



นักโทษเสียงทนาย

ในเรือนจำแห่งหนึ่งมีกลุ่มนักโทษ 500 คน วันหนึ่งผู้คุมเสนอโอกาสเพื่อให้นักโทษทุกคนได้รับการปล่อยตัว ผู้คุมคนนั้นวางถุงที่มีเงินใส่ไว้สองถุงได้แก่ ถุง A และถุง B ไว้ในห้องแห่งหนึ่ง โดยในแต่ละถุงจะมีเหรียญใส่ไว้ตั้งแต่ 1 ถึง N เหรียญ จำนวนเหรียญในถุง A จะไม่เท่ากับจำนวนเหรียญในถุง B จากนั้นผู้คุมก็เสนอกระบวนการประลองให้กับกลุ่มนักโทษ โดยเป้าหมายของกลุ่มนักโทษคือการระบุถุงที่มีจำนวนเหรียญน้อยกว่า

ภายในห้องนั้นนอกจากมีถุงเงินแล้ว ยังมีจำนวนเต็มค่าหนึ่งเขียนอยู่บนไวท์บอร์ดตลอดเวลา โดยในเริ่มแรกจำนวนที่อยู่บนไวท์บอร์ดคือ 0

จากนั้น ผู้คุมจะเรียกนักโทษเข้ามาในห้องทีละคน นักโทษที่เข้ามาในห้องจะไม่ทราบว่ามีนักโทษคนใดหรือมีนักโทษกี่คนที่ได้เข้ามาในห้องก่อนหน้าเขา ทุกครั้งที่นักโทษเดินเข้ามาในห้อง นักโทษจะอ่านจำนวนที่อยู่บนไวท์บอร์ด หลังจากอ่านจำนวนดังกล่าวแล้ว นักโทษจะต้องเลือกถุง A หรือถุง B จากนั้นนักโทษจะ **ตรวจสอบ** ถุงที่เขาเลือก ทำให้ทราบจำนวนเหรียญที่อยู่ภายในถุงนั้น หลังจากนั้นนักโทษจะต้องเลือกการกระทำอันใดอันหนึ่งจากสองทางเลือกนี้

- ลบจำนวนเดิมและเขียนจำนวนเต็มที่ไม่เป็นลบลงบนไวท์บอร์ด และออกไปจากห้อง ทั้งนี้เขาสามารถที่จะเปลี่ยนหรือใช้จำนวนเดิมก็ได้ จากนั้นกระบวนการประลองจะดำเนินต่อไป (เว้นแต่นักโทษทั้ง 500 คนได้เข้ามาในห้องแล้ว)
- ระบุว่าถุงใดถุงหนึ่งเป็นถุงที่มีจำนวนเหรียญน้อยกว่า และจบการประลองทันที

โดยผู้คุมจะไม่เรียกนักโทษที่ออกจากห้องไปแล้วเข้ามาอีก

กลุ่มนักโทษจะชนะการประลองถ้ามีนักโทษคนใดคนหนึ่งระบุถุงที่มีจำนวนเหรียญน้อยกว่าได้ถูกต้อง กลุ่มนักโทษจะแพ้การประลองถ้ามีนักโทษคนหนึ่งเลือกถุงไม่ถูกต้อง หรือนักโทษทั้ง 500 คนได้เข้ามาในห้องแล้ว แต่ไม่มีนักโทษคนใดที่สามารถระบุถุงที่มีจำนวนเหรียญน้อยกว่า

ก่อนการประลองจะเริ่มขึ้น นักโทษทุกคนจะรวมตัวกันในห้องโถงเรือนจำและออกแบบ **กลยุทธ์** การประลองร่วมกันผ่านสามขั้นตอนนี้

- กลุ่มนักโทษเลือกจำนวนเต็มที่ไม่เป็นลบ x ซึ่งเป็นจำนวนที่มีค่ามากที่สุดที่นักโทษสามารถเขียนบนไวท์บอร์ด
- กลุ่มนักโทษตัดสินใจว่า สำหรับจำนวน i ที่ถูกเขียนไว้บนไวท์บอร์ด ($0 \leq i \leq x$) นักโทษควรจะเลือกตรวจสอบถุงใดเมื่อเขาเห็นจำนวน i ปรากฏอยู่บนไวท์บอร์ดขณะที่เขาเข้ามาในห้อง
- กลุ่มนักโทษตัดสินใจเลือกการกระทำที่นักโทษในห้องต้องทำหลังจากที่เขาทราบจำนวนเหรียญในถุงที่ถูกเลือก กล่าวคือสำหรับจำนวน i ที่ถูกเขียนไว้บนไวท์บอร์ด ($0 \leq i \leq x$) และสำหรับจำนวนเหรียญ j ที่เขาเห็นจากการตรวจสอบถุง ($1 \leq j \leq N$) กลุ่มนักโทษต้องตัดสินใจว่า
 - จำนวนใดตั้งแต่ 0 ถึง x ที่ควรจะถูกเขียนบนไวท์บอร์ด หรือ
 - พิจารณาว่าถุงใดควรเป็นถุงที่จะถูกระบุว่าเป็นถุงที่มีเหรียญน้อยกว่า

ในกรณีที่ชนะการประลองแล้ว ผู้คุมจะขังนักโทษต่อไปอีก x วันจึงจะปล่อย

ปัญหาของคุณคือการคัดเลือกกลยุทธ์ที่จะทำให้กลุ่มนักโทษชนะการประลองนี้เสมอ (ไม่ว่าในถุง A และถุง B จะมีจำนวนเหรียญอยู่เท่าใด) คะแนนของคุณจะขึ้นอยู่กับค่าของ x (ดูรายละเอียดในปัญหาย่อย)

รายละเอียดการเขียนโปรแกรม

คุณต้องเขียนฟังก์ชันต่อไปนี้:

```
int[][] devise_strategy(int N)
```

- N : จำนวนเหรียญที่มากที่สุดที่เป็นไปได้ในแต่ละถุง
- ฟังก์ชันนี้ต้องคืนค่าเป็นอาร์เรย์ s ของอาร์เรย์ของจำนวนเต็ม $N + 1$ ตัว เพื่ออธิบายกลยุทธ์ของคุณ ค่าของ x คือขนาดของอาร์เรย์ s ลบหนึ่ง. สำหรับค่า i โดยที่ $0 \leq i \leq x$, อาร์เรย์ $s[i]$ จะกำหนดว่านักโทษต้องเลือกการกระทำใดถ้าเขาเห็นจำนวน i บนไวท์บอร์ดเมื่อเข้ามาในห้อง:
 1. ค่าของ $s[i][0]$ คือ 0 ถ้านักโทษต้องเลือกตรวจสอบถุง A, หรือ 1 ถ้านักโทษต้องเลือกตรวจสอบถุง B
 2. ให้ j เป็นจำนวนเหรียญที่อยู่ในถุงที่ถูกตรวจสอบ นักโทษจะเลือกกระทำการดังต่อไปนี้:
 - ถ้าค่าของ $s[i][j]$ เป็น -1 , นักโทษจะระบุให้ถุง A เป็นถุงที่มีเหรียญน้อยกว่า
 - ถ้าค่าของ $s[i][j]$ เป็น -2 , นักโทษจะระบุให้ถุง B เป็นถุงที่มีเหรียญน้อยกว่า
 - ถ้าค่าของ $s[i][j]$ คือจำนวนที่ไม่เป็นลบ นักโทษจะต้องเขียนจำนวนนั้นลงบนไวท์บอร์ด ทั้งนี้ $s[i][j]$ ต้องมีค่าไม่เกิน x
- ฟังก์ชันนี้จะถูกเรียกเพียงหนึ่งครั้ง

ตัวอย่าง

พิจารณาการเรียกฟังก์ชันต่อไปนี้:

```
devise_strategy(3)
```

ให้ x แทนจำนวนที่นักโทษอ่านได้จากไวท์บอร์ดเมื่อเขาเดินเข้ามาในห้อง กลยุทธ์หนึ่งที่ถูกต้องการคือกลยุทธ์ต่อไปนี้:

- ถ้า $x = 0$ (รวมถึงจำนวนแรกด้วย), เลือกให้ตรวจสอบถุง A
 - ถ้ามี 1 เหรียญ, ระบุให้ถุง A เป็นถุงที่มีเหรียญน้อยกว่า
 - ถ้ามี 3 เหรียญ, ระบุให้ถุง B เป็นถุงที่มีเหรียญน้อยกว่า
 - ถ้ามี 2 เหรียญ, เขียน 1 บนไวท์บอร์ด (ทดแทน 0 เดิม)
- ถ้า $x = 1$, เลือกให้ตรวจสอบถุง B
 - ถ้ามี 1 เหรียญ, ระบุให้ถุง B เป็นถุงที่มีเหรียญน้อยกว่า
 - ถ้ามี 3 เหรียญ, ระบุให้ถุง A เป็นถุงที่มีเหรียญน้อยกว่า
 - ถ้ามี 2 เหรียญ, เขียน 0 บนไวท์บอร์ด (ทดแทน 1 เดิม) อย่างไรก็ตาม กรณีนี้ไม่สามารถเกิดขึ้นได้ เนื่องจากเราสามารถสรุปได้ว่าทั้งสองถุงมีถุงละ 2 เหรียญ ซึ่งเป็นไปไม่ได้

เพื่อเลือกใช้กลยุทธ์นี้ ฟังก์ชันต้องคืนค่า $[[0, -1, 1, -2], [1, -2, 0, -1]]$

เนื่องจากขนาดของอาร์เรย์ที่คืนมาคือ 2 ดังนั้น ค่าของ x คือ $2 - 1 = 1$

ข้อจำกัด

- $2 \leq N \leq 5000$

ปัญหาย่อย

- (5 คะแนน) $N \leq 500$, ค่าของ x ต้องไม่มากกว่า 500
- (5 คะแนน) $N \leq 500$, ค่าของ x ต้องไม่มากกว่า 70
- (90 คะแนน) ค่าของ x ต้องไม่มากกว่า 60

ทั้งนี้ ถ้ามีรายการทดสอบรายการใดที่ทำให้อาร์เรย์ที่ถูกคืนค่ามาจาก `devise_strategy` ไม่ใช่กลยุทธ์ที่ถูกต้อง คะแนนของคุณในปัญหาย่อยนั้นจะเป็น 0

ในปัญหาย่อยที่ 3 คุณสามารถได้คะแนนบางส่วน โดยให้ m เป็นค่าสูงสุดของ x จากอาร์เรย์ที่ถูกคืนค่ามาจากรายการทดสอบทั้งหมด ในปัญหาย่อยนั้น คะแนนของคุณในปัญหาย่อยนี้จะถูกคำนวณจากตารางต่อไปนี้:

เงื่อนไข	คะแนน
$40 \leq m \leq 60$	20
$26 \leq m \leq 39$	$25 + 1.5 \times (40 - m)$
$m = 25$	50
$m = 24$	55
$m = 23$	62
$m = 22$	70
$m = 21$	80
$m \leq 20$	90

เกรตเตอร์ตัวอย่าง

เกรตเตอร์ตัวอย่างจะอ่านค่าอินพุตในรูปแบบดังนี้:

- บรรทัดที่ 1: N
- บรรทัดที่ $2 + k$ ($0 \leq k$): $A[k]$ $B[k]$
- บรรทัดสุดท้าย: -1

โดยในแต่ละบรรทัดยกเว้นบรรทัดแรกและบรรทัดสุดท้าย จะแทนสถานการณ์แต่ละสถานการณ์ เราเรียกสถานการณ์ที่ระบุในบรรทัดที่ $2 + k$ ว่าสถานการณ์ที่ k ในสถานการณ์ที่ k ลูก A มี $A[k]$ เหรียญ และลูก B มี $B[k]$ เหรียญ

เกรตเตอร์ตัวอย่างจะเริ่มโดยการเรียก `devise_strategy(N)` เนื่องจากค่าของ x คือขนาดของอาร์เรย์ที่คืนค่ามาลบบ้าง ดังนั้น ถ้าเกรตเตอร์ตัวอย่างตรวจพบว่าอาร์เรย์ที่คืนค่ามาจาก `devise_strategy` ไม่สอดคล้องกับข้อกำหนดที่ระบุไว้ในรายละเอียดการเขียนโปรแกรม เกรตเตอร์จะพิมพ์ข้อความผิดพลาดต่อไปนี้และออกจากการทำงาน:

- s is an empty array: s เป็นอาร์เรย์ว่าง (ซึ่งไม่ใช่กลยุทธ์ที่ถูกต้อง)
- $s[i]$ contains incorrect length: มีดัชนีตำแหน่ง i ($0 \leq i \leq x$) ที่ความยาวของ $s[i]$ ไม่ใช่ $N + 1$
- First element of $s[i]$ is non-binary: มีดัชนีตำแหน่ง i ($0 \leq i \leq x$) ที่ $s[i][0]$ มีค่าไม่เป็น 0 หรือ 1

- $s[i][j]$ contains incorrect value: มีดัชนีตำแหน่ง i, j ($0 \leq i \leq x, 1 \leq j \leq N$) ที่ $s[i][j]$ ไม่อยู่ระหว่าง -2 และ x

ถ้าไม่พบปัญหา เกรดเดอร์ตัวอย่างจะสร้างเอาท์พุตสองชุด

ชุดแรก เกรดเดอร์ตัวอย่างจะพิมพ์เอาท์พุตของกลยุทธ์ของคุณในรูปแบบต่อไปนี้:

- บรรทัดที่ $1 + k$ ($0 \leq k$): ผลของกลยุทธ์ของคุณในสถานการณ์ที่ k ถ้าใช้กลยุทธ์แล้วทำให้นักโทษระบุ A ให้เป็นถุงที่มีเหรียญน้อยกว่า เอาท์พุตจะเป็นตัวอักษร A ถ้าใช้กลยุทธ์แล้วทำให้นักโทษระบุ B ให้เป็นถุงที่มีเหรียญน้อยกว่า เอาท์พุตจะเป็นตัวอักษร B ถ้าใช้กลยุทธ์แล้วไม่มีนักโทษคนใดระบุให้เป็นถุงที่มีเหรียญน้อยกว่า เอาท์พุตจะเป็นตัวอักษร X

ชุดที่สอง เกรดเดอร์จะเขียนไฟล์ `log.txt` ในไดเรกทอรีปัจจุบันในรูปแบบต่อไปนี้:

- บรรทัดที่ $1 + k$ ($0 \leq k$): $w[k][0] w[k][1] \dots$

ลำดับในบรรทัดที่ $1 + k$ จะสอดคล้องกับสถานการณ์ที่ k และแสดงจำนวนที่ถูกเขียนบนไวท์บอร์ด โดย $w[k][l]$ คือจำนวนที่ถูกเขียนโดยนักโทษคนที่ $l + 1$ ที่เข้ามาในห้อง