

เปรียบเทียบพืชพรรณ (plants)

นักพฤกษศาสตร์ชื่อว่าเฮเซลได้เยี่ยมชมนิทรรศการที่ Singapore Botanical Gardens ในนิทรรศการนี้ ต้นไม้ทั้งสิ้น n ต้นที่มีความสูงแตกต่างกันทั้งหมด และถูกวางเรียงกันเป็นวงกลม ต้นไม้เหล่านี้มีหมายเลขระบุจาก 0 จนถึง $n - 1$ เรียงตามเข็มนาฬิกา โดยที่ต้นไม้ต้นที่ $n - 1$ จะอยู่ติดกับต้นไม้ที่ 0

สำหรับต้นไม้แต่ละต้น ต้นที่ i (สำหรับ $0 \leq i \leq n - 1$) เฮเซลจะนำต้นไม้ต้นดังกล่าวไปเปรียบเทียบกับ ต้นไม้ $k - 1$ ต้นถัดไปตามเข็มนาฬิกา ทีละต้น แล้วจึงเขียนบันทึกค่า $r[i]$ ซึ่งระบุว่าในบรรดาต้นไม้ $k - 1$ ต้นนั้น จะมีกี่ต้นที่สูงกว่าต้นที่ i ดังนั้นแล้ว ค่า $r[i]$ แต่ละค่าจะขึ้นกับความสูงสัมพัทธ์ของต้นไม้ k ต้นที่อยู่ติดกัน

ยกตัวอย่างเช่น ให้ $n = 5$, $k = 3$ และ $i = 3$ ต้นไม้ $k - 1 = 2$ ต้นที่อยู่ถัดไปตามเข็มนาฬิกาจากต้นที่ $i = 3$ ก็คือต้นที่ 4 และต้นที่ 0 ถ้าต้นที่ 4 สูงกว่าต้นที่ 3 และต้นที่ 0 เตี้ยกว่าต้นที่ 3 แล้วเฮเซลจะบันทึกว่า $r[3] = 1$

คุณสามารถสมมติได้ว่าเฮเซลจะบันทึกข้อมูลค่า $r[i]$ อย่างถูกต้องทั้งหมด ดังนั้นแล้วจะมีรูปแบบของความสูงของต้นไม้ทั้งหมดที่เป็นไปได้อย่างน้อยหนึ่งรูปแบบ ที่สอดคล้องกับค่าที่เฮเซลบันทึกข้างต้น

คุณได้รับมอบหมายให้เปรียบเทียบความสูงของต้นไม้ทั้งสิ้น q คู่ แต่น่าเสียดายที่คุณไม่มีโอกาสเข้าถึงนิทรรศการนี้ แหล่งข้อมูลเดียวของคุณเกี่ยวกับนิทรรศการดังกล่าวคือสมุดบันทึกของเฮเซล ซึ่งจะระบุค่าของ k และลำดับค่าของ $r[0], \dots, r[n - 1]$

หากเราต้องนำต้นไม้ต้นที่ x และต้นที่ y (ซึ่งแตกต่างกัน) มาเปรียบเทียบกันแล้ว จงหาว่าหนึ่งในสามสถานการณ์ต่อไปนี้ ข้อใดเป็นจริง:

- ต้นไม้ต้นที่ x สูงกว่าต้นที่ y อย่างแน่นอน: สำหรับรูปแบบใด ๆ ของความสูงของต้นไม้ (ซึ่งมีค่าไม่ซ้ำกัน) $h[0], \dots, h[n - 1]$ ที่สอดคล้องกับอาร์เรย์ r แล้วจะได้ว่า $h[x] > h[y]$ เสมอ
- ต้นไม้ต้นที่ x เตี้ยกว่าต้นที่ y อย่างแน่นอน: สำหรับรูปแบบใด ๆ ของความสูงของต้นไม้ (ซึ่งมีค่าไม่ซ้ำกัน) $h[0], \dots, h[n - 1]$ ที่สอดคล้องกับอาร์เรย์ r แล้วจะได้ว่า $h[x] < h[y]$ เสมอ
- ไม่สามารถสรุปได้ว่าต้นไม้สองต้นดังกล่าว ต้นไม้ต้นใดสูงกว่ากัน: นั่นแปลว่าสถานการณ์สองข้อข้างต้นไม่เป็นจริง

รายละเอียดการเขียนโปรแกรม

คุณจะต้องเขียนฟังก์ชันต่อไปนี้:

```
void init(int k, int[] r)
```

- k : จำนวนต้นไม้ที่อยู่ติดกันซึ่งความสูงของต้นไม้เหล่านั้นจะส่งผลถึงค่า $r[i]$ แต่ละจำนวนที่บันทึกได้

- r : อาร์เรย์ความยาว n โดยที่ $r[i]$ ระบุจำนวนของต้นไม้ที่สูงกว่าต้นไม้ที่ i ในบรรดาต้นไม้ $k - 1$ ถัดไปตามเข็มนาฬิกา
- ฟังก์ชันนี้จะถูกเรียกใช้เพียงครั้งเดียวพอดี และจะเกิดขึ้นก่อน การเรียกใช้ฟังก์ชัน `compare_plants`

```
int compare_plants(int x, int y)
```

- x, y : หมายเลขของต้นไม้สองต้นที่จะถูกนำมาเปรียบเทียบกัน
- ฟังก์ชันนี้จะต้องคืนค่าหนึ่งในสามค่าดังนี้:
 - 1 ถ้าต้นไม้ต้นที่ x สูงกว่าต้นไม้ที่ y อย่างแน่นอน
 - -1 ถ้าต้นไม้ต้นที่ x ต่ำกว่าต้นไม้ที่ y อย่างแน่นอน
 - 0 ถ้าสรุปไม่ได้ว่าต้นไม้ต้นใดสูงกว่ากัน
- ฟังก์ชันนี้จะถูกเรียกใช้ q ครั้งพอดี

ตัวอย่าง

ตัวอย่างที่ 1

พิจารณาการเรียกต่อไปนี้:

```
init(3, [0, 1, 1, 2])
```

สมมติว่าเกรดเดอร์เรียกใช้ฟังก์ชัน `compare_plants(0, 2)` เนื่องจาก $r[0] = 0$ เราจึงสรุปได้ทันทีว่าต้นไม้ต้นที่ 2 จะไม่สูงกว่าต้นไม้ที่ 0 ดังนั้นแล้วฟังก์ชันนี้จะต้องคืนค่า 1

สมมติว่าถัดมา เกรดเดอร์เรียกใช้ฟังก์ชัน `compare_plants(1, 2)` สำหรับรูปแบบใด ๆ ของความสูงต้นไม้ที่สอดคล้องกับเงื่อนไขที่กำหนดให้ ต้นไม้ต้นที่ 1 จะเตี้ยกว่าต้นไม้ที่ 2 ดังนั้นแล้วฟังก์ชันนี้จะต้องคืนค่า -1

ตัวอย่างที่ 2

พิจารณาการเรียกต่อไปนี้:

```
init(2, [0, 1, 0, 1])
```

สมมติว่าเกรดเดอร์เรียกใช้ฟังก์ชัน `compare_plants(0, 3)` เนื่องจาก $r[3] = 1$ เราจึงสรุปได้ทันทีว่าต้นไม้ต้นที่ 0 จะสูงกว่าต้นไม้ที่ 3 ดังนั้นแล้วฟังก์ชันนี้จะต้องคืนค่า 1

สมมติว่าถัดมา เกรดเดอร์เรียกใช้ฟังก์ชัน `compare_plants(1, 2)` รูปแบบของความสูงของต้นไม้สองรูปแบบ ได้แก่ $[3, 1, 4, 2]$ และ $[3, 2, 4, 1]$ ต่างก็สอดคล้องกับค่าที่จัดบันทึกโดยเฮเซล เนื่องจากต้นไม้ต้นที่ 1 เตี้ยกว่าต้นไม้ที่ 3 ในรูปแบบของความสูงรูปแบบหนึ่ง แต่ต้นไม้ต้นที่ 1 สูงกว่าต้นไม้ที่ 3 ในอีกรูปแบบหนึ่ง ดังนั้นแล้วฟังก์ชันนี้จะต้องคืนค่า 0

ข้อจำกัด

- $2 \leq k \leq n \leq 200\,000$
- $1 \leq q \leq 200\,000$
- $0 \leq r[i] \leq k - 1$ (สำหรับ $0 \leq i \leq n - 1$)
- $0 \leq x < y \leq n - 1$
- จะมีรูปแบบของความสูงของต้นไม้ (ที่มีค่าไม่ซ้ำกัน) อย่างน้อย 1 รูปแบบที่สอดคล้องกับอาร์เรย์ r

ปัญหาย่อย

1. (5 คะแนน) $k = 2$
2. (14 คะแนน) $n \leq 5000, 2 \cdot k > n$
3. (13 คะแนน) $2 \cdot k > n$
4. (17 คะแนน) ค่าตอบที่ถูกต้องของการเรียกใช้ฟังก์ชัน `compare_plants` แต่ละครั้งคือ 1 หรือ -1
5. (11 คะแนน) $n \leq 300, q \leq \frac{n \cdot (n-1)}{2}$
6. (15 คะแนน) $x = 0$ สำหรับแต่ละครั้งที่เรียกใช้ฟังก์ชัน `compare_plants`
7. (25 คะแนน) ไม่มีเงื่อนไขใด ๆ เพิ่มเติม

ตัวอย่างเกรดเดอร์

เกรดเดอร์ตัวอย่างจะอ่านข้อมูลนำเข้าในรูปแบบต่อไปนี้:

- บรรทัดที่ 1: $n \ k \ q$
- บรรทัดที่ 2: $r[0] \ r[1] \ \dots \ r[n - 1]$
- บรรทัดที่ $3 + i$ (สำหรับ $0 \leq i \leq q - 1$): $x \ y$ สำหรับการเรียกใช้ฟังก์ชัน `compare_plants` ครั้งที่ i

เกรดเดอร์ตัวอย่างเขียนข้อมูลส่งออกในรูปแบบต่อไปนี้:

- บรรทัดที่ $1 + i$ (สำหรับ $0 \leq i \leq q - 1$): ค่าที่ถูกคืนจากการเรียกใช้ฟังก์ชัน `compare_plants` ครั้งที่ i