่งบรรทัด แสดงข้อความทักทายผู้ใช้อย่างเหมาะสมตามเพศของผู้ใช้ ตามรูปแบบดังนี้

<b>Hello</b>	ตัวอักษรช่องว่าง 1 ตัว	คำนำหน้าหญิง ใช้ <b>Ms.</b> ชาย ใช้ <b>Mr.</b>	ตัวอักษรช่องว่าง 1 ตัว	นามสกุลของผู้ใช้	ตัวอักษรจุลภาค (ลูกน้ำ หรือ comma)	ตัวอักษรช่องว่าง 1 ตัว	ชื่อต้นของผู้ใช้
--------------	------------------------	--	------------------------	------------------	------------------------------------	------------------------	------------------

#### ตัวอย่างที่ 1

ข้อมูลเข้า	หมายเหตุ (ข้อมูลเข้า)	ข้อมูลส่งออก
Wanida Putt F	ตัวอักษรช่องว่าง 1 ตัว คั่นระหว่างชื่อต้นและนามสกุล	Hello Ms. Putt, Wanida

#### ตัวอย่างที่ 2

ข้อมูลเข้า	หมายเหตุ (ข้อมูลเข้า)	ข้อมูลส่งออก
Tham Rakkarndee M	ตัวอักษรช่องว่าง 3 ตัว คั่นระหว่างชื่อต้นและนามสกุล	Hello Mr. Rakkarndee, Tham

## ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
จำนวนชุดทดสอบ	10
คะแนนของแต่ละชุดทดสอบ	-
คะแนนเต็มทั้งสิ้น	-
เงื่อนไขในการรันโปรแกรม	โปรแกรมจะต้องประมวลผลชุดข้อมูลตัวอย่างที่ให้มาได้ถูกต้อง

## ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรกดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

## นาฬิกาทราย (Hourglass)



กำหนด A เป็นอาร์เรย์สองมิติขนาด  $n \times n$

นิยามของนาฬิกาทราย (Hourglass) ในอาร์เรย์ A คือ สมาชิกที่อยู่ในตำแหน่งที่เป็นแผ่นผั่งของนาฬิกาทราย ซึ่งประกอบด้วย 3 แถว และ 3 คอลัมน์ ดังนี้

a b c  
d  
e f g

ตัวอย่างเช่น อาร์เรย์ขนาด  $6 \times 6$  ดังแสดงในรูปที่ 1 จะมีแผ่นผั่งรูปนาฬิกาทรายได้ทั้งหมด 16 รูป

1	2	3	0	0	0
0	4	0	0	0	0
5	6	7	0	0	0
0	0	2	4	4	0
0	0	0	2	0	0
0	0	1	2	4	0

รูปที่ 1 แสดงข้อมูลของอาร์เรย์ขนาด  $6 \times 6$

ให้นักศึกษาเขียนโปรแกรมเพื่อคำนวณผลรวมของตัวเลขในนาฬิกาทรายแต่ละรูป และให้หาผลรวมที่มากที่สุด

ตัวอย่างเช่น อาร์เรย์ในรูปที่ 1 ประกอบด้วยแผ่นผั่งรูปนาฬิกาทราย 16 รูป ดังนี้

1	2	3	2	3	0	3	0	0	0	0	0
	4			0			0			0	
5	6	7	6	7	0	7	0	0	0	0	0
0	4	0	4	0	0	0	0	0	0	0	0
	6			7			0			0	
0	0	2	0	2	4	2	4	4	4	4	0
5	6	7	6	7	0	7	0	0	0	0	0
	0			2			4			4	
0	0	0	0	0	2	0	2	0	2	0	0
0	0	2	0	2	4	2	4	4	4	4	0
	0			0			2			0	
0	0	1	0	1	2	1	2	4	2	4	0

นาฬิกาทรายที่มีค่าผลรวมของตัวเลขมากที่สุด คือ 28

## ข้อมูลเข้า

ข้อมูลเข้ามี  $n + 1$  บรรทัด ดังนี้

- บรรทัดที่หนึ่ง ประกอบด้วยเลขจำนวนหนึ่งตัว แทนค่า  $n$  ซึ่งเป็นการระบุขนาดของอาร์เรย์  $n \times n$  โดยที่  $3 \leq n \leq 20$
- บรรทัดที่สอง ถึงบรรทัดที่  $n + 1$  ประกอบด้วยเลขจำนวนเต็ม  $n$  ตัว แต่ละตัวคั่นด้วยช่องว่างหนึ่งช่อง แทนค่าสมาชิกในอาร์เรย์ ใน 1 แถว ข้อมูลแต่ละตัวมีค่าอยู่ระหว่าง -9 ถึง 9



### หมายเหตุ

- กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และ เซ็ตของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

ข้อมูลส่งออกมีหนึ่งบรรทัด แสดงผลลัพธ์เป็นเลขจำนวนเต็มหนึ่งตัว บอกผลรวมของตัวเลขมากที่สุดของรูปแบบนาฬิกาทราย

### ตัวอย่างที่ 1

ข้อมูลเข้า	ข้อมูลส่งออก
6 1 2 3 0 0 0 0 4 0 0 0 0 5 6 7 0 0 0 0 0 2 4 4 0 0 0 0 2 0 0 0 0 1 2 4 0	28

### ตัวอย่างที่ 2

ข้อมูลเข้า	ข้อมูลส่งออก
4 0 1 1 1 0 0 1 0 0 1 1 1	7

### ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

## ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรกดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็น ชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

## อินเตอร์เซกชัน (Intersection)

บรรจทำงานในเทศบาลเมืองแห่งหนึ่ง เขากำลังสำรวจข้อมูลประชาชนเพื่อวางแผนการรณรงค์การใช้รถยนต์พลังงานไฟฟ้าในเมืองแห่งหนึ่งโดยเทศบาลจะรณรงค์โดยใช้การส่งจดหมายโฆษณาการใช้รถยนต์พลังงานไฟฟ้าไปที่บ้านประชาชน เพื่อลดจำนวนจดหมายที่ต้องส่ง บรรจต้องการรู้รหัสประจำตัวประชาชนที่เป็นกลุ่มเป้าหมาย โดยที่มันหมายความว่าประชาชนที่จะใช้รถยนต์พลังงานไฟฟ้าน่าจะมีอายุมากกว่า 30 ปี บรรจจึงขอข้อมูลรหัสประจำตัวของทุกคนในเมืองที่เป็นไปตามเงื่อนไข ซึ่งกำหนดให้ประชาชนกลุ่มนี้เป็นเซต  $X$  ในเวลาต่อมาบรรจรู้ภายหลังว่าเฉพาะประชาชนที่อยู่บริเวณใจกลางเมืองต้องการใช้รถยนต์พลังงานไฟฟ้าเนื่องจากรอบนอกของเมืองยังไม่มีสถานีชาร์จไฟฟ้ามากพอ บรรจจึงขอข้อมูลของประชาชนที่อาศัยบริเวณใจกลางเมือง โดยกำหนดให้ประชาชนกลุ่มนี้เป็นเซต  $Y$  และบรรจพบว่าจำนวนสมาชิกของ  $X$  มากกว่าจำนวนสมาชิกของ  $Y$  เขาต้องการทราบรหัสประจำตัวประชาชนของคนที่อยู่ในทั้งสองกลุ่มเพื่อจะทำการส่งจดหมายเชิญชวนในขั้นถัดไป

จงเขียนโปรแกรมเพื่อหารหัสประจำตัวประชาชนที่อยู่ทั้งสองเซต โดยเรียงรหัสประจำตัวจากน้อยไปมาก

### ข้อมูลเข้า

เซต  $X$  และ  $Y$  มีสมาชิกเป็นจำนวนเต็มบวก แต่ละจำนวนแทนรหัสประจำตัวประชาชนแต่ละคน

โดย  $M$  และ  $N$  คือขนาดของเซตทั้งสอง ซึ่ง  $M > N$

ข้อมูลเข้ามี 4 บรรทัด

- บรรทัดแรก ประกอบด้วยจำนวนเต็มบวก 1 จำนวน ได้แก่  $M$  โดยกำหนดให้  $1 \leq M \leq 100,000$
- บรรทัดที่สอง เป็นสมาชิกในเซต  $X$  ประกอบด้วยจำนวนเต็มบวกที่ไม่ซ้ำกัน  $x_i$  จำนวน โดย  $i$  มีค่าตั้งแต่ 1 ถึง  $M$  คั่นแต่ละจำนวนด้วยช่องว่าง โดย  $0 < x_i < 10,000,000$
- บรรทัดที่สาม ประกอบด้วยจำนวนเต็มบวก 1 จำนวน ได้แก่  $N$  โดยกำหนดให้  $1 \leq N \leq 10,000$
- บรรทัดที่สี่ เป็นสมาชิกในเซต  $Y$  ประกอบด้วยจำนวนเต็มบวกที่ไม่ซ้ำกัน  $y_i$  จำนวน โดย  $i$  มีค่าตั้งแต่ 1 ถึง  $N$  คั่นแต่ละจำนวนด้วยช่องว่าง โดย  $0 < y_i < 10,000,000$

### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และเซตของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้อง

ต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

ข้อมูลส่งออกมี 1 บรรทัด แสดงจำนวนเต็มบวกทั้งหมดที่เป็นคำตอบ โดยเรียงลำดับคำตอบจากน้อยไปมาก

### ตัวอย่างที่ 1

ข้อมูลเข้า	ข้อมูลส่งออก
8 5 2 10 4 8 1 7 9 7 10 1 13 7 2 6 3	1 2 7 10

## ตัวอย่างที่ 2

ข้อมูลเข้า	ข้อมูลส่งออก
14 12 43 33 8 14 2 16 42 19 15 25 37 1 22 9 38 3 17 25 14 12 44 31 39	12 14 25

### ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

### ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรมดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มีการสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

## สมุดงานวิชาโอเอส (OSWorkbook)

จอมแจ่มได้รับสมุดงานวิชาระบบปฏิบัติการจากอาจารย์ประจำวรรณ ซึ่งสมุดงานที่ได้รับนั้นเป็นโจทย์แบบฝึกหัดสำหรับบทเรียน  $N$  บท เพื่อให้นักศึกษาใช้เป็นแบบฝึกหัดทบทวนบทเรียน และจะมีโจทย์บางข้อที่จะถูกเลือกจากอาจารย์ประจำวรรณไปออกเป็นข้อสอบปลายภาค สมุดงานที่อาจารย์ประจำวรรณแจกให้จะมี

- จำนวนบททั้งหมด  $N$  บท
- แต่ละบทจะมีจำนวนข้อ คือ  $t_1, t_2, t_3, \dots, t_N$  ตามลำดับซึ่งไม่จำเป็นต้องเท่ากัน และแต่ละบทจะเริ่มต้นด้วยข้อ 1, 2, 3,  $\dots, t_i$  โดยที่  $i = 1 \dots N$
- แต่ละบทจะเริ่มต้นหน้าใหม่เสมอ
- ในแต่ละหน้าจะบรรจุได้  $k$  ข้อเท่ากันทุกหน้า ยกเว้นหน้าสุดท้ายของแต่ละบทอาจจะน้อยกว่า  $k$  ข้อได้
- หมายเลขหน้าจะเริ่มต้นที่ 1

จอมแจ่มเป็นเด็กขยัน ได้พยายามทำการบ้านด้วยตัวเองทุกข้อและด้วยความพากเพียร แต่เนื่องจากการบ้านมีมากดังนั้นจอมแจ่มจึงต้องการให้ความสำคัญกับบางข้อที่คาดว่าจะออกเป็นข้อสอบเป็นพิเศษ ด้วยความเชื่อส่วนตัวของจอมแจ่ม จอมแจ่มเชื่อว่าข้อพิเศษที่อาจารย์ประจำวรรณจะเลือกมาออกเป็นข้อสอบนั้น คือข้อที่มีหมายเลขข้อเป็นเลขเดียวกับหมายเลขหน้า

ตัวอย่างเช่น สมุดงานที่อาจารย์ประจำวรรณแจกให้ มี 5 บท มีจำนวนข้อของแต่ละบทคือ 4 2 6 1 และ 10 ตามลำดับ ถ้าแต่ละหน้าจะบรรจุได้ 3 ข้อ ข้อพิเศษที่จอมแจ่มคาดว่าจะออกเป็นข้อสอบจะมี 4 ข้อ ได้แก่ ข้อ 1 ของบทที่ 1, ข้อ 5 ของบทที่ 3, ข้อ 9 ของบทที่ 5 และ ข้อ 10 ของบทที่ 5 ดังแสดงในรูปที่ 1

บทที่ 1	บทที่ 2	บทที่ 3	บทที่ 4	บทที่ 5
1. <input type="text"/>	1. <input type="text"/>	1. <input type="text"/>	1. <input type="text"/>	1. <input type="text"/>
2. <input type="text"/>	2. <input type="text"/>	2. <input type="text"/>	2. <input type="text"/>	2. <input type="text"/>
3. <input type="text"/>	3. <input type="text"/>	3. <input type="text"/>	3. <input type="text"/>	3. <input type="text"/>
4. <input type="text"/>	4. <input type="text"/>	4. <input type="text"/>	4. <input type="text"/>	4. <input type="text"/>
5. <input type="text"/>	5. <input type="text"/>	5. <input type="text"/>	5. <input type="text"/>	5. <input type="text"/>
6. <input type="text"/>	6. <input type="text"/>	6. <input type="text"/>	6. <input type="text"/>	6. <input type="text"/>
7. <input type="text"/>	7. <input type="text"/>	7. <input type="text"/>	7. <input type="text"/>	7. <input type="text"/>
8. <input type="text"/>	8. <input type="text"/>	8. <input type="text"/>	8. <input type="text"/>	8. <input type="text"/>
9. <input type="text"/>	9. <input type="text"/>	9. <input type="text"/>	9. <input type="text"/>	9. <input type="text"/>
10. <input type="text"/>	10. <input type="text"/>	10. <input type="text"/>	10. <input type="text"/>	10. <input type="text"/>

รูปที่ 1 แสดงการวางจำนวนข้อในแต่ละหน้าของแต่ละบท และข้อพิเศษ

จงเขียนโปรแกรมเพื่อนับว่าข้อพิเศษที่จอมแจ่มคาดว่าจะออกเป็นข้อสอบมีทั้งหมดกี่ข้อ

### ข้อมูลเข้า

ข้อมูลเข้ามีสองบรรทัด

1. บรรทัดแรก ประกอบด้วยจำนวนเต็มบวกสองจำนวน ได้แก่ จำนวนบท  $N$  และ จำนวนโจทย์ที่สามารถมีได้ในแต่ละหน้า  $k$  คั่นแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง โดยกำหนดให้  $1 \leq N \leq 30$  และ  $1 \leq k \leq 30$
2. บรรทัดที่สอง ประกอบด้วยจำนวนเต็มบวก  $N$  จำนวน แทนจำนวนข้อในแต่ละบท แต่ละจำนวนระบุค่าได้ตั้งแต่ 0 ถึง 100 โดยเริ่มบทที่ 1 ถึง บทที่  $N$  จากซ้ายไปขวา คั่นแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง

### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และเซตของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

ข้อมูลส่งออกมี 1 บรรทัด แสดงจำนวนข้อพิเศษที่มีหมายเลขข้อเป็นเลขเดียวกับหมายเลขหน้า

ตัวอย่างที่ 1 (จากตัวอย่างในรูปแบบที่ 1)

ข้อมูลเข้า	ข้อมูลส่งออก
5 3 4 2 6 1 10	4

ตัวอย่างที่ 2 ถ้ามี 3 บท แต่ละบทมี 7, 0, 4 ข้อตามลำดับ และแต่ละหน้ามีได้ 4 ข้อ ก็จะมีข้อพิเศษได้ 2 ข้อคือ ข้อ 1 ของบทที่ 1 และข้อ 3 ของบทที่ 3

ข้อมูลเข้า	ข้อมูลส่งออก
3 4 7 0 5	2

ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

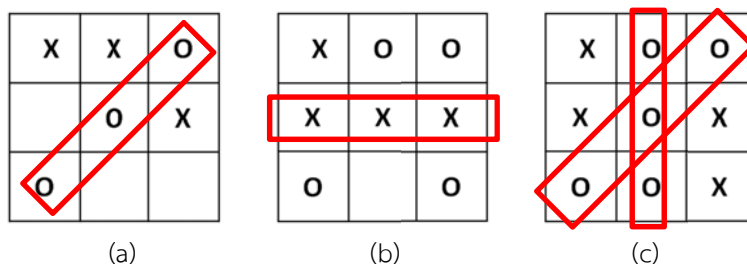
ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

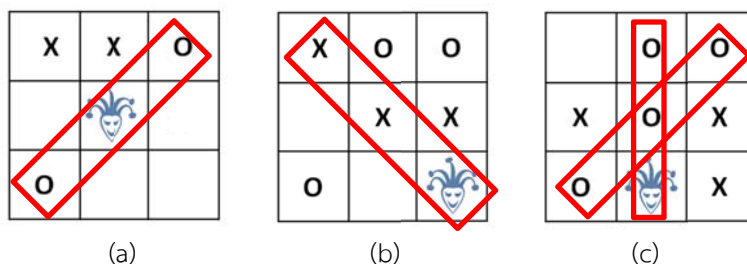
## เกมโอเอ็กซ์ (OX)

เกมโอเอ็กซ์เป็นเกมที่มีกติกาง่ายๆ เล่นกันเพียงระหว่างผู้เล่นสองคนบนตารางขนาด  $N \times N$  โดยผู้เล่นแต่ละคนจะเลือกตัวอักษร O หรือ X แทนสัญลักษณ์ของตัวเอง และผลัดกันใส่ตัวอักษรสัญลักษณ์ที่เลือกลงในตาราง ผู้เล่นที่มีสัญลักษณ์ของตนเองครบ  $N$  ตัวตลอดแนวคอลัมน์เดียวกัน หรือตลอดแนวแถวเดียวกัน หรือตลอดแนวทแยงด้านซ้ายหรือด้านขวาได้ก่อน ถือเป็นผู้ชนะ ยกตัวอย่างตารางเกมดังรูปข้างล่าง



รูปที่ 1 ตัวอย่างตารางเกมขนาด  $3 \times 3$  ที่ (a) ผู้เล่นที่เลือกตัวอักษร O ชนะในตาที่สาม (b) ผู้เล่นที่เลือกตัวอักษร x ชนะในตาที่สี่ และ (c) ผู้เล่นที่เลือกตัวอักษร O ชนะในตาที่ห้า

นอกจากนี้เพื่อเพิ่มความท้าทายในการเล่น อาจมี Joker (J) อยู่ในช่องหนึ่งของตาราง ผู้เล่นไม่สามารถใส่ตัวอักษรของตนเองลงในช่องที่มี Joker อยู่ได้ แต่สามารถนับ Joker เป็นตัวอักษรของตนเองได้ ดังนั้นหากช่องอื่นๆ ในแนวคอลัมน์เดียวกัน หรือแนวแถวเดียวกัน หรือแนวทแยงมีตัวอักษรของตนเองอยู่แล้ว ก็สามารถชนะได้ทันที ยกตัวอย่างเช่นในรูปที่ 2



รูปที่ 2 ตัวอย่างตารางเกมขนาด  $3 \times 3$  ที่ (a) ผู้เล่นที่เลือกตัวอักษร O ชนะในตาที่สอง (b) ผู้เล่นที่เลือกตัวอักษร x ชนะในตาที่สาม และ (c) ผู้เล่นที่เลือกตัวอักษร O ชนะในตาที่สี่

จงเขียนโปรแกรมเพื่อตัดสินว่าด้วยตารางในเกมโอเอ็กซ์ที่ให้มา ผู้เล่นคนใดเป็นผู้ชนะ และเป็นตารางเกมในตาที่เท่าไร

### ข้อมูลเข้า

ข้อมูลเข้ามี  $N+1$  บรรทัด

1. บรรทัดแรก ประกอบด้วยจำนวนเต็มบวกหนึ่งจำนวน ระบุขนาดตาราง ( $N$ ) โดยกำหนดให้  $3 \leq N \leq 15$
2. บรรทัดที่  $i = 2$  ถึงบรรทัดที่  $N+1$  แต่ละบรรทัด  $i$  ระบุตัวอักษร O, X, J หรือ - ที่แต่ละตำแหน่งของแถวที่  $i-1$  ของตาราง โดยตัวอักษร J แทน Joker และตัวอักษร - แทนช่องว่างในตาราง คั่นแต่ละตัวอักษรด้วยช่องว่างหนึ่งช่อง

### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และเซตของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

## ข้อมูลส่งออก

ข้อมูลส่งออกมีหนึ่งบรรทัด แสดงตัวอักษรหนึ่งตัวคั่นด้วยช่องว่างหนึ่งช่องและจำนวนเต็มหนึ่งจำนวน

โดยตัวอักษรแรก คือ O เมื่อผู้เล่นที่เลือกตัวอักษร O ชนะ หรือคือ X เมื่อผู้เล่นที่เลือกตัวอักษร X ชนะ หรือคือ - เมื่อไม่มีผู้ชนะ

ส่วนจำนวนเต็มหนึ่งจำนวน ระบุว่าเป็นตารางเกมในตาที่เท่าไร ทั้งนี้กำหนดให้ตาที่เริ่มเล่นตาแรกเป็นตาที่ 1

### ตัวอย่างที่ 1 (จากตัวอย่างในรูป 1(a))

ข้อมูลเข้า	ข้อมูลส่งออก
3 X X O - O X O - -	O 3

### ตัวอย่างที่ 2 (จากตัวอย่างในรูป 1(c))

ข้อมูลเข้า	ข้อมูลส่งออก
3 X O O X O X O O X	O 5

### ตัวอย่างที่ 3 (จากตัวอย่างในรูป 2(b))

ข้อมูลเข้า	ข้อมูลส่งออก
3 X O O - X X O - J	X 3

### ตัวอย่างที่ 4

ข้อมูลเข้า	ข้อมูลส่งออก
5 X O X X X O - - - - X O O X - X - - - - O O - - O	- 7



## ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

## ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรมดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็น ชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

## ใครป๊อบกว่ากัน (PopularVote)

สำนักข่าวบันเทิงรายหนึ่งคิดวิธีชว่นปดหัวในการวัดความนิยมของตัวละครสองตัวในซีรีส์เกาหลีชื่อดัง กับต้นยู และหมอคัง ด้วยการแสดงบิลบอร์ดคะแนนนิยมของทั้งสองร่วมกันเป็นเลขฐานสอง  $N$  หลัก หมอคังได้ตัวเลข  $k$  หลักจากทางซ้าย ในขณะที่กับต้นยูได้ตัวเลขทั้งหมด  $u$  หลักที่เหลือ ( $N = k + u$ ) ตัวอย่างเช่น ในรูปที่ 1 จะได้ว่าหมอคังและกับต้นยูได้รับความนิยมจากผู้ชมเท่าๆ กัน นั่นคือ 011 และ 0011 ซึ่งมีค่าเท่ากับ 3 ทั้งคู่ ในขณะที่ตัวอย่างในรูปที่ 2 นั้นหมอคังได้รับคะแนนความนิยม 0110 (=6) ซึ่งมากกว่าคะแนนความนิยมของกับต้นยู 011 (=3)

0	1	1	0	0	1	1
---	---	---	---	---	---	---

รูปที่ 1 ตัวอย่างบิลบอร์ดแสดงเลขฐานสองทั้งหมด  $N = 7$  หลัก  $k = 3$  และ  $u = 4$

0	1	1	0	0	1	1
---	---	---	---	---	---	---

รูปที่ 2 ตัวอย่างบิลบอร์ดแสดงเลขฐานสองทั้งหมด  $N = 7$  หลัก  $k = 4$  และ  $u = 3$

จงเขียนโปรแกรมเพื่อหาว่าจากบิลบอร์ดคะแนนที่ให้มา ใครที่ได้รับคะแนนความนิยมหรือมีความป๊อบปูลาร์มากกว่ากัน

### ข้อมูลเข้า

ข้อมูลเข้ามีสองบรรทัด

- บรรทัดแรก ประกอบด้วยจำนวนเต็มบวกสองจำนวน ได้แก่  $N$  และ  $k$  คั่นแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง โดยกำหนดให้  $2 \leq N \leq 30$  และ  $1 \leq k \leq 25$
- บรรทัดที่สอง ประกอบด้วยจำนวนเต็มบวก  $N$  จำนวน แต่ละจำนวนระบุค่า 0 หรือ 1 แทนเลขฐานสองในแต่ละหลัก จากซ้ายไปขวา คั่นแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง

### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และเซตของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

ข้อมูลส่งออกมีสามบรรทัด

- บรรทัดแรก แสดงตัวเลขฐานสองจำนวน  $k$  หลัก (ค่าแต่ละหลักเป็น 0 หรือ 1) ระบุคะแนนความนิยมของหมอคัง คั่นแต่ละหลักด้วยช่องว่างหนึ่งช่อง
- บรรทัดที่สอง แสดงตัวเลขฐานสองจำนวน  $u$  หลัก โดย  $u = N - k$  (ค่าแต่ละหลักเป็น 0 หรือ 1) ระบุคะแนนความนิยมของกับต้นยู คั่นแต่ละหลักด้วยช่องว่างหนึ่งช่อง
- บรรทัดที่สาม แสดงจำนวนเต็มหนึ่งตัว ค่าเป็น 0 หากทั้งคู่ได้รับความนิยมเท่ากัน หรือค่าเป็น 1 หากหมอคังได้รับความนิยมมากกว่า หรือค่าเป็น 2 หากกับต้นยูได้รับความนิยมมากกว่า

ตัวอย่างที่ 1 (จากตัวอย่างในรูปแบบที่ 1)

ข้อมูลเข้า	ข้อมูลส่งออก
7 3	0 1 1
0 1 1 0 0 1 1	0 0 1 1
	0

ตัวอย่างที่ 2 (จากตัวอย่างในรูปแบบที่ 2)

ข้อมูลเข้า	ข้อมูลส่งออก
7 4	0 1 1 0
0 1 1 0 0 1 1	0 1 1
	1

ตัวอย่างที่ 3

ข้อมูลเข้า	ข้อมูลส่งออก
10 5	0 1 1 1 0
0 1 1 1 0 1 1 0 0 0	1 1 0 0 0
	2

ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรกดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย

*/	ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java
----	---

## เป่า-ยิง-จับ (roshambo)

คช. หนึ่งและคณ. สองเล่นเกมเป่า-ยิง-จับกัน โดยมีกติกาอยู่ว่าจะเล่นกันทั้งหมด  $N$  รอบ ( $1 \leq N \leq 100$ ) ในการเล่นแต่ละรอบผู้เล่นแต่ละคนสามารถทำมือออกมาเป็นรูปค้อน กระดาศ หรือกรรไกรก็ได้ แล้วคิดคะแนนในแต่ละรอบจากรูปมือของผู้เล่นทั้งสองดังนี้

- ค้อน และ กรรไกร: คนที่ออกค้อนได้ 1 แต้ม คนที่ออกกรรไกรไม่ได้แต้มในรอบนั้น
- กระดาศ และ ค้อน: คนที่ออกกระดาศได้ 1 แต้ม คนที่ออกค้อนไม่ได้แต้มในรอบนั้น
- กรรไกร และ กระดาศ: คนที่ออกกรรไกรได้ 1 แต้ม คนที่ออกกระดาศไม่ได้แต้มในรอบนั้น
- รูปเดียวกัน (ค้อน และ ค้อน, กระดาศ และ กระดาศ, หรือ กรรไกร และ กรรไกร): ทั้งคู่ไม่ได้แต้มในรอบนั้น

ผู้ชนะคือผู้ที่ได้คะแนนรวมสูงสุดหลังจากเล่นครบทั้ง  $N$  รอบ และผู้ชนะจะได้รับเงินรางวัลเท่ากับห้าเท่าของผลต่างของคะแนนรวม กล่าวคือ เงินรางวัล =  $5 * (\text{คะแนนรวมผู้ชนะ} - \text{คะแนนรวมผู้แพ้})$  หากผู้เล่นทั้งสองได้คะแนนเท่ากันถือว่าเสมอ ไม่มีใครได้เงินรางวัล ตัวอย่างดังรูปที่ 1

รอบที่	รูปมือของคช.หนึ่ง	รูปมือของคณ.สอง	คะแนนรวมของคช.หนึ่ง	คะแนนรวมของคณ.สอง
1			1	0
2			2	0
3			2	0
ผู้ชนะคือ คช.หนึ่ง			เงินรางวัลที่ได้ = $5 * (2 - 0) = 10$ บาท	

รูปที่ 1 ตัวอย่างการเล่นเกมเป่า-ยิง-จับ 3 รอบ ( $N=3$ ) และการคิดผู้ชนะและเงินรางวัลที่ได้

จงเขียนโปรแกรมเพื่อช่วยคำนวณคะแนนรวมของผู้เล่นจากการแข่งขันทุกรอบ เพื่อตัดสินว่าใครเป็นผู้ชนะในการเล่นเกมนี้อ และได้รับเงินรางวัลเท่าไร

## ข้อมูลเข้า

มี  $N+1$  บรรทัด แต่ละบรรทัดมีข้อมูลดังนี้

บรรทัดที่ 1 มีจำนวนเต็มหนึ่งจำนวน ระบุจำนวนรอบ ( $N$ )

บรรทัดที่ 2 ถึง  $N+1$  แต่ละบรรทัดมีจำนวนเต็มสองจำนวน คั่นระหว่างจำนวนด้วยช่องว่างหนึ่งช่อง

จำนวนแรกแทนรูปมือของ A และจำนวนที่สองแทนรูปมือของ B โดยเลขจำนวนเต็มแต่ละจำนวนมีค่า 0, 1 หรือ 2 แทนรูปค้อน กระดาศ และกรรไกรตามลำดับ

### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และเซตของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

มีสองบรรทัด ดังนี้

บรรทัดแรก เป็นเลขจำนวนเต็มบวกหนึ่งจำนวน แสดงค่า 1 หากคช. หนึ่งเป็นผู้ชนะ, ค่า 2 หากคญ. สองเป็นผู้ชนะ และค่า 0 หากทั้งคู่เสมอกัน

บรรทัดที่สอง เป็นเลขจำนวนเต็มบวกหนึ่งจำนวน แสดงจำนวนเงินรางวัลที่ผู้ชนะได้รับ

### ตัวอย่างที่ 1 (จากรูปที่ 1)

ข้อมูลเข้า	ข้อมูลส่งออก
3	1
1 0	10
2 1	
2 2	

### ตัวอย่างที่ 2

ข้อมูลเข้า	ข้อมูลส่งออก
5	2
0 0	5
1 2	
2 0	
1 1	
2 1	

### ตัวอย่างที่ 3

ข้อมูลเข้า	ข้อมูลส่งออก
1	0
1 1	0

### ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรมดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็น ชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

## ปราบปีศาจหิมะ (SnowMonster)

ปีศาจหิมะที่ดุร้ายออกอาละวาดในเมืองแสนสุข ผู้ครองเมืองประกาศหาเหล่านักรบผู้กล้ามาช่วยกันปราบปีศาจหิมะตัวนี้ และได้นักรบผู้กล้ามารวมกันทั้งหมด  $N$  คน มีหมายเลขประจำตัว ( $i$ ) ตั้งแต่ 1 ถึง  $N$  และต่างมีท่าไม้ตายที่จะสามารถลดพลังชีวิตของปีศาจหิมะลงได้บางส่วนหรือทั้งหมด ทั้งนี้ท่าไม้ตายของนักรบผู้กล้าคนที่  $i$  สามารถลดพลังชีวิตปีศาจหิมะได้  $D_i$  หน่วย ( $1 \leq i \leq N$ ) ปีศาจหิมะนี้มีพลังชีวิตทั้งหมด  $E$  หน่วย ปีศาจหิมะจะถูกปราบลงได้ก็ต่อเมื่อมีนักรบผู้กล้าสองคน  $i$  และ  $j$  ผนึกกำลังกันแล้วสามารถกำจัดพลังชีวิตของปีศาจหิมะได้หมดพอดี ไม่ขาดไม่เกินแม้หน่วยเดียว นอกจากนี้คนหนึ่งจะต้องมีหมายเลขประจำตัวเป็นเลขคู่ และอีกคนจะต้องมีหมายเลขประจำตัวเป็นเลขคี่ด้วย

ยกตัวอย่างเช่น หากปีศาจหิมะมีพลังชีวิต 48 หน่วย แล้วมีกลุ่มนักรบผู้กล้า 5 คน ซึ่งมีท่าไม้ตายที่ลดพลังชีวิตปีศาจหิมะได้ 10, 30, 28, 20 และ 18 หน่วยตามลำดับ จะได้ว่ามีนักรบผู้กล้า 2 คู่ที่สามารถส่งไปปราบปีศาจหิมะได้ นั่นคือ คู่แรกได้แก่นักรบผู้กล้าคนที่สองและคนที่ห้า คู่ที่สองได้แก่นักรบผู้กล้าคนที่สามและสี่

จงเขียนโปรแกรมเพื่อค้นหาว่าในบรรดานักรบผู้กล้าที่มารวมกันนี้มีนักรบผู้กล้าทั้งหมดกี่คู่ที่สามารถปราบปีศาจหิมะตัวนี้ได้

### ข้อมูลเข้า

ข้อมูลเข้ามี  $N+1$  บรรทัด

- บรรทัดแรก ประกอบด้วยจำนวนเต็มบวก 2 จำนวน ได้แก่  $N$  และ  $E$  ตามลำดับ คั่นแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง โดยกำหนดให้  $2 \leq N \leq 100$  และ  $2 \leq E \leq 1,000$
- บรรทัดที่สองถึงบรรทัดที่  $N+1$  แต่ละบรรทัดมีจำนวนเต็มบวกหนึ่งจำนวน ระบุ  $D_i$  ซึ่งเป็นปริมาณพลังชีวิตของปีศาจหิมะที่ท่าไม้ตายของนักรบผู้กล้าคนที่  $i$  สามารถลดได้ คั่นแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง กำหนดให้  $1 \leq i \leq N$  และ  $1 \leq D_i \leq 1,000$

### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และขีดของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

ข้อมูลส่งออกมีสองบรรทัด

- บรรทัดแรก แสดงจำนวนเต็มบวกหนึ่งจำนวน ระบุจำนวนคู่ของนักรบผู้กล้าทั้งหมดที่สามารถผนึกกำลังกันปราบปีศาจหิมะได้
- บรรทัดที่สอง แสดงจำนวนเต็มบวกหนึ่งจำนวน ระบุหมายเลขประจำตัวที่มีค่ามากที่สุดของนักรบผู้กล้าที่สามารถเป็นหนึ่งในคู่ปราบปีศาจหิมะจากข้อ 1.



ตัวอย่างที่ 1 (จากตัวอย่างในโจทย์)

ข้อมูลเข้า	ข้อมูลส่งออก
5 48	2
10	5
30	
28	
20	
18	

ตัวอย่างที่ 2

ข้อมูลเข้า	ข้อมูลส่งออก
9 112	3
10	
1	
2	
10	
102	
56	
3	
56	
56	

หมายเหตุ ทั้งสามคู่ที่ปราบบีศาจหิมะได้ คือ (1) นักรบคนที่ 4 และ 5, (2) นักรบคนที่ 6 และ 9, (3) นักรบคนที่ 8 และ 9

ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรกดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C	/* LANG: C++

COMPILER: WDC */	COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็น ชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

## แท็กซี่ (Taxi)

บรรจงบเปิดธุรกิจบริการรถแท็กซี่ โดยลูกค้าจะติดต่อมาที่บริษัทเพื่อนัดหมายเวลาที่จะให้ไปรับ บรรจก็ได้เขียนโปรแกรมเพื่อบันทึกข้อมูลงานในแต่ละวัน ได้แก่ เวลาที่ลูกค้านัดหมายรถแท็กซี่ และจัดรถแท็กซี่สำหรับให้บริการแก่ลูกค้า ผลลัพธ์ที่ได้จากโปรแกรมของบรรจมีการจัดเก็บในรูปแบบตารางที่มีสองคอลัมน์ โดยที่คอลัมน์แรกระบุเวลาที่ลูกค้านัดหมายรถแท็กซี่ ส่วนคอลัมน์ที่สองเป็นรหัสรถแท็กซี่ที่โปรแกรมจัดไว้สำหรับให้บริการแก่ลูกค้า บรรจพบปัญหาของโปรแกรมของเขาซึ่งสืบเนื่องจากตอนที่ผู้ใช้บริการติดต่อเข้ามาได้ระบุเวลานัดหมายรถแบบไม่เรียงตามลำดับ บรรจต้องการเรียงลำดับข้อมูลในตารางผลลัพธ์ที่ได้จากโปรแกรมของเขาใหม่โดยให้ข้อมูลเรียงลำดับตามเวลานัดหมายที่น้อยที่สุด (เข้าที่สุด) ไปเวลานัดหมายที่มากที่สุด (ออกที่สุด) เพื่อความสะดวกในการทำงาน โดยภายหลังจากเรียงลำดับข้อมูลแล้ว รถแท็กซี่แต่ละคันต้องยังคงให้บริการลูกค้าในเวลาเดิมตามที่กำหนดไว้จากโปรแกรมของบรรจ

ตารางที่1: ข้อมูลก่อนการจัดเรียง

เวลาที่ต้องการรถแท็กซี่	รถที่จะให้บริการ
21:15	0004
06:00	0002
05:45	0007
21:15	1025

ตารางที่2: ข้อมูลหลังการจัดเรียงตามลำดับเวลา

เวลาที่ต้องการรถแท็กซี่	รถที่จะให้บริการ
05:45	0007
06:00	0002
21:15	0004
21:15	1025

จงเขียนโปรแกรมเพื่อจัดเรียงข้อมูลในตารางตามเวลานัดหมายลูกค้าจากเวลาที่น้อยที่สุดไปเวลานัดหมายที่มากที่สุด

### ข้อมูลเข้า

ข้อมูลเข้ามี  $N+1$  บรรทัด

- บรรทัดแรก ประกอบด้วยจำนวนเต็มบวก 1 จำนวน ได้แก่  $N$  แทนจำนวนผลลัพธ์ทั้งหมดจากโปรแกรมของบรรจ โดยกำหนดให้  $1 \leq N \leq 100$
- บรรทัดที่สองถึงบรรทัดที่  $N+1$  แต่ละบรรทัดมีจำนวนเต็มบวก 2 จำนวน  $t$  และ  $u$  คั่นแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง โดยที่  $u$  คือ รหัสรถแท็กซี่ที่ต้องการ และ  $t$  คือ เวลาที่ระบุในรูปแบบ  $hhmm$  สองตัวแรกระบุชั่วโมง  $00 \leq hh \leq 23$  และ สองตัวหลังระบุนาที  $00 \leq mm \leq 59$  (รหัสรถแท็กซี่  $u$  ในตารางนี้ไม่มีค่าซ้ำ)

### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และเซตของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

ข้อมูลส่งออกมี  $N$  บรรทัด แต่ละบรรทัดที่  $i$  แสดงจำนวนเต็มสองจำนวน ดังนี้

- จำนวนแรก คือ ตัวเลขจำนวนเต็ม 4 หลัก ที่ระบุเวลาในรูปแบบ  $hhmm$
- จำนวนที่สอง คือ รหัสรถแท็กซี่

ตัวอย่างที่ 1 (จากตัวอย่างในโจทย์)

ข้อมูลเข้า	ข้อมูลส่งออก
4	0545 0007
2115 0004	0600 0002
0600 0002	2115 0004
0545 0007	2115 1025
2115 1025	

ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรมดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มีการสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

## กล้วยไม้ป่า (WildOrchid)

เขตรักษาพันธุ์สัตว์ป่าภูหลวง จ.เลย ได้ชื่อว่าเป็นมรดกแห่งอีสาน ที่นี้ได้รับการยกย่องว่าเป็นดินแดนแห่งพืชพันธุ์ไม้ที่สวยงามในอันดับต้นๆของเมืองไทย โดยเฉพาะกับกล้วยไม้ป่าหายากที่ภูหลวงมีให้ชมเป็นจำนวนมาก อาทิ กุหลาบพันปีแดง กุหลาบพันปีขาว เอื้องตาเหิน เอื้องม่อนไข่ สิงโตใบพาย เอื้องสำเภางาม สร้อยระย้า ก้อยมอญ ฯลฯ แต่ละเดือนจะมีกล้วยไม้บานต่างชนิดกันตลอดทั้งปี เจ้าหน้าที่เขตรักษาพันธุ์สัตว์ป่าต้องการเก็บข้อมูลว่าพบกล้วยไม้อะไรบ้างในเดือนไหนบ้างเพื่อเป็นข้อมูลประชาสัมพันธ์ให้กับนักท่องเที่ยว จึงทำการเดินสำรวจทุกเดือนและบันทึกลงเครื่องส่งสัญญาณที่เชื่อมต่อไปยังคอมพิวเตอร์เพื่อประมวลผล ดังนั้นเมื่อพบกล้วยไม้ที่เด่นๆบาน (มีไม่เกิน 26 ชนิด) ก็จะบันทึกเป็นรหัส 2 หลัก หลักแรกเป็นรหัสเดือนตั้งแต่มกราคมถึงธันวาคมแทนด้วยเลขฐาน 16 (1 ถึง C) หลักที่สองเป็นรหัสกล้วยไม้แทนด้วยตัวอักษรภาษาอังกฤษเป็นตัวพิมพ์ใหญ่ (A ถึง Z) และในการสำรวจเจ้าหน้าที่จะสำรวจอาทิตย์ละครั้งและบันทึกข้อมูลต่อจากครั้งก่อนเสมอ ดังนั้นข้อมูลที่บันทึกจึงเรียงลำดับตามเดือน เช่น

1A1F1A1B1A1D1D1D1A2B2C2E3F3F3E3E

หมายถึง เดือนมกราคม มีกล้วยไม้ 4 ชนิด ได้แก่ A B D และ F บาน

เดือนกุมภาพันธ์ มีกล้วยไม้ 3 ชนิด ได้แก่ B C และ E บาน

เดือนมีนาคม มีกล้วยไม้ 2 ชนิด ได้แก่ E และ F บาน

เดือนมกราคมเป็นเดือนที่พบกล้วยไม้บานมากชนิดที่สุด คือ 4 ชนิด

ให้นักศึกษาเขียนโปรแกรมเพื่อนับจำนวนชนิดของกล้วยไม้ที่พบว่าบานในแต่ละเดือน และบอกด้วยว่าในเดือนที่พบกล้วยไม้บานมากชนิดที่สุด มีกล้วยไม้บานทั้งหมดกี่ชนิด

### ข้อมูลเข้า

ข้อมูลเข้ามี 1 บรรทัด เป็นชุดข้อมูลที่เจ้าหน้าที่บันทึกตลอดทั้งปีเกี่ยวกับการพบกล้วยไม้ที่บานในเขตรักษาพันธุ์สัตว์ป่าภูหลวง

#### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และ เซ็ตของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

ข้อมูลส่งออกมี 13 บรรทัด ดังนี้

บรรทัดที่ 1-12 จะเป็นเลขจำนวนเต็มหนึ่งจำนวน เป็นจำนวนของชนิดกล้วยไม้ที่บานในเดือนนั้น ตามลำดับของเดือน

บรรทัดที่ 13 ระบุถึงจำนวนชนิดของกล้วยไม้ที่บาน ในเดือนที่มีกล้วยไม้บานมากชนิดที่สุด

ตัวอย่างที่ 1

ข้อมูลเข้า	ข้อมูลส่งออก
1A1F1A1B1A1D1D1D1A2B2C2E3F3F3E3E	4
	3
	2
	0
	0
	0
	0
	0
	0
	0
	0
	0
	0
	4

ตัวอย่างที่ 2

ข้อมูลเข้า	ข้อมูลส่งออก
1O1R2F2F3R3H3J4Y5I6A9L9O9E9E9WCC	2
	1
	3
	1
	1
	1
	0
	0
	4
	0
	0
	1
	4



## ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

## ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรกดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.8.0_144	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

	<p>ICPC Pre-Contest 2017</p> <p>First Round</p>	
---	---	---

A	Songkran	
	Time Limit	1 second
	Memory Limit	128 MB

ช่วงเทศกาลสงกรานต์ที่ผ่านมา คุณเปิดร้านเติมน้ำใส่ถังสำหรับผู้ที่ต้องการเติมน้ำเพื่อนำไปเล่นสงกรานต์ มีผู้คนนำปิ่นฉีบน้ำและถังน้ำมาต่อแถวรอซื้อน้ำจากคุณเป็นจำนวนมาก ที่ร้านของคุณมีถังน้ำอยู่หลายขนาด แต่ละถังก็มีน้ำอยู่ไม่เท่ากัน เพื่อให้ทำงานได้สะดวกขึ้น คุณจึงคิดจะนำน้ำที่อยู่ในถังหนึ่ง ย้ายไปรวมกับอีกถังหนึ่ง เพื่อลดจำนวนถังที่ต้องใช้ ข้อกำหนดของการย้ายน้ำจากถังหนึ่งไปอีกถังหนึ่งคือ จะย้ายน้ำเป็นปริมาณมากที่สุดที่ได้ ที่ไม่ทำให้น้ำในถังที่ย้ายไปล้นออกมา ตัวอย่างเช่น หากต้องการย้ายน้ำในถัง A ขนาด 5 ลิตร ซึ่งมีน้ำ 3 ลิตร ไปรวมกับถัง B ขนาด 7 ลิตร ซึ่งมีน้ำ 5 ลิตร เมื่อย้ายน้ำจากถัง A ไปยังถัง B แล้ว จะทำให้ถัง A เหลือน้ำ 1 ลิตร ในขณะที่ถัง B มีน้ำ 7 ลิตร

ให้เขียนโปรแกรมเพื่อคำนวณปริมาณน้ำในแต่ละถังหลังทำการย้ายน้ำจากถัง A ไปยัง B แล้ว

### ข้อมูลเข้า

บรรทัดแรกรับจำนวนเต็ม  $A$   $X$  แทนขนาดของถัง A และปริมาณน้ำในถัง A ( $0 \leq X \leq A \leq 1000$ )

บรรทัดที่สองรับจำนวนเต็ม  $B$   $Y$  แทนขนาดของถัง B และปริมาณน้ำในถัง B ( $0 \leq Y \leq B \leq 1000$ )

### ข้อมูลออก

แสดงคำตอบบรรทัดเดียว เป็นจำนวนเต็ม 2 จำนวน แทนปริมาณน้ำในถัง A และถัง B หลังทำการย้ายน้ำจากถังหนึ่งไปอีกถังหนึ่งแล้ว คั่นด้วยเว้นวรรค

### ตัวอย่าง

ข้อมูลเข้า	ข้อมูลออก
5 3 7 5	1 7
10 2 12 4	0 6
9 1 8 8	1 8



## หุ่นน้อยคอยเก็บรัก (BB8)



บีบีเอท (BB8) เป็นหุ่นยนต์ตัวน้อยมีหน้าที่เป็นผู้ช่วยของเจไดคอยเก็บความรัก (♥) ที่ซ่อนอยู่ในสวนของสหพันธ์ ความรักที่เก็บได้จะเป็นพลังเสริมให้กับเจไดในการต่อสู้เพื่อนำพาความสงบสุขกลับมาให้กับกาแล็คซี่อีกครั้งหนึ่ง

กำหนดให้สวนของสหพันธ์เป็นรูปสี่เหลี่ยมผืนผ้า แบ่งเป็นแปลงปลูกความรักขนาดเท่าๆ กัน จำนวน  $M \times N$  แปลง สวนของสหพันธ์มีทางเข้าเพียง 1 ทาง คือ ที่แปลงมุมบนซ้ายสุด (top left cell) และมีทางออกแค่ 1 ทาง คือ ที่แปลงมุมล่างขวาสุดของสวน (bottom right cell) พื้นที่บางแปลงของสวนอาจปลูกความรักไม่ขึ้นเลย และสำหรับแปลงที่ปลูกรักได้ ก็จะมีผลผลิตจำนวนเพียง 1 รักเท่านั้น เนื่องจากความรักปลูกได้ยากมากอีกทั้งคุณภาพดินของพื้นที่สวนแต่ละแปลงไม่เท่ากัน

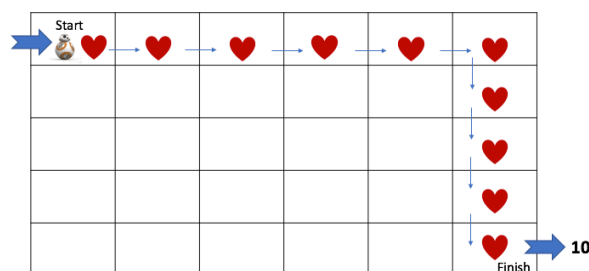
ในการเก็บความรัก หุ่นบีบีเอทจะทำงานตามข้อกำหนด ต่อไปนี้

1. จะเริ่มต้นทำงานที่แปลงมุมบนซ้ายสุดของสวนเสมอ (top left cell)
2. จากแปลงปลูกรักแปลงใด ๆ ในสวน บีบีเอทสามารถเคลื่อนที่ได้แค่สองทิศทาง คือ เคลื่อนไปที่แปลงด้านขวา 1 แปลง (move one cell to the right) หรือ เคลื่อนที่ลงด้านล่าง 1 แปลง (move one cell down)
3. จะมุ่งหน้าไปที่แปลงล่างขวาสุด (bottom right cell) ซึ่งเป็นทางออกจากสวน
4. เมื่อเข้าไปในแปลงใดที่มีความรักขึ้นในแปลงนั้น บีบีเอทจะเก็บความรักที่ซ่อนอยู่ในแปลงนั้นเสมอ

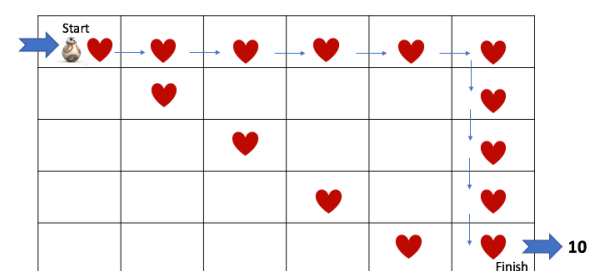
หุ่นบีบีเอทมีเป้าหมายที่สำคัญ คือ ต้องการเก็บความรักให้ได้มากที่สุดในแต่ละรอบที่เข้าไปในสวน

ให้นักศึกษาเขียนโปรแกรมเพื่อรับข้อมูล จำนวนแปลงปลูกความรัก  $M \times N$  แปลง และข้อมูลว่าแปลงในสวนของสหพันธ์แปลงใดบ้างซึ่งมีความรักปลูกขึ้น เพื่อคำนวณว่าบีบีเอทจะสามารถเก็บความรักได้มากที่สุดเท่าใดในการเคลื่อนที่จากแปลงมุมบนซ้ายสุด (ทางเข้า) มาที่แปลงมุมล่างขวาสุด (ทางออก)

**ตัวอย่างที่ 1** สวนสหพันธ์ที่มีจำนวนแปลงปลูกความรัก  $5 \times 6$  แปลง มีแปลงซึ่งปลูกความรักขึ้น ดังภาพ บีบีเอทจะสามารถเก็บความรักได้มากที่สุดเป็นจำนวน 10 รัก

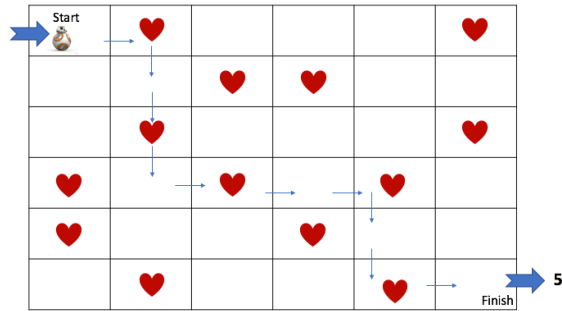


**ตัวอย่างที่ 2** สวนสหพันธ์ที่มีจำนวนแปลงปลูกความรัก  $5 \times 6$  แปลง มีแปลงซึ่งปลูกความรักขึ้น ดังภาพ บีบีเอทจะสามารถเก็บความรักได้มากที่สุดเป็นจำนวน 10 รัก



**ตัวอย่างที่ 3** สวนสหพันธ์ที่มีจำนวนแปลงปลูกความรัก  $6 \times 6$  แปลง มีแปลงซึ่งปลูกความรักขึ้น ดังภาพ บีบีเอทจะสามารถเก็บความรักได้มากที่สุดเป็นจำนวน 5 รัก

(นศ.โปรดทราบ รูปแบบการเคลื่อนที่ของบีบีเอทเพื่อเก็บความรักให้ได้จำนวนมากที่สุด คือ 5 รัก อาจมีมากกว่า 1 แบบ)



### ข้อมูลเข้า

ข้อมูลเข้ามี  $M+2$  บรรทัด ดังนี้

บรรทัดที่หนึ่ง ประกอบด้วยเลขจำนวนเต็มหนึ่งตัว ระบุค่า  $M$  กำหนดให้  $1 \leq M \leq 15$

บรรทัดที่สอง ประกอบด้วยเลขจำนวนเต็มหนึ่งตัว ระบุค่า  $N$  กำหนดให้  $1 \leq N \leq 15$

บรรทัดที่ 3 ถึง  $M+2$  แต่ละบรรทัดประกอบด้วยเลข 0 หรือ 1 จำนวนรวมทั้งสิ้น  $N$  ตัว คำนวณแต่ละจำนวนด้วยช่องว่างหนึ่งช่อง ระบุผลผลิตความรักในแต่ละแปลงของสวน

0 หมายถึง แปลงที่ปลูกความรักไม่ขึ้น

1 หมายถึง แปลงที่ปลูกความรักขึ้น

### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และ เซตของค่าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

ข้อมูลส่งออกมีหนึ่งบรรทัด แสดงผลลัพธ์เป็นเลขจำนวนเต็มหนึ่งตัว ระบุจำนวนความรักมากที่สุดที่บีบีเอทจะเก็บได้ในการเคลื่อนที่เข้าไปในสวน 1 รอบ

### ตัวอย่างที่ 1

ข้อมูลเข้า	ข้อมูลส่งออก
5	10
6	
1 1 1 1 1 1	
0 0 0 0 0 1	
0 0 0 0 0 1	
0 0 0 0 0 1	
0 0 0 0 0 1	
0 0 0 0 0 1	

### ตัวอย่างที่ 2

ข้อมูลเข้า	ข้อมูลส่งออก
5	10
6	
1 1 1 1 1 1	
0 1 0 0 0 1	
0 0 1 0 0 1	
0 0 0 1 0 1	
0 0 0 0 1 1	
0 0 0 0 1 1	

### ตัวอย่างที่ 3

ข้อมูลเข้า	ข้อมูลส่งออก
6	5
6	
0 1 0 0 0 1	
0 0 1 1 0 0	
0 1 0 0 0 1	
1 0 1 0 1 0	
1 0 0 1 0 0	
0 1 0 0 1 0	

### ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

### ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรกดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.8.0_144	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

## ติดดีฐานสอง (Binary)

บริษัทไฮสตรคอมพิวเตอร์ ต้องการสร้างเพื่อนเล่นฝึกทักษะคณิตศาสตร์ ชื่อ “ติดดี” ให้นักเรียนมัธยมสนุกกับการฝึกแปลงเลขฐานสองเป็นเลขฐานสิบ ฝึกคำนวณ และแปลงเลขฐานสิบเป็นฐานสอง จึงออกแบบตัวติดดีเป็นตัวละครตลกน่ารักน่ารัก ให้ผู้เล่นพูดออกคำสั่งกับติดดี โดยออกคำสั่งแบ่งเป็น 3 ส่วน คือ ผู้เล่นบอกจำนวนที่ต้องการเลขฐานสองกี่บิต (กี่หลัก) จากนั้นอ่านออกเสียงค่าของตัวเลขฐานสองนั้นทีละหลักตามลำดับการอ่านปกติจากซ้ายไปขวา และส่วนสุดท้ายให้พูดเลขฐานสิบตัวใดตัวหนึ่งในห้าตัวนี้ (2, 4, 8, 16, 32) เมื่อผู้ใช้บอกคำสั่งเสร็จ ติดดีก็จะทำตามทางคิดเลขแล้วก็ทำหน้าตาฉลาด ก่อนจะพูดคำตอบออกมา โดยคำตอบเป็นผลคูณของเลขฐานสองกับเลขฐานสิบที่ผู้ใช้พูด ติดดีจะพูดผลลัพธ์เป็นฐานสองทีละหลักตามลำดับการอ่านปกติจากซ้ายไปขวา ตัวอย่างเช่น ผู้ใช้พูด 3 101 2 หมายถึงเลขฐานสอง 3 บิตมีค่า  $101_2$  ไปคูณกับเลขฐานสิบคือเลข 2 และติดดีจะตอบว่า 1010 บริษัทไฮสตรคอมพิวเตอร์ได้สร้างส่วนรับเสียงพูดของผู้ใช้มาแปลงเป็นข้อมูลตัวเลขได้แล้ว แต่ต้องโปรแกรมที่ทำหน้าที่เป็นสมองของติดดี ที่จะทำการคำนวณค่าและส่งผลลัพธ์ที่ส่วนแสดงผลเสียงพูด จึงขอให้นักศึกษาเขียนโปรแกรมส่วนนั้นให้

### ข้อมูลเข้า

ให้นักศึกษารับข้อมูลเข้าจากคีย์บอร์ด โดยข้อมูลเข้ามีจำนวน  $N+2$  บรรทัด

- บรรทัดแรกรับจำนวนเต็มบวก  $N$  ( $1 \leq N \leq 8$ ) โดยที่  $N$  เป็นจำนวนหลัก (Bit) ของเลขฐานสองที่ผู้ใช้พูด
- บรรทัดที่สอง ถึง  $N+1$  ระบุเลขฐานสองบรรทัดละหนึ่งหลัก ซึ่งเป็นเลขฐานสองที่ผู้ใช้พูดตามลำดับการอ่านปกติจากซ้ายไปขวา
- บรรทัดที่  $N+2$  ระบุจำนวนเต็มฐานสิบ ( $m$ ) ที่จะนำไปคูณกับค่าจำนวนฐานสองที่ผู้ใช้พูด ( $m$  มีค่า 2, 4, 8, 16, 32 ค่าใดค่าหนึ่ง)

**หมายเหตุ** กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามขอบเขตเสมอ

### ข้อมูลส่งออก

ให้นักศึกษาแสดงผลลัพธ์ส่งออกที่จอภาพ ข้อมูลส่งออก มี  $X$  บรรทัด โดย  $X$  เป็นจำนวนหลัก (จำนวนบิตของผลลัพธ์) แต่ละบรรทัดแสดงค่าเลขฐานสองหนึ่งหลักที่ติดดีจะพูด โดยแสดงผลตามลำดับการอ่านปกติจากซ้ายไปขวา

#### ตัวอย่างที่ 1

ข้อมูลเข้า	ข้อมูลส่งออก
3	1
1	1
1	1
1	0
8	0
	0

#### ตัวอย่างที่ 2

ข้อมูลเข้า	ข้อมูลส่งออก
1	1
1	0
32	0
	0
	0
	0

#### ตัวอย่างที่ 3

ข้อมูลเข้า	ข้อมูลส่งออก
4	1
1	0
0	1
1	0
0	0
2	

### ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
จำนวนชุดทดสอบ	10
คะแนนของแต่ละชุดทดสอบ	10
คะแนนเต็มทั้งสิ้น	100
เงื่อนไขในการรันโปรแกรม	โปรแกรมจะต้องประมวลผลชุดข้อมูลตัวอย่างที่ให้มาได้ถูกต้อง

### ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรกดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

### รถประจำทาง (Bus)

นักศึกษาโดยสารรถประจำทางสาย 1023 เป็นประจำทุกวันในเวลาเร่งด่วน ตั้งแต่ 6:00 ถึง 8:00 น. ที่ป้ายรถเมล์อนุสาวรีย์ชัยสมรภูมิ นักศึกษาคิดจะทำป้ายรถเมล์อัจฉริยะซึ่งสามารถบอกเวลาเฉลี่ยในการรอรถเมล์สายดังกล่าวในช่วงเวลาเร่งด่วน เพื่อให้ผู้ที่พลาดรถประจำทางคันหนึ่ง มีข้อมูลในการตัดสินใจไปรับประทานอาหารเช้าหรือทำกิจธุระอื่นก่อนที่รถคันถัดไปจะมาถึง นักศึกษาจึงต้องเขียนโปรแกรมเพื่อคำนวณว่าเมื่อรถโดยสารประจำทางสาย 1023 คันหนึ่งผ่านไป แล้ว ผู้โดยสารจะต้องรออีกประมาณกี่นาที รถประจำทางสาย 1023 คันถัดไปจึงจะมาถึงป้าย

#### ข้อมูลเข้า

ข้อมูลเข้ามีจำนวน  $N+1$  บรรทัด

- บรรทัดแรกเป็นจำนวนเต็มบวก  $N$  แทนปริมาณรถโดยสารประจำทางสาย 1023 ที่ผ่านป้ายอนุสาวรีย์ชัยสมรภูมิในช่วงเวลาเร่งด่วน โดยที่  $2 \leq N \leq 500$
- บรรทัดที่ 2 ถึงบรรทัดที่  $N+1$  เป็นจำนวนเต็มบวก  $H$   $M$  ซึ่ง  $H$  แทนชั่วโมง และ  $M$  แทนนาทีที่รถโดยสารประจำทางสาย 1023 มาถึงป้ายอนุสาวรีย์ชัยสมรภูมิ โดยที่  $6 \leq H \leq 8$  และ  $0 \leq M \leq 59$  และเป็นเวลาที่อยู่ในช่วง 6:00 – 8:00 น.

**หมายเหตุ** กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามขอบเขตเสมอ

#### ข้อมูลส่งออก

ข้อมูลส่งออกมีหนึ่งบรรทัด แสดงผลลัพธ์เป็นเลขจำนวนเต็มบวกหนึ่งจำนวน แทนเวลาเฉลี่ย (ในหน่วย นาที) ของการรอรถโดยสารประจำทางสาย 1023 ในช่วงเวลาเร่งด่วน ณ ป้ายโดยสารอนุสาวรีย์ชัยสมรภูมิ

#### ตัวอย่างที่ 1

ข้อมูลเข้า	ข้อมูลส่งออก
3	60
6 0	
7 0	
8 0	

#### ตัวอย่างที่ 2

ข้อมูลเข้า	ข้อมูลส่งออก
6	24
6 0	
6 20	
6 50	
7 10	
7 40	
8 0	

### ตัวอย่างที่ 3

ข้อมูลเข้า	ข้อมูลส่งออก
6	24
6 0	
6 10	
6 20	
7 30	
7 55	
8 0	

### ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
จำนวนชุดทดสอบ	10
คะแนนของแต่ละชุดทดสอบ	10
คะแนนเต็มทั้งสิ้น	100
เงื่อนไขในการรันโปรแกรม	โปรแกรมจะต้องประมวลผลชุดข้อมูลตัวอย่างที่ให้มาได้ถูกต้อง

### ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรมดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

## น้องเเมอเรียงบัตรตัวเลข (CardSorting)

น้องเเมอได้รับชุดบัตร  $N$  ใบ ( $1 \leq N \leq 9$ ) มาจากคุณครู ซึ่งหน้าบัตรแต่ละใบแสดงตัวเลขจำนวนเต็ม ที่มีค่าเป็นไปได้ตั้งแต่ 0 ถึง 999 คุณครูสั่งให้น้องเเมอนำบัตรตัวเลขทั้งหมดนั้นมาเรียงต่อกัน เพื่อให้ดูเหมือนเป็นเลขจำนวนเต็มหนึ่งจำนวนที่มีค่าน้อยที่สุดเท่าที่จะเป็นไปได้

ตัวอย่างเช่น ถ้าน้องเเมอได้รับชุดบัตร 3 ใบต่อไปนี้

703 4 20

น้องเเมอต้องเรียงบัตรให้ผลลัพธ์เป็นชุดตัวเลขดังนี้

204703

เนื่องจากหากเรียงแบบอื่นๆ (ทั้งหมดทุกวิธีที่เป็นไปได้) จะได้ชุดตัวเลขต่อไปนี้

703420, 703204, 470320, 420703, 207034 ซึ่งทุกแบบ ล้วนแต่มีค่ามากกว่า 204703 ดังนั้นคำตอบที่ถูกต้องของชุดบัตรตัวเลขสามใบนี้ ต้องเป็น 204703

ให้นักศึกษาเขียนโปรแกรมเพื่อช่วยน้องเเมอเรียงบัตรตัวเลขให้ถูกต้องตามที่คุณครูสั่ง

### ข้อมูลเข้า

ข้อมูลเข้ามี  $N + 1$  บรรทัด

บรรทัดแรก เป็นจำนวนบัตรตัวเลขที่น้องเเมอจะได้รับจากคุณครู ( $N$  ใบ เมื่อ  $N$  เป็นเลขจำนวนเต็มใดๆ ระหว่าง 1-9)

อีก  $N$  บรรทัดที่เหลือ เป็นตัวเลขที่แสดงอยู่บนหน้าบัตรแต่ละใบ

### หมายเหตุ

กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามรูปแบบ ขอบเขต และ เป็นไปตามเงื่อนไขที่โจทย์กำหนดเสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

ข้อมูลส่งออกมี 1 บรรทัด

เป็นชุดตัวเลขที่ปรากฏบนหน้าบัตร เรียงต่อกันตามลำดับที่โจทย์กำหนด (ไม่มีเว้นวรรค)

### ตัวอย่างที่ 1

ข้อมูลเข้า	ข้อมูลส่งออก
3	204703
703	
4	
20	





ตัวอย่างที่ 2

ข้อมูลเข้า	ข้อมูลส่งออก
5	18632507490
50	
186	
32	
74	
90	

ตัวอย่างที่ 3

ข้อมูลเข้า	ข้อมูลส่งออก
4	0557039
703	
9	
0	
55	

ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรมดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.8.0_144	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็นชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java