



กับดักความยากระดับ sss++

1 second, 256 megabytes

By Detective conan, Mingyuanz

หลังจากที่ njoop วางแผน DevJ เพื่อตัดกำลังร้านเบเกอรี่คู่แข่งได้สำเร็จแล้ว เขายกได้พัฒนาร้านขาย taiyaki ของเขาต่ออย่างราบรื่น แต่เค้าได้ทราบข่าวมาจากการเพื่อนของเค้าที่หลบหน่อยยุ่งไปประเทศอุกานด้า ว่ามีนักเล่นแร่แปรธาตุช่วยสายลับ DevJ ในภารกิจที่ njoop วางแผนตอนที่แล้ว njoop จึงต้องเร่งวางแผนกับดักชนิดใหม่ทันที โดย njoop จึงได้สั่งกับดักชนิดใหม่จากเพื่อนของเขาว่าอยู่ที่ประเทศอุกานด้า โดยใช้เวลาขนส่ง 109 วินาที แต่นั่นก็ไม่ใช่ปัญหา เพราะ DevJ ต้องใช้เวลาขนสิ่งของ 1010 วินาที เพื่อจะมาถึงสถานที่แห่งนี้ โดยเนื่องจากกับดักเสร็จก็หนีไป และไม่ลืมที่จะทิ้งจดหมายเยี่ยหยันถึงสายลับ DevJ ต่อมาเมื่อ DevJ และ a0ms1n เดินทางมาถึง เค้าได้เห็นจดหมายเยี่ยหยันที่ njoop เขียนไว้เจิงโกรธจัด และเสาะหาวิธีที่จะใช้ในการแก้กับดักชนิดนี้ และด้วยความที่ DevJ เป็นสายลับอัจฉริยะ เขายังใช้เวลาเพียง 5 นาที ในการหาวิธีการแก้กับดักเหล่านี้ เขายังเกตได้ว่า กับดักมีทั้งหมด n ชิ้น กับดักแต่ละชิ้นจะมีหมายเลขเป็นของตัวเอง ตั้งแต่หมายเลข 1 ถึง n และกับดักหมายเลข i ($1 \leq i \leq n$) มีค่าความยากเท่ากับ x_i และการแก้กับดักทั้งหมดต้องแก้เรียงจากกับดักที่หมายเลขน้อยไปหามาก การแก้กับดักชิ้นที่ i จากเงื่อนไขนี้ จึงมีทั้งหมด 2^{n-1} กรณีดังนี้

- การแก้กับดักหมายเลข i ต้องใช้ข้อมูลจากกับดักหมายเลขที่ j โดยที่ j คือหมายเลขของกับดักชื่นได้ที่ยังไม่ถูกแก้ และมีค่าความยากน้อยกว่ากับดักหมายเลข i ทำให้ DevJ ต้องเดินจากกับดักหมายเลข i ไปที่กับดักหมายเลข j เพื่อไปจดข้อมูลลับมาใช้แก้กับดักหมายเลข i ซึ่งเขาจะใช้เวลาเดินทางไป-กลับรวมทั้งหมด $|i - j|$ วินาที แต่ในกรณีที่ไม่มีกับดักหมายเลข j ที่ตรงกับเงื่อนไขดังกล่าวเลย จะถือว่าสามารถข้ามขั้นตอนนี้ได้โดย ก็คือเวลาที่ใช้ในขั้นตอนนี้เท่ากับ 0 วินาที
 - หลังจากได้ข้อมูลมาแล้ว ก็ถึงเวลาลงมือแก้กับดัก แต่เนื่องจาก DevJ นั้นอัจฉริยะเกินไปหน่อย เขายังหาวิธีแก้กับดักได้หลากหลายวิธีอีกด้วย รวมทั้งหมด m วิธี โดยวิธีที่ k จะใช้เวลาแก้ t_k วินาที แต่จะใช้แก้ได้เพียงแค่กับดักที่มีค่าความยากอย่างน้อย d_k เท่านั้น เขายังต้องเลือกมาเพียงวิธีเดียวที่ตรงกับเงื่อนไขมาใช้ในการแก้กับดัก (วิธีแก้หนึ่งวิธีสามารถใช้แก้กับดักหลายชิ้นได้)

กำหนดให้ DevJ ไม่ใช้เวลาใดๆระหว่างขั้นตอนที่ 1 กับ 2 และเมื่อ DevJ ทำขั้นตอนที่ 2 ของกับดักหมายเลข i เสร็จแล้ว จะถือว่าแก้กับดักหมายเลข i ได้สำเร็จ และจะสามารถเริ่มทำขั้นตอนที่ 1 ของกับดักขั้นต่อไป (หมายเลข $i+1$) ต่อได้ทันทีโดยไม่ใช้เวลาใดๆระหว่างขั้นตอนเช่นกัน และเนื่องจากการแก้กับดักเป็นงานที่อันตรายมาก DevJ ซึ่งเป็นผู้เชี่ยวชาญนั้น จะเป็นคนที่แก้กับดักเพียงคนเดียว

แต่ถึงจะรู้แล้วว่าต้องแก้ยังไง เขาก็ไม่สามารถคิดได้เองว่า ในการแก้กับดักแต่ละชิ้นในแต่ละขั้นตอน ต้องเดินไปจดข้อมูลที่กับดักชิ้นไหน และจะใช้วิธีไหนแก้กับดักแต่ละชิ้นดี จึงจะทำให้แก้ทุกชิ้นได้สำเร็จเร็วที่สุด เพราะว่า DevJ ไม่อยากเสียเวลาแก้กับดักพอกวนนี้มากเกินจำเป็น เขายังต้องการให้คุณ ซึ่งเป็น programmer ที่เก่งที่สุดในจักรวาล(ของคุณ) ช่วยเขียนโปรแกรมหาระยะเวลาที่น้อยที่สุดที่ DevJ จะใช้ในการแก้กับดักแต่ละชิ้นในแต่ละขั้นตอนได้ และหาเวลาร่วมทั้งหมดที่ DevJ ต้องใช้ในการแก้กับดักทุกชิ้นได้สำเร็จ โดยคุณต้องได้คำตอบภายในเวลา 1 วินาที เพราะ DevJ ต้องรีบไปตามล่า njoop ต่อ DevJ ต้องพึ่งคุณแล้วล่ะ!



โจทย์ จงเขียนโปรแกรมเพื่อหาระยะเวลาที่น้อยที่สุดที่ DevJ จะใช้ในการแก้กับดักแต่ละชิ้น ในแต่ละขั้นตอน และคำนวณเวลารวมทั้งหมดที่เขาต้องใช้ในการแก้กับดักทุกชิ้นได้สำเร็จ

ข้อมูลนำเข้า

บรรทัดแรก จำนวนเต็ม n แทน จำนวนกับดักที่ gjoop ได้ติดตั้งไว้ ($1 \leq n \leq 5 \cdot 10^5$)

บรรทัดที่ 2 ประกอบด้วยจำนวนเต็ม n จำนวน คืนด้วยซึ่งว่างหนึ่งช่อง

แต่ละจำนวนคือ x_i แทนค่าความยากของกับดักหมายเลข i ($1 \leq x_i \leq 10^6$) และ ($1 \leq i \leq n$)

บรรทัดที่ 3 จำนวนเต็ม m แทนจำนวนวิธีแก้กับดักที่ DevJ คิดมาได้ ($1 \leq m \leq 5 \cdot 10^5$)

บรรทัดที่ 4 ถึง $m + 3$ บรรทัดที่ $i + 3$ ($1 \leq i \leq m$)

ประกอบไปด้วยจำนวนเต็ม 2 จำนวน t_i และ d_i คืนด้วยซึ่งว่างหนึ่งช่อง

t_i แทน เวลาที่วิธีที่ i ใช้ในการแก้กับดัก มีหน่วยเป็นวินาที ($1 \leq t_i \leq 10^6$) และ

d_i แทน ค่าความยากของกับดักที่ต่ำที่สุดที่จะใช้วิธีที่ i ในการแก้กับดักได้ โดยการันตีว่า $d_1 = 1$ เสมอ ($1 \leq d_i \leq 10^6$)

ข้อมูลส่งออก

บรรทัดที่หนึ่ง ประกอบด้วยจำนวนเต็ม n จำนวน คืนด้วยซึ่งว่างหนึ่งช่อง โดยตัวที่ i ($1 \leq i \leq n$) คือ

ระยะเวลาที่น้อยที่สุดที่ DevJ จะใช้ในการเดินไป-กลับ เพื่อจดข้อมูลกลับมาใช้แก้กับดักหมายเลข i มีหน่วยเป็นวินาที

บรรทัดที่ 2 ประกอบด้วยจำนวนเต็ม n จำนวน คืนด้วยซึ่งว่างหนึ่งช่อง โดยตัวที่ i ($1 \leq i \leq n$) คือ

ระยะเวลาที่น้อยที่สุดที่ DevJ จะใช้ในการลงมือแก้กับดักหมายเลข i มีหน่วยเป็นวินาที

บรรทัดที่ 3 ประกอบด้วยจำนวนเต็ม 1 จำนวน คือ

ระยะเวลารวมทั้งหมดที่น้อยที่สุดที่ DevJ ต้องใช้ในการแก้กับดักทุกชิ้นได้สำเร็จ มีหน่วยเป็นวินาที



ตัวอย่างข้อมูลนำเข้าและข้อมูลส่งออก

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
7	2 1 3 1 1 0 0
8 10 2 12 6 1 7	7 7 11 7 9 15 9
7	73
15 1	
12 3	
10 5	
9 12	
7 8	
11 2	
9 3	

คำอธิบายตัวอย่าง

คำอธิบายข้อมูลส่งออก บรรทัดที่ 1

2 มาจากการที่ DevJ เดินจากกับดักที่ 1 ไปยังกับดักที่ 3 จึงใช้เวลา $|1 - 3| = 2$ วินาที

1 มาจากการที่ DevJ เดินจากกับดักที่ 2 ไปยังกับดักที่ 3 จึงใช้เวลา $|2 - 3| = 1$ วินาที

3 มาจากการที่ DevJ เดินจากกับดักที่ 3 ไปยังกับดักที่ 6 จึงใช้เวลา $|3 - 6| = 3$ วินาที

1 มาจากการที่ DevJ เดินจากกับดักที่ 4 ไปยังกับดักที่ 5 จึงใช้เวลา $|4 - 5| = 1$ วินาที

1 มาจากการที่ DevJ เดินจากกับดักที่ 5 ไปยังกับดักที่ 6 จึงใช้เวลา $|5 - 6| = 1$ วินาที

0 เนื่องจากไม่มีกับดักที่ยังไม่แกะและมีค่าความยากน้อยกว่ากับดักที่ 6 จึงไม่ต้องเดินไปเอาข้อมูล และใช้เวลา 0 วินาที

0 เนื่องจากไม่มีกับดักที่ยังไม่แกะและมีค่าความยากน้อยกว่ากับดักที่ 7 จึงไม่ต้องเดินไปเอาข้อมูล และใช้เวลา 0 วินาที

คำอธิบายข้อมูลส่งออก บรรทัดที่ 2

7 มาจากการแก้กับดักที่ 1 DevJ เลือกใช้วิธีแก้วรีที่ 5 ซึ่งใช้เวลา 7 วินาที ที่สามารถแก้ได้เมื่อความยากไม่ต่ำกว่า 8

7 มาจากการแก้กับดักที่ 2 DevJ เลือกใช้วิธีแก้วรีที่ 5 ซึ่งใช้เวลา 7 วินาที ที่สามารถแก้ได้เมื่อความยากไม่ต่ำกว่า 8

11 มาจากการแก้กับดักที่ 3 DevJ เลือกใช้วิธีแก้วรีที่ 6 ซึ่งใช้เวลา 11 วินาที ที่สามารถแก้ได้เมื่อความยากไม่ต่ำกว่า 2

7 มาจากการแก้กับดักที่ 4 DevJ เลือกใช้วิธีแก้วรีที่ 5 ซึ่งใช้เวลา 7 วินาที ที่สามารถแก้ได้เมื่อความยากไม่ต่ำกว่า 8

9 มาจากการแก้กับดักที่ 5 DevJ เลือกใช้วิธีแก้วรีที่ 7 ซึ่งใช้เวลา 9 วินาที ที่สามารถแก้ได้เมื่อความยากไม่ต่ำกว่า 3

15 มาจากการแก้กับดักที่ 6 DevJ เลือกใช้วิธีแก้วรีที่ 1 ซึ่งใช้เวลา 15 วินาที ที่สามารถแก้ได้เมื่อความยากไม่ต่ำกว่า 1

9 มาจากการแก้กับดักที่ 7 DevJ เลือกใช้วิธีแก้วรีที่ 7 ซึ่งใช้เวลา 9 วินาที ที่สามารถแก้ได้เมื่อความยากไม่ต่ำกว่า 3

คำอธิบายข้อมูลส่งออก บรรทัดที่ 3

มาจากการเอาระยะเวลาทั้งหมดที่ DevJ ใช้ในการเดินทางไป-กลับเพื่อเก็บข้อมูลทั้งหมด รวมกับเวลาที่ใช้ในการแก้กับดักทั้งหมด จะได้ว่า $2 + 1 + 3 + 1 + 1 + 0 + 0 + 7 + 7 + 11 + 7 + 9 + 15 + 9 = 73$ วินาที



การให้คะแนน

คะแนนเต็ม 400 คะแนน มี 6 กลุ่มชุดทดสอบ

10 คะแนน: $n \leq 5 \cdot 10^5$; $m = 1$; $x_i > x_{i+1}$ ($1 \leq i < n$)

75 คะแนน: $n \leq 10^4$; $m = 1$

100 คะแนน: $n \leq 5 \cdot 10^5$; $m = 1$

100 คะแนน: $n, m \leq 10^4$; $x_i > x_{i+1}$ ($1 \leq i < n$)

75 คะแนน: $n, m \leq 5 \cdot 10^5$; $x_i > x_{i+1}$ ($1 \leq i < n$)

40 คะแนน: ไม่มีเงื่อนไขเพิ่มเติม

**จะได้คะแนนในแต่ละกลุ่มชุดทดสอบ ก็ต่อเมื่อโปรแกรมให้ผลลัพธ์ถูกต้องในชุดทดสอบอย่างทั้งหมด

คำแนะนำ

หากใช้ภาษา C++ และนำให้เพิ่มคำสั่ง `cin.tie(nullptr)->sync_with_stdio(false);`
และให้ใช้ '\n' แทน endl เช่น `cout << "Hello World" << '\n';`

หากใช้ภาษา C/C++ และนำให้ใช้คอมไพล์เตอร์ **GNU G++17 7.3.0** ในการ Submit Code