วิ่งวิ่ง (Fatigue)

ณ เมืองแห่งหนึ่งได้มีการจัดงานออกกำลังกายเป็นประจำทุกสัปดาห์เพื่อส่งเสริมสมรรถภาพทางกายและจิตใจ ซึ่งในวันที่ 26 ตุลาคมของทุก ๆ ปีจะมีการจัดกิจกรรมใหญ่ที่ชื่อว่า "วิ่งเพื่อสุขภาพ" ซึ่งจะจัดท่ามกลางชุมชนที่เรียงรายเป็นเส้นตรง

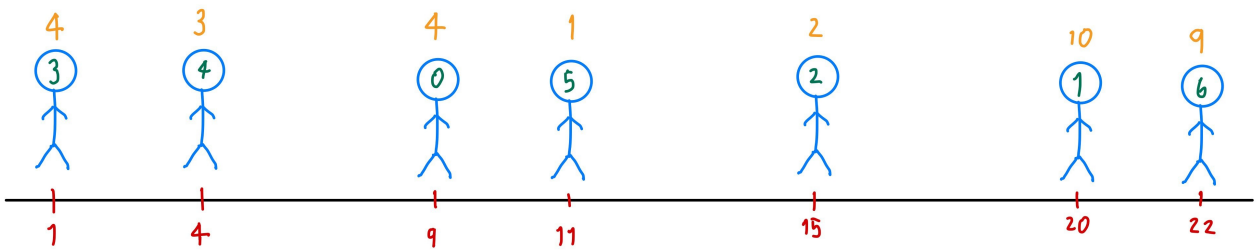
ในการจัดงานปีนี้ได้มีผู้เข้าร่วมกิจกรรมทั้งหมด N คน แต่ละคนจะประจำที่ที่พิกัด a_i ในตอนแรก และจะมีของติดตัวระหว่างวิ่งน้ำหนัก w_i โดยที่**สามารถเปลี่ยนน้ำหนักได้** การวิ่งจะเป็นในลักษณะดังนี้

1. การวิ่งนี้ผู้วิ่งแต่ละคนจะทำการวิ่งเป็นกลุ่มขนาดตั้งแต่ 1 คนเป็นต้นไป(รวมตัวเอง)
2. เมื่อเริ่มวิ่ง กลุ่มวิ่งทุกกลุ่มจะทำการตัดสินใจว่าจะวิ่งไปซ้าย(ค่าพิกัดน้อยกว่าปัจจุบัน) หรือขวา(ค่าพิกัดมากกว่าปัจจุบัน) หรืออยู่กับที่ หากเลือกที่จะวิ่งจะเพิ่มความล้าไป $W \times D$ เมื่อ W คือผลรวมของค่า w_i ในกลุ่มนั้น ๆ และ D คือระยะพิกัดที่เปลี่ยนแปลงไป
3. หากนักวิ่งสองกลุ่มวิ่งมาอยู่ที่พิกัดเดียวกันหรือวิ่งผ่านกัน นักวิ่งทั้งสองกลุ่มจะเชิญชวนกันให้รวมเป็นกลุ่มเดียวกัน ตัวอย่างเช่นนักวิ่งคนที่ 1,2,7 อยู่กลุ่มแรกและนักวิ่งคนที่ 5,11 อยู่กลุ่มที่ 2 แล้ววิ่งมาที่พิกัดเดียวกัน กลุ่มใหม่ที่จะเกิดขึ้นแทนคือมีนักวิ่งคนที่ 1,2,5,7,11 ในกลุ่มเดียวกัน

เพื่อเป็นการเผยแพร่ความยิ่งใหญ่ในงานดังกล่าว จึงได้มีการเก็บรูปภาพกลุ่มนักวิ่ง แต่เนื่องจากเหตุการณ์ในอนาคตนั้นมีความไม่แน่นอนในทุก ๆ เรื่อง ทางทีมจัดงานจึงคาดเดาเหตุการณ์ที่เกิดขึ้นไว้ทั้งหมด M เหตุการณ์ดังนี้

1. คนวิ่งที่ X มีการเปลี่ยนของพกติดตัวเป็นน้ำหนัก V
2. หากจะมีการ**ถ่ายรูปคนเป็นจำนวน** 1 **กลุ่ม** ประกอบไปด้วยนักวิ่ง p คน เป็นนักวิ่งคนที่ $B_0, B_1, ..., B_{p-1}$ **สามารถระบุคนซ้ำกันได้** ถามว่าค่าความล้ารวมที่น้อยที่สุดคือเท่าใด หากสามารถถ่ายรูปได้ทันทีหลังจากทั้ง p คน**รวมกลุ่มเป็นกลุ่มเดียวกัน**
3. หากจะมีการ**ถ่ายรูปคนเป็นจำนวน** 1 **กลุ่ม** ประกอบไปด้วยนักวิ่ง q ช่วง เป็นนักวิ่งคนที่ $[L_0, R_0], [L_1, R_1], \ldots, [L_{q-1}, R_{q-1}]$ โดยที่**ช่วงสามารถซ้อนทับกันได้** ถามว่าค่าความล้ารวมที่น้อยที่สุดคือเท่าใด หากสามารถถ่ายรูปได้ทันทีหลังจากทั้ง q ช่วง**รวมกลุ่มเป็นกลุ่มเดียวกัน**

พิจารณาตัวอย่างประกอบ



ให้สีเขียวแทนนักวิ่งคนที่เท่าใด สีส้มแทนค่า w_i และสีแดงแทนพิกัดเริ่มต้น หากจะถ่ายรูปนักวิ่งคนที่ 0,2 ความล้าที่น้อยที่สุดสามารถทำได้ดังนี้

- นักวิ่งคนที่ 2 วิ่งไปที่พิกัด 11 และรวมกลุ่มกับนักวิ่งคนที่ 5 (ค่าความล้ารวม 8)
- กลุ่มนักวิ่งคนที่ 2,5 วิ่งไปที่พิกัด 9 และรวมกลุ่มกับนักวิ่งคนที่ 0 (ค่าความล้ารวม 14)
- ทำการถ่ายรูปที่พิกัด 9 (กลุ่มมีนักวิ่งคนที่ 0,2 ครบ)

หากจะถ่ายรูปนักวิ่งคนที่ [0,1] และ [3,5] และ ความล้าที่น้อยที่สุดสามารถทำได้ดังนี้

- นักวิ่งคนที่ 1 วิ่งไปที่พิกัด 15 และรวมกลุ่มกับนักวิ่งคนที่ 2 (ค่าความล้ารวม 50)
- นักวิ่งคนที่ 0 วิ่งไปที่พิกัด 11 และรวมกลุ่มกับนักวิ่งคนที่ 5 (ค่าความล้ารวม 58)
- นักวิ่งคนที่ 3 วิ่งไปที่พิกัด 4 และรวมกลุ่มกับนักวิ่งคนที่ 4 (ค่าความล้ารวม 70)
- กลุ่มนักวิ่งคนที่ 3,4 วิ่งไปที่พิกัด 11 และรวมกลุ่มกับกลุ่มนักวิ่งคนที่ 0,5 (ค่าความล้ารวม 119)
- กลุ่มนักวิ่งคนที่ 0,3,4,5 วิ่งไปที่พิกัด 12 (ค่าความล้ารวม 131)
- กลุ่มนักวิ่งคนที่ 1,2 วิ่งไปที่พิกัด 12 และรวมกลุ่มกับกลุ่มนักวิ่งคนที่ 0,3,4,5 (ค่าความล้ารวม 167)
- ทำการถ่ายรูปที่พิกัด 12 (กลุ่มมีนักวิ่งคนที่ 0,1,3,4,5 ครบ)

จากทั้งสองตัวอย่างพบว่าจะไม่สามารถวิ่งผ่านนักวิ่งโดยไม่ชวนมาร่วมกลุ่มได้ สำหรับเหตุการณ์ประเภทที่ 2 และ 3 ทางทีมผู้จัดงานขอให้คุณออกแบบโปรแกรมที่มีประสิทธิภาพเพียงพอในการตอบคำถามดังกล่าว

รายละเอียดการเขียนโปรแกรม

```
void initialize(int N, vector<int> A, vector<int> W)
```

- ฟังก์ชันนี้จะถูกเรียกเพียงครั้งเดียว
- vector A และ W จะมีขนาด N แทนพิกัด a_i และค่า w_i ของแต่ละคน
- ฟังก์ชันนี้ไม่ต้องคืนค่า

```
void change_weight(int X, int V)
```

- ฟังก์ชันนี้จะถูกเรียกตามจำนวนเหตุการณ์ประเภทที่ 1
- จำนวนเต็ม X และ V คือคนวิ่งที่ X มีการเปลี่ยนน้ำหนักของเป็น V
- ฟังก์ชันนี้ไม่ต้องคืนค่า

```
long long point_fatigue(vector<int> B)
```

- ฟังก์ชันนี้จะถูกเรียกตามจำนวนเหตุการณ์ประเภทที่ 2
- vector B จะมีขนาด p โดยที่ B[i] แทน B_i
- ฟังก์ชันนี้จะต้องทำการคืนค่าเป็นจำนวนเต็ม 1 จำนวนแทนค่าความล้ารวมที่น้อยที่สุดที่จะทำให้นักวิ่งคนที่ $B_0, B_1, ..., B_{p-1}$ รวมอยู่ในกลุ่มเดียวกันได้

```
long long interval_fatigue(vector<vector<int>> In)
```

- ฟังก์ชันนี้จะถูกเรียกตามจำนวนเหตุการณ์ประเภทที่ 3
- vector In จะมีขนาด q โดยที่ In[i] มีขนาด 2 แทน L_i, R_i
- ฟังก์ชันนี้จะต้องทำการคืนค่าเป็นจำนวนเต็ม 1 จำนวนแทนค่าความล้ารวมที่น้อยที่สุดที่จะทำให้นักวิ่งคนที่

$[L_0, R_0], [L_1, R_1], \ldots, [L_{q-1}, R_{q-1}]$ รวมอยู่ในกลุ่มเดียวกันได้

ขอบเขต

- $3 \leq N \leq 100\ 000$
- $0 \leq X \leq N - 1$
- $1 \leq a_i, w_i, V \leq 10^6$
- $0 \leq L_i \leq R_i \leq N - 1$
- $\sum p + \sum q \leq 300\ 000$

ปัญหาย่อย

1. (9 คะแนน) จะมีเหตุการณ์ประเภทที่ 1 และ 2 เท่านั้น เหตุการณ์ประเภทที่ 2 จะเรียกนักวิ่งเพียงสองคนที่ไม่มีนักวิ่งอยู่ระหว่างทางวิ่งไปหากัน และนักวิ่งทุกคนประจำตำแหน่งไม่ซ้ำกัน
2. (10 คะแนน) จะมีเหตุการณ์ประเภทที่ 1 และ 2 เท่านั้น และ $N \leq 300, M \leq 600, a_i < N$
3. (11 คะแนน) จะมีเหตุการณ์ประเภทที่ 1 และ 2 เท่านั้น และ $N \leq 3\ 000, M \leq 6\ 000$
4. (15 คะแนน) จะมีเหตุการณ์ประเภทที่ 2 เท่านั้น และ $w_i = 1$
5. (6 คะแนน) จะมีเหตุการณ์ประเภทที่ 2 เท่านั้น
6. (9 คะแนน) จะมีเหตุการณ์ประเภทที่ 2 และ 3 เท่านั้น และ $w_i = 1$
7. (9 คะแนน) จะมีเหตุการณ์ประเภทที่ 2 และ 3 เท่านั้น
8. (14 คะแนน) จะมีเหตุการณ์ประเภทที่ 1 และ 2 เท่านั้น
9. (17 คะแนน) ไม่มีเงื่อนไขเพิ่มเติม

ตัวอย่าง

พิจารณาตัวอย่าง เกรดเดอร์จะเรียกฟังก์ชันดังนี้

```
initialize(7, [9, 20, 15, 1, 4, 11, 22], [4, 10, 2, 4, 3, 1, 9])
```

จากนั้นเรียก

```
point_fatigue([0, 2])
```

ซึ่งจะต้องคืนค่า

```
14
```

จากนั้นเรียก

```
interval_fatigue([[0, 1], [3, 5]])
```

ซึ่งจะต้องคืนค่า

```
167
```

จากนั้นเรียก

```
interval_fatigue([[0, 5], [3, 4]])
```

ซึ่งจะต้องคืนค่า

```
167
```

จากนั้นเรียก

```
change_weight(0, 2)
```

จากนั้นเรียก

```
point_fatigue([0, 2])
```

ซึ่งจะต้องคืนค่า

```
12
```

เกรดเดอร์ตัวอย่าง

- บรรทัดที่ 1: $N\ M$
- บรรทัดต่อมา : $A[0]\ A[1]\ \ldots\ A[N-1]$
- บรรทัดต่อมา : $W[0]\ W[1]\ \ldots\ W[N-1]$
- M บรรทัดต่อมา e
  - หาก $e = 1$: $X\ V$
  - หาก $e = 2$: $p\ B[0]\ B[1]\ \ldots\ B[p-1]$
  - หาก $e = 3$: $q\ In[0][0]\ In[0][1]\ In[1][0]\ In[1][1]\ \ldots\ In[q-1][0]\ In[q-1][1]$

ข้อมูลส่งออกสำหรับเกรดเดอร์ตัวอย่างจะพิมพ์ค่าที่ได้จากการเรียกฟังก์ชัน `point_fatigue` และฟังก์ชัน `interval_fatigue`

ข้อจำกัด

- Time limit: 2 seconds
- Memory limit: 256 MB