



ข้อสอบคอมพิวเตอร์โอลิมปิก สอน.

เวลาสอบ 9.00-12.00 น.

ข้อสอบมี 3 ข้อ ให้ทำทุกข้อ

ภาคอาหาร

โรงเรียนประจำแห่งหนึ่ง นักเรียนจะต้องเข้าแถวเพื่อรับภาคอาหารกลางวันตั้งแต่วันที่ 12:00 น. ของทุกวัน คุณครูจะให้ นักเรียนทุกชั้นเข้าแถวเดียวกัน โดยมีระเบียบในการเข้าแถวอยู่ดังนี้

เมื่อเริ่มต้น นักเรียนคนใดมาก่อน คุณครูจะให้ยืนที่หัวแถว และนักเรียนคนที่จะมาเข้าแถวคนต่อไป คุณครูจะ สำนวนตำแหน่งของนักเรียน โดยเริ่มต้นจากหัวแถว และจะแทรกนักเรียนคนนั้นเข้าไปในตำแหน่งต่อจากคนสุดท้าย ของนักเรียนในชั้นเรียนเดียวกัน แต่ถ้าไม่มีนักเรียนในชั้นเดียวกันอยู่ในแถว คุณครูจะให้นักเรียนคนนั้นไปต่อที่ท้ายแถว

การออกจากแถวเพื่อไปรับภาคอาหาร นักเรียนที่อยู่หัวแถวที่ได้ออกจากแถวก่อน และให้แสดงเลขประจำตัว นักเรียนที่ได้รับภาคอาหารแล้วไม่สามารถกลับเข้ามาในแถวได้อีก

ทั้งนี้คุณครูจะทราบเลขประจำตัวและชั้นเรียนของนักเรียนทุกคนอยู่แล้ว และนักเรียนทุกคนจะมีเลขประจำตัว ไม่ซ้ำกัน นักเรียนบางคนอาจไม่ได้ถูกเรียกมาเข้าแถว และ นักเรียนบางคนอาจจะไม่ได้ออกจากแถว

จึงเขียนโปรแกรมเพื่อจัดแถวเข้ารับภาคอาหารตามระเบียบของโรงเรียนแห่งนี้ แสดงลำดับการรับภาคอาหาร ของนักเรียน กรณีที่ไม่มีนักเรียนในแถวให้เขียนข้อความว่า “empty” และ ให้แสดงเลขประจำตัว นักเรียนที่ได้ออกจาก แถว

ข้อมูลนำเข้า

บรรทัดแรกมีจำนวนเต็มสองตัว N_c และ N_s แทนจำนวนชั้นเรียนและจำนวนนักเรียนตามลำดับ โดยที่ $1 \leq N_c \leq 10$ และ $1 \leq N_s \leq 1000$ ตัวเลขทั้งสองถูกคั่นด้วยช่องว่างหนึ่งช่อง

จากนั้นอีก N_s บรรทัดเก็บรายละเอียดของนักเรียนแต่ละคนด้วยจำนวนเต็มสองค่าคั่นด้วยช่องว่างหนึ่งช่อง คือ c และ s ซึ่งแทน หมายเลขชั้นเรียนและเลขประจำตัวของนักเรียนตามลำดับ โดยที่ $1 \leq c \leq N_c$ และ $1 \leq s \leq 10000$

การมาเข้าแถวและการนำนักเรียนออกจากหัวแถวเพื่อไปรับภาคอาหาร แต่ละบรรทัดจะมีรูปแบบคำสั่ง ดังต่อไปนี้

- | | |
|------|--|
| E id | เป็นการนำนักเรียนที่มีเลขประจำตัว id มาเข้าแถว |
| D | เป็นการนำนักเรียนที่อยู่หัวแถวออกจากแถว |
| X | เป็นการระบุว่าเป็นคำสั่งสุดท้าย |



ข้อมูลส่งออก

แต่ละบรรทัดแสดงเลขประจำตัวนักเรียนที่ถูกนำออกจากแถวเพื่อรับอาหารตามลำดับ โดยบรรทัดสุดท้ายให้ใส่จำนวนเต็มศูนย์ ตัวอย่าง

ข้อมูลนำเข้า	ข้อมูลส่งออก
2 6	41
1 41	201
1 42	202
1 43	203
2 201	42
2 202	0
2 203	
E 41	
E 201	
D	
E 202	
E 42	
E 43	
D	
E 203	
D	
D	
D	
X	

ข้อกำหนด

หัวข้อ	เงื่อนไข
ชื่อแฟ้มข้อมูลนำเข้า	queue.in
ชื่อแฟ้มข้อมูลส่งออก	queue.out
ระยะเวลาสูงสุดที่ใช้ในการประมวลผล ต่อชุดทดสอบหนึ่งชุด	2 วินาที
หน่วยความจำสูงสุดที่ใช้ในการประมวลผล ต่อชุดทดสอบหนึ่งชุด	64 กิโลไบต์
จำนวนชุดทดสอบ (แยกทดสอบทดสอบแต่ละชุด)	10
คะแนนสำหรับชุดทดสอบแต่ละชุด	10
คะแนนสูงสุดของโจทย์	100
เงื่อนไขการการรับโปรแกรม	โปรแกรมจะต้องประมวลผลข้อมูลตามตัวอย่างที่ให้มาได้
การระบุภาษา	จะต้องระบุภาษาที่ใช้เขียนในส่วนหัวของโปรแกรมตามที่โจทย์กำหนด



ข้อมูลคำสั่งเพิ่มเติม

ส่วนหัวของโปรแกรมเพื่อระบุชื่อโจทย์

ภาษา C

```
/*  
LANG: C  
TASK: PLATE  
AUTHOR: YourName YourLastname  
CENTER: YourCenter  
*/
```

ภาษา C++

```
/*  
LANG: C++  
TASK: PLATE  
AUTHOR: YourName YourLastname  
CENTER: YourCenter  
*/
```



โซ่คำ (Word Chain)

โซ่คำคือลำดับของคำที่มีจำนวนอักขระเท่ากันและแต่ละคำที่มีลำดับติดกันจะต้องมีตำแหน่งที่มีตัวอักขระต่างกันไม่เกินสองตำแหน่ง เช่น HEAD และ HEAP จะต่างกันตำแหน่งเดียวคือ D และ P ในตำแหน่งตัวอักขระที่ 4 ของคำ ในขณะที่ REAR กับ BAER จะมีตำแหน่งต่างกัน 3 ตำแหน่ง คือ ตำแหน่งที่ 1 (R กับ B) ตำแหน่งที่ 2 (E และ A) และตำแหน่งที่ 3 (A และ E)

ตัวอย่างของโซ่คำที่ต่อเนื่องได้แก่ HEAD HEAP LEAP TEAR REAR และ EGG EAG GAE GAP TAP TIN

ตัวอย่างของโซ่คำที่ขาดได้แก่ LEAP TEAR REAR BAER BAET BEEP ซึ่งจะขาดที่ คำว่า BAER

ให้ชุดของโซ่คำมาชุดหนึ่ง จงเขียนโปรแกรมเพื่อหาคำสุดท้ายในโซ่คำ ก่อนที่โซ่คำจะขาด

ข้อมูลนำเข้า

อ่านข้อมูลจากแฟ้ม

บรรทัดแรก เก็บจำนวนเต็ม L แทนจำนวนตัวอักขระของแต่ละคำ โดยที่ $3 \leq L \leq 1000$

บรรทัดที่สอง เก็บจำนวนเต็ม N แทนจำนวนคำทั้งหมดในแฟ้มข้อมูล โดยที่ $1 \leq N \leq 30000$

บรรทัดที่ $3..(N+2)$ เก็บลำดับของคำที่มีจำนวนตัวอักขระ L ตัว แต่ละบรรทัดเก็บคำที่เขียนด้วยตัวอักษร ('A' ถึง 'Z') ที่เป็นตัวพิมพ์ใหญ่

ข้อมูลส่งออก

แฟ้มผลลัพธ์ มี 1 บรรทัด เก็บคำสุดท้ายของโซ่คำชุดแรก

ตัวอย่าง

ข้อมูลนำเข้า	ข้อมูลส่งออก
4 12 HEAD HEAP LEAP TEAR REAR BAER BAET BEEP JEEP JOIP JEIP AEIO	REAR



ข้อกำหนด

หัวข้อ	เงื่อนไข
ชื่อแฟ้มข้อมูลนำเข้า	chain.in
ชื่อแฟ้มข้อมูลส่งออก	chain.out
ระยะเวลาสูงสุดที่ใช้ในการประมวลผล ต่อชุดทดสอบหนึ่งชุด	2 วินาที
หน่วยความจำสูงสุดที่ใช้ในการประมวลผล ต่อชุดทดสอบหนึ่งชุด	64 กิโลไบต์
จำนวนชุดทดสอบ	20
คะแนนสำหรับชุดทดสอบแต่ละชุด	5
คะแนนสูงสุดของโจทย์	100
เงื่อนไขการการรับโปรแกรม	โปรแกรมจะต้องประมวลผลข้อมูลตามตัวอย่างที่ให้มาได้
การระบุภาษา	จะต้องระบุภาษาที่ใช้เขียนในส่วนหัวของโปรแกรมตามที่โจทย์กำหนด

ข้อมูลคำสั่งเพิ่มเติม

ส่วนหัวของโปรแกรมเพื่อระบุชื่อโจทย์

ภาษา C

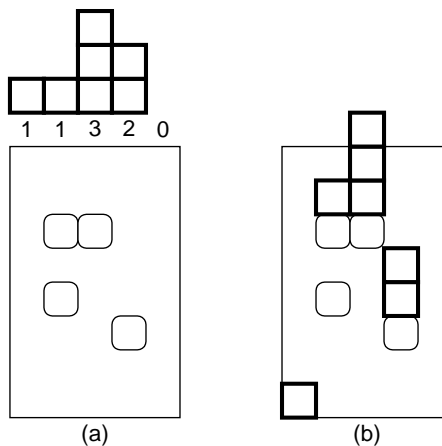
```
/*  
Lang: C  
TASK: CHAIN  
AUTHOR: YourName YourLastName  
CENTER: YourCenter  
*/
```

ภาษา C++

```
/*  
Lang: C++  
TASK: CHAIN  
AUTHOR: YourName YourLastName  
CENTER: YourCenter  
*/
```

ก่อนอิฐ

ในตารางเกมขนาด N แถว M คอลัมน์ โดยในตารางมีสิ่งกีดขวางวางเอาไว้ ด้านบนมีก้อนอิฐหลายๆ ก้อนที่กำลังจะหล่นลงมา ตัวอย่างของเกมแสดงดังรูปที่ 1 เป็นตารางขนาด $(N=8) \times (M=5)$ ซึ่งสถานะเริ่มต้นของตารางเกมแสดงในรูป 1 (a) และเมื่อเกมได้ประมวลผลแล้ว ซึ่งก็คืออิฐตกจากด้านบนลงสู่ด้านล่าง จะเห็นว่าอิฐจะมีการตกค้างที่สิ่งกีดขวาง และผลลัพธ์หลังจากประมวลเสร็จสิ้นดังแสดงในรูปที่ 1 (b)



รูปที่ 1 a. แสดงตารางเกมเริ่มต้น b. แสดงผลลัพธ์หลังจากที่อิฐด้านบนหล่นมาหมดแล้ว

จงเขียนโปรแกรมเพื่อรับตารางเกมเริ่มต้นและจำนวนอิฐที่จะตกลงมาในแต่ละคอลัมน์ ให้ประมวลผลก้อนอิฐทุกก้อน โดยมีเงื่อนไขดังนี้ (1) ถ้าก้อนอิฐตกลงมาแล้วพบสิ่งกีดขวางที่อยู่ในตารางเกม ก็จะค้างอยู่ ณ ตำแหน่งที่พบสิ่งกีดขวาง (2) ถ้าก้อนอิฐไม่พบสิ่งกีดขวางจะตกลงมาอยู่แถวล่างสุด เมื่อประมวลผลครบทุกก้อนอิฐให้แสดงผลสถานะของตารางเกม

ข้อมูลนำเข้า

รับข้อมูลจากแฟ้มข้อมูลนำเข้า

บรรทัดแรก จะระบุจำนวนเต็มสองจำนวน N และ M โดยที่ $1 \leq N \leq 20$ และ $1 \leq M \leq 20$

N บรรทัดถัดไป จะเป็นการระบุตารางเกม โดยในบรรทัดที่ $1 + I$ จะเป็นข้อมูลของตารางเกมแถวที่ I ซึ่งจะระบุเป็นสายอักขระความยาว M ตัวอักขระ ที่มีรูปแบบดังนี้ (1) เครื่องหมายจุด '.' แทนช่องที่ว่างในตารางเกม และ (2) ตัวอักษร 'o' ((ตัวพิมพ์ใหญ่โอ) แทนช่องที่มีสิ่งกีดขวางอยู่

บรรทัดสุดท้าย ประกอบด้วยตัวเลข M ตัวคือ $a_1, a_2, a_3, \dots, a_M$ แต่ละตัวคั่นด้วยช่องว่างหนึ่งช่อง ตัวเลข a_j คือจำนวนก้อนอิฐที่จะตกลงมาในคอลัมน์ที่ j โดยที่ $0 \leq a_j \leq 20$



ข้อมูลส่งออก

ให้เขียนผลลัพธ์ออกทางแฟ้ม โดยให้ระบุตารางเกมผลลัพธ์ในรูปแบบเดียวกับในแฟ้มข้อมูลนำเข้า ให้ใช้เครื่องหมาย '#' แทนก้อนอิฐที่อยู่ในตาราง

ตัวอย่าง

ข้อมูลนำเข้า	ข้อมูลส่งออก
8 5OO..O.. ...O. 1 1 3 2 0	..#.. .##.. .OO.. ...#. .O.#. ...O. #....

ข้อกำหนด

หัวข้อ	เงื่อนไข
ชื่อแฟ้มข้อมูลนำเข้า	brick.in
ชื่อแฟ้มข้อมูลส่งออก	brick.out
ระยะเวลาสูงสุดที่ใช้ในการประมวลผล ต่อชุดทดสอบหนึ่งชุด	2 วินาที
หน่วยความจำสูงสุดที่ใช้ในการประมวลผล ต่อชุดทดสอบหนึ่งชุด	64 กิโลไบต์
จำนวนชุดทดสอบ	10
คะแนนสำหรับชุดทดสอบแต่ละชุด	10
คะแนนสูงสุดของโจทย์	100
เงื่อนไขการการรับโปรแกรม	โปรแกรมจะต้องประมวลผลข้อมูลตามตัวอย่างที่ให้มาได้
การระบุภาษา	จะต้องระบุภาษาที่ใช้เขียนในส่วนหัวของโปรแกรมตามที่โจทย์กำหนด

ข้อมูลคำสั่งเพิ่มเติม

ส่วนหัวของโปรแกรมเพื่อระบุชื่อโจทย์

ภาษา C	ภาษา C++
/ *	/ *



<pre>TASK: BRICK LANG: C AUTHOR: YourName YourLastName CENTER: YourCenter */</pre>	<pre>TASK: BRICK LANG: C++ AUTHOR: YourName YourLastName CENTER: YourCenter */</pre>
--	--