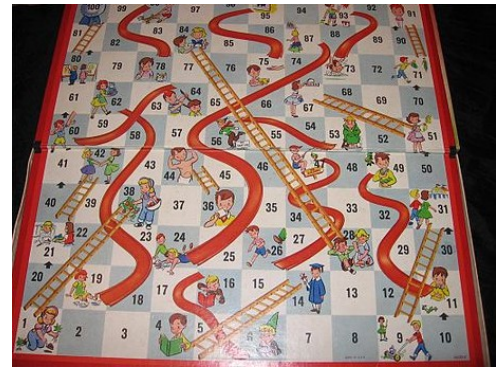


เกมบันไดงู

1 second, 32 MB

เกมบันไดงูเป็นเกมคลาสสิก ในเกมนี้ผู้เล่นจะทอยลูกเต๋า และเดินไปตามตาราง ถ้าตกในช่องที่เป็นบันได ก็จะไต่ขึ้นไปในช่องถัดไป ถ้าตกในช่องที่เป็นงู ก็จะร่วงย้อนกลับไปช่องก่อนหน้า

โจทย์ข้อนี้คุณจะต้องเขียนโปรแกรมเพื่อจำลองการเล่นเกมบันไดงู โดยมีรายละเอียดดังนี้



รูปจาก: <http://en.wikipedia.org/wiki/File:Cnl03.jpg>

- ตารางจะมี N ช่อง เรียกเป็นช่องที่ 1 ถึง N เมื่อเริ่มต้นคุณจะถูกวางพิจารณาว่าเป็นช่อง 0 ก็ได้
- เมื่อทอยลูกเต๋าก็ได้แต้มเท่าใด คุณจะเดินไปด้านหน้าเป็นจำนวนเท่ากับแต้มนั้น
- ทุกช่องในตารางจะมีตัวเลขระบุไว้ ถ้าระบุเป็น 0 จะเป็นช่องธรรมดา นั่นคือ ถ้าตกที่ช่องนี้ก็จะไม่มีการเพิ่มหรือลดตำแหน่ง แต่ถ้าเป็นเลขอื่น ๆ เช่น 5 ก็จะได้เดินเพิ่ม 5 ช่อง หรือถ้าเป็น -20 ก็จะถอยกลับไป 20 ช่อง ช่องที่ N จะมีหมายเลขเป็นเลข 0 เสมอ
- ในการเดินหน้าหรือถอยกลับด้วยหมายเลขในช่องนี้ จะไม่มีการดำเนินการซ้ำแม้ว่าจะเดินเพิ่มหรือถอยไปยังช่องที่มีหมายเลขอื่นนอกจาก 0 อยู่ (ดูตัวอย่างข้อมูลทดสอบ)
- ถ้าผู้เล่นถอยหลังจนเกินช่อง 1 ให้ถือว่ากลับไปเริ่มต้นใหม่ (ไม่ต้องคิดติลล) ถ้าผู้เล่นเดินจนเกินช่องที่ N จะไม่มีการกระโดดกลับไปมาเหมือนการเล่นปกติ ให้ถือว่าผู้เล่นชนะ และหยุดที่ช่องที่ N เท่านั้น

คุณจะได้รับข้อมูลของการทอยลูกเต๋า M ครั้ง ให้หาว่าเมื่อทอยได้ผลตามลำดับดังกล่าวแล้ว คุณจะอยู่ที่ตำแหน่งใด

ข้อมูลนำเข้า

บรรทัดแรกระบุจำนวนเต็มสองจำนวน N และ M ($1 \leq N \leq 1,000$; $1 \leq M \leq 1,000$)

ในบรรทัดถัดไประบุข้อมูลของตารางเกมเป็นจำนวนเต็ม N จำนวน กล่าวคือ ในบรรทัดที่ 2 จะระบุจำนวนเต็ม $X_1 X_2 \dots X_N$ เมื่อ X_i แทนตัวเลขในตารางเกมช่องที่ i สำหรับ $i=1, \dots, N$

อีกบรรทัดถัดไประบุผลการทอยลูกเต๋า โดยจะระบุเป็นจำนวนเต็มมีค่าตั้งแต่ 1 ถึง 6 บรรทัดจำนวน M ตัว

หมายเหตุ: การอ่านข้อมูลหลายตัวในบรรทัดเดียวกันสามารถใช้ `scanf` อ่านทีละตัวได้เหมือนปกติ

ข้อมูลส่งออก

มีหนึ่งบรรทัด ระบุหมายเลขช่องที่ผู้เล่นอยู่ ถ้าผู้เล่นออกนอกตารางพอดีเนื่องจากได้ค่าติลล ให้ตอบ 0

ตัวอย่าง 1

Input	Output
5 3 0 0 0 0 0 3 3 2	5

ตัวอย่าง 2

Input	Output
5 2 2 0 0 0 0 1 1	4

ตัวอย่าง 3

Input	Output
5 2 0 0 0 -10 0 2 2	0

ตัวอย่าง 4

Input	Output
5 2 0 2 0 -10 0 1 1	4

ตัวอย่าง 5

Input	Output
5 2 0 2 0 -10 0 3 6	5