



Wall

เจียนเจียกำลังสร้างกำแพงอันหนึ่งด้วยการวางก้อนอิฐขนาดเท่ากันซ้อน ๆ กัน กำแพงอันนี้ประกอบด้วยก้อนอิฐ n คอลัมน์ แต่ละคอลัมน์ถูกกำกับด้วยตัวเลข 0 ถึง $n - 1$ จากซ้ายไปขวา โดยแต่ละคอลัมน์อาจมีความสูงไม่เท่ากัน ความสูงของแต่ละคอลัมน์คือจำนวนก้อนอิฐที่อยู่ในคอลัมน์นั้น

เจียนเจียใช้วิธีต่อไปนี้ในการสร้างกำแพง เริ่มแรกในทุกคอลัมน์จะไม่มีก้อนอิฐอยู่ จากนั้น เจียนเจียจะ **เพิ่ม** หรือ **ลด** ก้อนอิฐเป็นจำนวน k รอบ การสร้างกำแพงจะจบลงเมื่อทำงานครบทั้ง k รอบแล้ว ในแต่ละรอบ เจียนเจียจะได้รับช่วงของคอลัมน์ของอิฐที่ต่อกัน และค่าความสูง h ซึ่งเขาจะดำเนินการดังต่อไปนี้

- ในรอบที่เป็นการ **เพิ่ม** ก้อนอิฐ กรณีที่คอลัมน์ใดในช่วงที่กำหนดมีก้อนอิฐจำนวนน้อยกว่า h ก้อน เจียนเจียจะวางก้อนอิฐเพิ่มในคอลัมน์นั้นเพื่อให้ก้อนอิฐในช่วงนั้นมีจำนวน h ก้อนพอดี แต่เขาจะไม่ทำอะไรกับคอลัมน์ที่มีจำนวนก้อนอิฐมากกว่าหรือเท่ากับ h ก้อนอยู่แล้ว
- ในรอบที่เป็นการ **ลด** ก้อนอิฐ กรณีที่คอลัมน์ใดในช่วงที่กำหนดมีก้อนอิฐจำนวนมากกว่า h ก้อน, เจียนเจียจะเอาก้อนอิฐออกจนกว่าคอลัมน์นั้นจะมีก้อนอิฐ h ก้อนพอดี แต่เขาจะไม่ทำอะไรกับคอลัมน์ที่มีจำนวนก้อนอิฐน้อยกว่าหรือเท่ากับ h ก้อนอยู่แล้ว

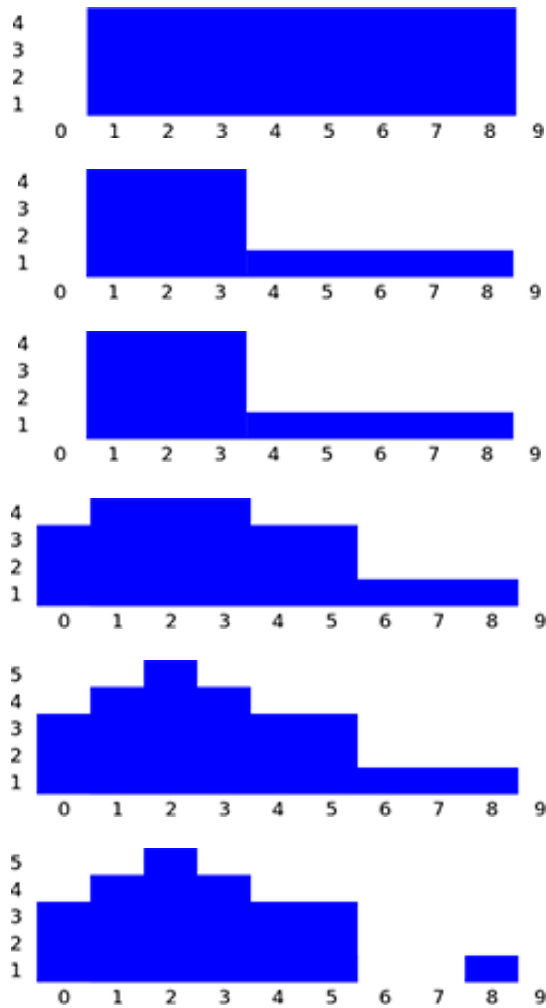
งานของคุณคือการคำนวณหารูปร่างสุดท้ายของกำแพง

ตัวอย่าง

เราสมมติว่ากำแพงที่จะสร้างมี 10 คอลัมน์ และมีจำนวนรอบของการสร้างกำแพงอยู่ 6 รอบ ช่วงของคอลัมน์ที่กำหนดให้ในตารางต่อไปนี้จะรวมถึงคอลัมน์แรกและคอลัมน์สุดท้ายด้วย รูปร่างของกำแพงที่เกิดขึ้นหลังจากการสร้างในแต่ละรอบถูกแสดงอยู่ด้านล่าง

รอบ	ประเภทของรอบ	ช่วงของคอลัมน์	ค่าความสูง
0	เพิ่ม	คอลัมน์ที่ 1 ถึง 8	4
1	ลด	คอลัมน์ที่ 4 ถึง 9	1
2	ลด	คอลัมน์ที่ 3 ถึง 6	5
3	เพิ่ม	คอลัมน์ที่ 0 ถึง 5	3
4	เพิ่ม	คอลัมน์ที่ 2	5
5	ลด	คอลัมน์ที่ 6 ถึง 7	0

เนื่องจากในตอนแรกจะไม่มีอิฐอยู่ในแต่ละคอลัมน์ หลังจากรอบที่ 0 คอลัมน์ตั้งแต่ 1 ถึง 8 จะมีอิฐอยู่ 4 ก้อน คอลัมน์ที่ 0 และ 9 ยังคงไม่มีก้อนอิฐอยู่ ในรอบที่ 1 ก้อนอิฐจะถูกเอ้อออกจากคอลัมน์ที่ 4 ถึง 8 จนกว่าคอลัมน์ดังกล่าวจะมีก้อนอิฐ 1 ก้อน ส่วนคอลัมน์ที่ 9 ยังคงไม่มีก้อนอิฐ คอลัมน์ที่ 0 ถึง 3 ซึ่งไม่อยู่ในช่วงที่กำหนดจะไม่มีการเปลี่ยนแปลง รอบที่ 2 ไม่มีการเปลี่ยนแปลงใด ๆ เกิดขึ้นเนื่องจากคอลัมน์ที่ 3 ถึง 6 มีก้อนอิฐไม่เกิน 5 ก้อนอยู่แล้ว หลังจากรอบที่ 3 จำนวนของก้อนอิฐในคอลัมน์ที่ 0, 4 และ 5 ถูกเพิ่มเป็น 3 ก้อน หลังจากรอบที่ 4 คอลัมน์ที่ 2 มีจำนวนอิฐเป็น 5 ก้อน และในรอบที่ 5 เป็นการเอาก้อนอิฐทั้งหมดออกจากคอลัมน์ที่ 6 และ 7



งานของคุณ

เมื่อให้รายละเอียดของการสร้างกำแพงทั้ง k รอบ, คุณจะต้องคำนวณจำนวนก้อนอิฐในแต่ละคอลัมน์ หลังจากการสร้างกำแพงเสร็จสมบูรณ์ โดยคุณต้องเขียนฟังก์ชัน `buildWall`

- `buildWall(n, k, op, left, right, height, finalHeight)`
 - n : จำนวนของคอลัมน์ของกำแพง
 - k : จำนวนรอบ
 - op : อาร์เรย์ขนาด k โดย $op[i]$ เป็นประเภทของรอบที่ i กล่าวคือจะมีค่าเป็น 1 สำหรับรอบที่เพิ่มก้อนอิฐ และ 2 สำหรับรอบที่ลดก้อนอิฐ โดยที่ $0 \leq i \leq k - 1$.
 - $left$ และ $right$: อาร์เรย์ขนาด k โดยช่วงของคอลัมน์ที่กำหนดให้ในรอบ i เริ่มจากคอลัมน์ที่ $left[i]$ และสิ้นสุดในคอลัมน์ที่ $right[i]$ (โดยรวมทั้งคอลัมน์ที่ $left[i]$ และ $right[i]$ ด้วย) โดยที่ $0 \leq i \leq k - 1$ เราจะรับประกันว่า $left[i] \leq right[i]$ เสมอ
 - $height$: อาร์เรย์ขนาด k โดย $height[i]$ เป็นค่าความสูงสำหรับรอบที่ i โดย $0 \leq i \leq k - 1$
 - $finalHeight$: อาร์เรย์ขนาด n คุณต้องให้ผลลัพธ์โดยนำจำนวนก้อนอิฐของคอลัมน์ที่ i ใส่ใน $finalHeight[i]$ โดย $0 \leq i \leq n - 1$

งานย่อย

ในแต่ละงานย่อย ค่าสูงสุดของค่าความสูงสำหรับทุกรอบจะเป็นจำนวนเต็มที่ไม่เป็นลบและมีค่าน้อยกว่าหรือเท่ากับ **100,000**

งานย่อย	คะแนน	n	k	หมายเหตุ
1	8	$1 \leq n \leq 10,000$	$1 \leq k \leq 5,000$	ไม่มีข้อกำหนดอื่น
2	24	$1 \leq n \leq 100,000$	$1 \leq k \leq 500,000$	รอบที่เพิ่มก้อนอิฐทั้งหมดจะเกิดขึ้นก่อนรอบที่ลดก้อนอิฐ
3	29	$1 \leq n \leq 100,000$	$1 \leq k \leq 500,000$	ไม่มีข้อกำหนดอื่น
4	39	$1 \leq n \leq 2,000,000$	$1 \leq k \leq 500,000$	ไม่มีข้อกำหนดอื่น

รายละเอียดการเขียนโปรแกรม

คุณต้องส่งไฟล์หนึ่งไฟล์เท่านั้น โดยอาจจะเป็นไฟล์ชื่อ `wall.c`, `wall.cpp` หรือ `wall.pas` โดยที่ไฟล์นี้จะต้องมีฟังก์ชันต่าง ๆ ที่ทำงานได้ตามที่ระบุไว้ข้างต้น คุณจะต้อง `include header wall.h` ในไฟล์ดังกล่าวถ้าคุณใช้ภาษา C/C++

โปรแกรม C/C++

```
void buildWall(int n, int k, int op[], int left[], int right[], int height[], int finalHeight[]);
```

โปรแกรม Pascal

```
procedure buildWall(n, k : longint; op, left, right, height : array of longint; var finalHeight : array of longint);
```

เกรตเตอร์ตัวอย่าง

เกรตเตอร์ตัวอย่างจะอ่านอินพุตในรูปแบบต่อไปนี้:

- บรรทัดที่ 1: n, k
- บรรทัดที่ $2 + i$ โดย $(0 \leq i \leq k - 1)$: $op[i], left[i], right[i], height[i]$