



## กลกลอนนกกก

(1 วินาที, 512 MB)

ณ ฟีนป่าอันอุดมสมบูรณ์ใจกลางอุทยานแห่งชาติเขาใหญ่ นกกกได้ชื่อว่าเป็นสัญลักษณ์ของรักแท้ เมื่อถึงฤดูเกี้ยวพาราสี นกกกตัวเมียคู่หนึ่งตั้งใจจะขับ "บทกลอนรัก" ร่วมกัน เพื่อประกาศความรักของพวกมันให้ก้องกังวานไปทั่วฟีนป่า

บทกลอนนี้มีความยาวทั้งสิ้น  $N$  คำ ที่เรียงลำดับต่อเนื่องกันไปตั้งแต่คำที่ 0 ถึงคำที่  $N - 1$  โดยนกกกที่ขับร้องคำที่  $i$  จะต้องเปล่งเป็นเสียงที่มีค่าระดับเสียงเท่ากับ  $A[i]$  ซึ่งนกกกทั้งสองตัวจะต้องช่วยกันขับบทกลอนนี้ให้จบสมบูรณ์ โดยมีเงื่อนไขว่า "คำแต่ละคำในบทกลอนจะต้องมีนกเพียงแค่ตัวเดียวเท่านั้นที่ เป็นผู้ขับ" ทั้งนี้ทั้งนกกกทั้งสองสามารถผลัดกันขับบทกลอนสลับไปมาทีละคำ หรือหลายคำติดกันก็ได้ ตราบใดที่ยังคงลำดับของคำตามบทกลอนอย่างถูกต้อง

อย่างไรก็ตาม ในการขับบทกลอนแต่ละครั้งจะทำให้นกมีอาการเหนื่อยล้าซึ่งเกิดจากการเปลี่ยนระดับเสียงในคำต่อไป โดยค่า "ความเหนื่อยล้า" ของนกแต่ละตัวจะถูกคำนวณจากผลรวมของค่าความต่างของระดับเสียงระหว่างคำกลอนที่ตนเองเพิ่งขับจบไปกับคำกลอนถัดไปที่ตนเองขับ

ยกตัวอย่างเช่น หากนกกกตัวหนึ่งขับคำที่มีค่าระดับเสียงเป็น 5, 9 และ 2 ตามลำดับ (โดยอาจมีนกกอีกตัวขับกลอนแทรกระหว่างคำเหล่านี้) ค่าความเหนื่อยล้าสะสมของนกตัวนี้จะมีค่าเท่ากับ  $|5 - 9| + |9 - 2| = 11$  ทั้งนี้ หากนกกตัวใดตัวหนึ่งขับกลอนเพียงแค่คำเดียว หรือไม่ได้ขับกลอนเลย จะถือว่าค่าความเหนื่อยล้าสะสมของนกตัวนั้นมีค่าเป็น 0

นกกกตัวเมียคู่นี้จึงต้องการวางแผนแบ่งเนื้อร้องให้กันและกันเพื่อให้ผลรวมความเหนื่อยล้าสะสมของนกทั้งสองตัวมีค่าน้อยที่สุดเท่าที่จะเป็นไปได้ จงหาวิธีแบ่งคำในบทกลอนให้ทั้งนกทั้งสองตัวนี้

## รายละเอียดการเขียนโปรแกรม

คุณจะต้องเขียนฟังก์ชันต่อไปนี้ เพื่อช่วยนกกกทั้งสองตัวแบ่งเนื้อร้อง:

```
vector<int> assign_notes(vector<int> A);
```

พารามิเตอร์:

- $A$ : อาร์เรย์ความยาว  $N$  เก็บค่าระดับเสียงของคำในบทกลอนตามลำดับ

**ค่าที่ต้องคืนกลับ (Return Value):** ฟังก์ชันจะต้องคืนค่าเป็นอาร์เรย์  $V$  ความยาว  $N$  ซึ่งเป็นการระบุหน้าที่ซบกลอนในแต่ละคำ โดยมีข้อกำหนดดังนี้:

- $V[i] = 0$  หมายความว่า ให้ **นกกกตัวผู้** เป็นตัวซบกลอนคำที่  $i$
- $V[i] = 1$  หมายความว่า ให้ **นกกกตัวเมีย** เป็นตัวซบกลอนคำที่  $i$

(หมายเหตุ: ในกรณีที่มีวิธีการแบ่งหน้าที่ซบกลอนที่ทำให้ผลรวมความเหนื่อยล้าสะสมที่น้อยที่สุดได้หลายวิธี คุณสามารถคืนค่าอาร์เรย์คำตอบของวิธีการใดก็ได้)

## ตัวอย่าง

พิจารณาการเรียกใช้งานต่อไปนี้

```
assign_notes([1, 5, 6, 3, 2])
```

หากให้นกกกตัวผู้ซบกลอนแค่ตัวเดียว ผลรวมความเหนื่อยล้าสะสมจะสูงถึง

$$|1 - 5| + |5 - 6| + |6 - 3| + |3 - 2| = 4 + 1 + 3 + 1 = 9$$

แต่วิธีที่ดีที่สุดในการแบ่งหน้าที่ซบกลอนคือ:

- **นกกกตัวผู้:** รับหน้าที่ซบกลอนคำที่ 1 และ 2 ซึ่งมีระดับเสียง 5 และ 6 ตามลำดับ ค่าความเหนื่อยล้าสะสมของนกตัวผู้จะเท่ากับ  $|5 - 6| = 1$
- **นกกกตัวเมีย:** รับหน้าที่ซบกลอนคำที่ 0, 3 และ 4 ซึ่งมีระดับเสียง 1, 3 และ 2 ตามลำดับ ค่าความเหนื่อยล้าสะสมของนกตัวเมียจะเท่ากับ  $|1 - 3| + |3 - 2| = 3$

ผลรวมความเหนื่อยล้าสะสมของนกทั้งสองตัวจะลดลงเหลือเพียง  $1 + 3 = 4$  ซึ่งเป็นผลรวมความเหนื่อยล้าสะสมที่น้อยที่สุดที่เป็นไปได้สำหรับบทกลอนนี้

ดังนั้น ฟังก์ชันนี้จะคืนค่าเป็น  $[1, 0, 0, 1, 1]$  ในทางกลับกัน ฟังก์ชันนี้จะคืนค่าคำตอบเป็น  $[0, 1, 1, 0, 0]$  ก็ได้ หากสลับการซบร้องระหว่างเพศผู้และเพศเมีย

## ขอบเขตของข้อมูล

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq A[i] \leq 10^9$  สำหรับทุก  $0 \leq i \leq N - 1$

## ปัญหาย่อย

ปัญหาย่อย	คะแนน	เงื่อนไขเพิ่มเติม
1	7	$N \leq 20$
2	11	$A[i] \leq A[i + 1]$ สำหรับทุก $0 \leq i < N - 1$
3	16	$N \leq 100$
4	18	ค่าที่แตกต่างกันใน $A$ มีไม่เกิน 20 ค่า
5	25	$N \leq 1000$
6	23	ไม่มีเงื่อนไขเพิ่มเติม

## เกรตเตอร์ตัวอย่าง

ไฟล์เกรตเตอร์จะรับข้อมูลนำเข้าในรูปแบบต่อไปนี้

- บรรทัดที่ 1: รับค่า  $N$
- บรรทัดที่ 2: รับค่า  $A[0]$  ถึง  $A[N - 1]$  ตามลำดับ

หลังจากนั้นไฟล์เกรตเตอร์จะเรียกฟังก์ชัน `assign_notes` ของผู้เข้าแข่งขัน แล้วแสดงผลลัพธ์ที่ได้จากฟังก์ชันออกทางหน้าจอ